TU WIEN Informatics

LLM Calibration

A Dual Approach of Post-Processing and Pre-Processing Calibration Techniques in Large Language Models for Medical Question Answering

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieurin

im Rahmen des Studiums

Medizinische Informatik

eingereicht von

Bettina Vogl, BSc.
Matrikelnummer 01648408

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Univ.Prof. Dr. Allan Hanbury, Institue of Information Systems Engineering, Technische Universtität Wien
Zweitbetreuung: Priv.Doz. Mag. Dr. Matthias Samwald, Institute of Artificial Intelligence, Medical University of Vienna

Wien, 8. April 2024

Bettina Vogl | Allan Hanbury

TU WIEN Informatics

LLM Calibration

A Dual Approach of Post-Processing and Pre-Processing Calibration Techniques in Large Language Models for Medical Question Answering

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieurin

in

Medical Informatics

by

Bettina Vogl, BSc.
Registration Number 01648408

to the Faculty of Informatics

at the TU Wien

Advisor: Univ.Prof. Dr. Allan Hanbury, Institue of Information Systems Engineering, Technische Universtität Wien
Second advisor: Priv.Doz. Mag. Dr. Matthias Samwald, Institute of Artificial Intelligence, Medical University of Vienna

Vienna, April 8, 2024

Bettina Vogl | Allan Hanbury

Erklärung zur Verfassung der Arbeit

Bettina Vogl, BSc.

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 8. April 2024

Bettina Vogl

Abstract

This thesis investigates the performance of Large Language Models in answering medical multiple-choice questions and explores strategies to enhance their accuracy, confidence estimation, and calibration. Specifically, we analyze the capabilities of GPT-3.5 and Cohere using the MedMCQA dataset, focusing on prompting techniques, revision strategies, and post-processing calibration methods. Our goals include assessing the efficacy of Chain of Thought prompting, examining the relationship between model confidence and correctness, and evaluating post-processing calibration techniques such as Platt Scaling, Beta Calibration, and Isotonic Regression.

Findings reveal GPT-3.5's superior accuracy compared to Cohere in medical question-answering. However, CoT prompting did not significantly improve model performance, suggesting its limited effectiveness in this context. Model confidence correlated with answer accuracy, but discrepancies between predicted and actual performance underscored the importance of robust calibration methods. Revision strategies marginally improved accuracy, with models adjusting responses when prompted to reconsider. Post-processing calibration techniques, particularly Isotonic Regression, demonstrated significant improvements in alignment between predicted probabilities and actual outcomes, enhancing model reliability.

Contents

CHAPTER 1

Introduction

In recent years, Large Language Models (LLMs) have revolutionized natural language processing (NLP) by exhibiting remarkable capabilities in understanding and generating human-like text. These advanced AI models, trained on vast amounts of textual data, have demonstrated proficiency across a wide range of tasks, from language translation to question-answering [OA24].

In the domain of healthcare, the potential of LLMs to comprehend and analyze medical information has garnered significant attention. Medical question-answering, in particular, presents a challenging yet crucial task, with implications for clinical decision-making, patient care, and medical education. The ability of LLMs to accurately interpret and respond to medical queries has the potential to streamline information retrieval, support diagnostic processes, and facilitate evidence-based practice.

This thesis focuses on evaluating the performance of LLMs, specifically GPT-3.5 and Cohere, in answering medical multiple-choice questions. By leveraging the MedMCQA dataset, I aim to assess the efficacy of different prompting strategies, confidence estimation techniques, revision strategies, and post-processing calibration methods in enhancing the accuracy, reliability, and calibration of LLM predictions.

Through empirical analysis and experimentation, I seek to elucidate the strengths and limitations of LLMs in medical question-answering tasks, elucidate the effectiveness of various strategies in optimizing model performance, and contribute to the ongoing discourse on the integration of AI technologies in healthcare.

By elucidating the nuances of LLM performance in medical question-answering and exploring avenues for improvement, this research endeavors to inform future developments in AI-assisted clinical decision support systems, ultimately advancing the quality and efficacy of healthcare delivery.

1.1 Research questions

The overarching goal of this research is to evaluate the effectiveness of calibration methods compared to Chain-of-Thought Reasoning in enhancing the trustworthiness of Large Language Models in the diagnostic domain. The hypothesis posits that Chain-Of-Thought prompting methods can achieve superior calibration compared to traditional post-processing calibration methods.

- How can post-processing calibration contribute to rendering LLMs more reliable and trustworthy for applications in clinical settings?
- To what extent do various post-processing calibration techniques exhibit comparative advantages and limitations in the calibration of LLMs for diagnostic purposes?
- Exploring the potential of Chain-of-Thought prompting: Can a more finely tuned calibration be achieved compared to traditional post-processing calibration methods? What is the procedural approach to achieving this calibration with the right prompting strategies?

CHAPTER 2

Literature Review

This chapter offers a comprehensive exploration of key themes and methodologies within the realm of natural language processing which are important for this thesis. It covers fundamental concepts such as large language models, logistic probabilities, and post-processing calibration methods. Additionally, the chapter discusses innovative techniques like Chain-of-Thought prompting, which aims to mimic human problem-solving strategies to bolster the reasoning capabilities of LLM models, and also introduces essential evaluation metrics for these models, like the Brier score and Expected Calibration Error.

2.1 Large Language Models

Large language models are advanced artificial intelligence systems designed to understand, generate, and interact with human language at a scale and complexity that closely mimics human language comprehension and production. These models are trained on vast datasets comprising text from the internet, books, articles, and other sources, enabling them to grasp a wide range of linguistic structures, styles, and content. The training process involves adjusting the parameters of the model to minimize the difference between the model's predictions and the actual data [TTE+23].

LLMs are built using deep learning frameworks, especially transformers, which are neural network models tailored for handling sequential data like text. Transformers utilize self-attention to weigh the importance of each word or token in a sequence relative to all others, allowing them to capture complex language dependencies efficiently. With multiple layers consisting of attention heads and feedforward neural networks, transformers can learn hierarchical representations of input data, from basic details to higher-level meanings. This enables LLMs to understand not only the structure of sentences but also their semantic nuances, producing coherent and contextually relevant text.

Pre-training is crucial for LLMs, involving exposing the model to large amounts of text data to learn language intricacies. During pre-training, the model predicts the next word or token in a sequence based on preceding context, known as masked language modeling. This helps the model grasp language patterns effectively, aiding its performance in various text generation tasks. Fine-tuning is another essential step where the pre-trained model adapts to specific tasks by further training on task-specific data. Through fine-tuning, the model can specialize in various applications, such as text classification, language translation, or text generation, by adjusting its parameters to better align with the target task's objectives [TTE+23].

2.2 Logistic Probabilities

At the core of LLMs, in tasks like text generation or next-word prediction, is the model's ability to assign probabilities to potential next words or tokens in a sequence. This is achieved through the softmax function, which is a generalization of logistic regression to multiple classes. For a given input sequence, the LLM processes it through multiple layers of the network to produce a set of scores (logits) for each token in the vocabulary. The softmax function then converts these logits into probabilities by taking the exponential of each logit, followed by normalizing these values so that they sum up to one. The formula for the softmax function is as follows:

$$P(y_i|\mathbf{x}) = \frac{e^{z_i}}{\sum_j e^{z_j}} \tag{2.1}$$

where $P(y_i|\mathbf{x})$ is the probability of the i-th token given the input sequence $\mathbf{x}$, e^{z_i} is the exponential of the score (logit) for the i-th token, and the denominator is the sum of exponentials of scores for all possible tokens in the vocabulary [PLSL17].

When generating text, an LLM selects the next token based on these logistic probabilities, often using techniques like sampling or beam search to ensure diversity and coherence in the generated text. This probabilistic approach allows LLMs to produce text that is not only grammatically correct but also contextually appropriate and varied [ZXG+24].

By examining these probabilities, researchers and practitioners can gain an understanding of the model's preferences, biases, and uncertainties. For instance, a high probability assigned to a specific word or sequence in a given context can indicate the model's confidence in its relevance or appropriateness, reflecting its learned associations and linguistic patterns. Conversely, a more uniform distribution of probabilities across multiple options might signal uncertainty or a lack of clear context. Analyzing changes in these probabilities in response to slight modifications in input can also reveal the sensitivity and robustness of the model to variations in language use. Thus, logistic probabilities serve not only as a mechanism for text generation and language understanding within

LLMs but also as a diagnostic tool that can help developers and researchers refine models, mitigate biases, and improve performance by offering a quantitative measure of the model's linguistic capabilities and limitations [ZXG+24].

2.3 Post-Processing Calibration Methods

2.3.1 Calibration

Post-processing calibration methods in LLMs are techniques applied after a model has been trained to align its confidence levels with the actual likelihood of predictions being correct. For a model to be well-calibrated, the confidence it expresses in its predictions should accurately reflect the true probabilities of those predictions being correct. In other words, if a model predicts an event with 70% confidence, that event should occur roughly 70% of the time if the model is well-calibrated [Fer22].

Calibration is crucial for several reasons:

- Trustworthiness: Users can trust and rely on the model's predictions and their associated confidence scores.
- Decision Making: Accurate confidence estimates are essential for risk-sensitive applications where decisions are made based on model predictions and their uncertainties.
- Comparability: Well-calibrated models provide a level playing field for comparing the performance of different models, especially in probabilistic tasks [Fer22].

To evaluate if an LLM is well-calibrated, one can use

- Calibration Metrics: Metrics like Brier Score (see more in Subsection 2.5.1), Expected Calibration Error (ECE) (see more in Subsection 2.5.2), and Log Loss provide quantitative measures of how well a model's confidence levels match with actual outcomes.
- Reliability Diagrams: These are graphical representations that plot predicted confidence levels against the actual fraction of predictions that were correct. For a perfectly calibrated model, this would result in a straight line at a 45-degree angle, indicating that the predicted probabilities match the observed frequencies [Fer22].

2.3.2 Calibration Techniques

Logistic Regression

Logistic regression is a statistical method used for binary classification that models the probability of a binary response based on one or more predictor variables (or features). It

is used to predict the probability that a given input belongs to a particular category (class 1 or 0). The logistic regression model applies a logistic function to a linear combination of the input features to produce a probability score between 0 and 1 [NC07].

Given an input feature vector $X = [x_1, x_2, ..., x_n]$, the logistic regression model estimates the probability (P) that the target variable Y is in a particular class (typically class 1), as shown in equation 2.2.

$$P(Y = 1|X) = \frac{1}{1 + \exp(-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n))} \tag{2.2}$$

Here, $\beta_0, \beta_1, ..., \beta_n$ are the parameters of the model, including the intercept β_0 and the coefficients $\beta_1, ..., \beta_n$ for each input feature $x_1, ..., x_n$. These parameters are estimated from the training data using a maximum likelihood estimation method, which aims to find the parameter values that make the observed data most probable.

The logistic function, also known as the sigmoid function, ensures that the output of the model is always in the range (0, 1), making it interpretable as a probability. This probability can then be used to make a classification decision, typically by selecting a threshold value (often 0.5) above which the model predicts class 1, and below which it predicts class 0 [NC07].

Platt Scaling

Platt scaling, also known as Platt calibration, is a method used to transform the output scores from a classification model (such as a support vector machine or other models that produce non-probabilistic outputs) into a probability distribution over classes [Pla00]. It is particularly useful for models that output scores which cannot be directly interpreted as probabilities, to calibrate them such that the scores represent the probability of a particular class given an input feature vector. The basic idea behind Platt scaling is to fit a logistic regression model to the scores output by the classifier. This is done using the output scores as the features, and the true class labels as the targets during the logistic regression training process.

Given a classifier that outputs a score $f(x)$ for a given input x, the Platt scaling algorithm models the probability that the target $y = 1$ (assuming a binary classification problem with labels $y \in \{0, 1\}$) given the score $f(x)$ as shown in equation 2.3 [Pla00].

$$[P(y = 1|f(x)) = \frac{1}{1 + \exp(Af(x) + B)} \tag{2.3}$$

Here, A and B are scalar parameters optimized through maximum likelihood estimation applied to the dataset of classifier scores $f(x)$ and their true labels, mapping them to their corresponding probabilities.

Isotonic Regression

Isotonic Regression is a non-parametric calibration technique used to adjust the predictions from a classification or regression model so that they form a monotonic sequence, aligning more closely with the observed frequencies of outcomes. This method is particularly useful for transforming non-probabilistic model outputs or those from models whose output does not naturally represent probabilities, into calibrated probabilities that more accurately reflect the true likelihood of each class given an input feature vector. Unlike Platt scaling, which fits a logistic regression model to the classifier's scores, Isotonic Regression does not assume any specific functional form between the scores and the target probabilities [ZE02].

Given a set of predictions $f(x)$ from a model for inputs x, and the corresponding true outcomes y, Isotonic Regression seeks to find a monotonic function g that minimizes the difference between $g(f(x))$ and the true outcomes y. The goal is to adjust $f(x)$ such that the adjusted predictions $g(f(x))$ are non-decreasing (or non-increasing, depending on the problem context) with respect to $f(x)$ and as close as possible to the actual observed probabilities of the outcomes.

The process involves sorting the data by the model's predictions $f(x)$, then finding a piecewise constant function g that is monotonically increasing with $f(x)$ and minimizes a loss function, typically the mean squared error, between the adjusted predictions $g(f(x))$ and the true labels y. This results in a stepwise non-linear transformation that calibrates the original model's outputs into probabilities that are better aligned with the observed distribution of the outcomes.

Isotonic Regression is particularly useful when the relationship between the model scores and the true probabilities is known to be monotonic but may not follow a specific parametric form, making it a flexible tool for calibration in various applications where maintaining the order of predictions is crucial [ZE02].

Beta Calibration

Beta Calibration is a method used to calibrate the output probabilities of a binary classification model, ensuring that the predicted probabilities accurately reflect the true likelihood of the outcomes. This technique is particularly useful for adjusting the outputs of models that already produce probabilistic predictions but might not be well-calibrated, meaning the predicted probability of an event occurring does not match the observed frequency of that event. Unlike Platt scaling, which fits a logistic regression to the scores, Beta Calibration applies a more flexible transformation that can capture a wider range of calibration issues [KSFF17].

Given a set of predicted probabilities p from a model and the corresponding true binary outcomes y, Beta Calibration seeks to adjust p using a transformation that is based on the Beta distribution 2.4.

$$\hat{p} = \frac{(p^{\alpha})^{a}}{(p^{\alpha})^{a} + ((1-p)^{\beta})^{b}} \tag{2.4}$$

Here, α and β are parameters that adjust the shape of the Beta distribution, allowing the calibration to capture different types of miscalibration patterns, such as sigmoid-shaped or reverse sigmoid-shaped deviations from perfect calibration. The parameters a and b are additional scaling factors that provide further flexibility in the calibration process. These parameters are typically estimated using maximum likelihood estimation on a calibration dataset, which consists of the model's predicted probabilities and the true outcome labels [KSFF17].

Gaussian Process Calibration

Gaussian Process (GP) Calibration, similar to Platt scaling, is a technique aimed at refining the output of a model to produce well-calibrated probability estimates, but it specifically leverages the framework of Gaussian Processes. While Platt scaling applies a logistic regression model to adjust the scores from a classifier, Gaussian Process Calibration uses a Gaussian Process model to achieve a more nuanced and flexible calibration, especially beneficial when dealing with complex, non-linear relationships between model scores and true probabilities [CPH23].

Given a model that outputs a score $f(x)$ for a given input x, Gaussian Process Calibration seeks to model the relationship between these scores and the true probabilities of belonging to a particular class (in the context of binary classification, for instance) through a Gaussian Process.

$$P(y = 1|f(x)) = G(f(x); \mu, \sigma^2) \tag{2.5}$$

Here, G represents a Gaussian Process with mean function μ and variance σ^2, which are functions of the input score $f(x)$. The goal is to learn the parameters of this Gaussian Process (typically the parameters defining μ and σ^2) from the data, such that the process accurately maps the original model scores to probabilities.

The calibration process involves training the Gaussian Process on a dataset consisting of the original model's scores and the true outcomes. The GP learns a function that, given a new score from the model, can predict a calibrated probability of the positive class. This approach is particularly powerful due to the non-parametric nature of Gaussian Processes, which allows for modeling complex, non-linear relationships without explicitly defining the form of such relationships beforehand [CPH23].

Hosmer Lemeshow Test

The Hosmer-Lemeshow Test is a statistical test used for evaluating the goodness-of-fit of logistic regression models. The test is based on the principle that a well-fitted model should show no significant difference between the observed and predicted probabilities of the outcome variable across different groups of predictor variables [HLM+20].

The test involves the following steps:

1. The data is divided into deciles (or other groups) based on the predicted probabilities of the outcome variable.
2. For each group, the observed number of events (i.e., instances of the outcome variable) and non-events are counted.
3. The expected number of events and non-events in each group are calculated based on the model's predicted probabilities.
4. A chi-square statistic is calculated based on the differences between the observed and expected counts. See Equation 2.6.
5. The p-value for the chi-square statistic is calculated. If the p-value is less than a chosen significance level (e.g., 0.05), the null hypothesis that the model fits the data well is rejected.

$$H = \sum_{g=1}^{G} \frac{(O_{1g} - E_{1g})^2}{E_{1g}} + \frac{(O_{0g} - E_{0g})^2}{E_{0g}} \tag{2.6}$$

where O_{1g} and O_{0g} are the observed numbers of events and non-events in group g, respectively, E_{1g} and E_{0g} are the expected numbers of events and non-events in group g, respectively, and G is the number of groups [HLM+20]..

Calibration Curve

Calibration curves are a visual tool used to assess the calibration of a predictive model, especially in the context of binary classification. The curve is a plot that compares the predicted probabilities of a model against the actual proportions of positive outcomes observed in the data .

A perfectly calibrated model would result in a calibration curve that is a straight diagonal line from (0,0) to (1,1), indicating that for any given predicted probability, the proportion of positive outcomes in the data is exactly the same. For instance, if a model predicts a probability of 0.7 for a set of instances, then around 70% of those instances should actually belong to the positive class [Fer22].

To construct a calibration curve, the following steps are typically followed:

- Sort the instances in the dataset based on the predicted probabilities from the model.
- Partition the sorted instances into a number of bins (e.g., 10 bins). Each bin contains instances with predicted probabilities within a certain range.
- For each bin, calculate the mean predicted probability and the actual proportion of positive outcomes.
- Plot the mean predicted probabilities (x-axis) against the actual proportions (y-axis). Each bin corresponds to one point on the plot.

The shape of the calibration curve can provide insights into the type of miscalibration that the model suffers from. For instance, a curve below the diagonal line suggests that the model is overconfident (i.e., it predicts probabilities higher than the actual proportions), while a curve above the diagonal line suggests that the model is underconfident (i.e., it predicts probabilities lower than the actual proportions) [Fer22].

Receiver Operating Characteristic

Receiver Operating Characteristic (ROC) curves are instrumental in assessing the performance of binary classification models. On these plots, the false positive rate (FPR) - the proportion of negative instances misclassified as positive - forms the x-axis. Simultaneously, the true positive rate (TPR), synonymous with sensitivity or recall, forms the y-axis. This rate measures the proportion of positive instances correctly identified.

The ROC curve depicts the interplay between TPR and FPR across varying classification thresholds. A model demonstrating perfect discrimination will display an ROC curve that adheres closely to the y-axis and plot's upper boundary, reflecting a TPR of 1 and an FPR of 0. In stark contrast, a model devoid of any discriminatory capacity will exhibit an ROC curve resembling a 45-degree diagonal line extending from the plot's bottom-left to the top-right corner.

Area Under the Curve

The Area Under the Curve (AUC) is a commonly used metric in machine learning that measures the two-dimensional area underneath the entire ROC curve from (0,0) to (1,1). It provides an aggregate measure of performance across all possible classification thresholds, serving as a single scalar value summarizing the ROC curve [Fer22].

AUC ranges in value from 0 to 1. An AUC of 1 signifies that the model has perfect discriminatory capacity; it can perfectly distinguish between positive and negative instances. Conversely, an AUC of 0.5 suggests that the model has no discriminatory capacity and is as good as random guessing.

2.4 Chain-of-Thought Prompting

Chain-Of-Thought (CoT) prompting is a technique used in the field of natural language processing (NLP) and, more specifically, in the interaction with LLMs to enhance their ability to solve complex problems, including arithmetic, logic puzzles, and reasoning tasks. This method involves structuring the input prompt to encourage the model to generate intermediate steps or reasoning paths that lead to the final answer, rather than attempting to reach the conclusion in a single step. The fundamental concept behind Chain-Of-Thought prompting is to mimic human problem-solving processes, where complex problems are often broken down into smaller, more manageable parts [WWS+22].

When employing Chain-Of-Thought prompting, the user crafts a prompt that not only asks the model to solve a problem but also to elaborate on the sequential steps or thought process that could logically lead to the solution. This approach helps in making the model's decision-making process more transparent and can significantly improve the model's performance on tasks that require multi-step reasoning or detailed explanations. [WWS+22].

Example Consider a complex arithmetic problem such as calculating the total cost of items bought in different quantities and prices. Instead of simply asking, "What is the total cost?", the prompt might be structured as:

"First, calculate the cost of 3 apples at $2 each. Then, add the cost of 5 bananas at $1 each. Finally, include the cost of 2 oranges at $1.50 each. What is the total cost?"

This prompt encourages the model to follow a clear, step-by-step reasoning path, detailing each calculation:

$$\begin{aligned} \text{Cost of apples} &= 3 \times \$2 = \$6 \\ \text{Cost of bananas} &= 5 \times \$1 = \$5 \\ \text{Cost of oranges} &= 2 \times \$1.50 = \$3 \end{aligned}$$

Thus, the total cost is calculated by summing up the individual costs:

$$\text{Total cost} = \$6 + \$5 + \$3 = \$14$$

Given a complex question or task Q, the Chain-Of-Thought prompting method can be conceptualized as guiding the model to generate a sequence of intermediate thoughts or steps $[T_1, T_2, ..., T_n]$ that logically connect the initial problem statement to the final answer A. The model is thus encouraged to output a narrative that reflects a reasoned path from Q through $[T_1, T_2, ..., T_n]$ to A, as shown in equation 2.7.

$$\text{Chain-Of-Thought: } Q \rightarrow [T_1, T_2, ..., T_n] \rightarrow A \tag{2.7}$$

In this approach, each T_i represents an intermediate thought or step that contributes to building a comprehensive understanding of how to approach and solve Q, ultimately leading to the answer A. This method not only aids in solving the given task more effectively but also in generating explanations that are more interpretable and educational for human users, aligning with the goal of making AI interactions more intuitive and insightful.

2.5 Evaluation Metrics

2.5.1 Brier Score

The Brier score is a post-processing calibration metric used to evaluate the accuracy of probabilistic predictions. It measures the mean squared difference between the predicted probabilities and the actual outcomes. The Brier score is particularly useful for assessing the performance of classification models that output probabilities for two or more classes. It can be applied to any model that generates probabilistic forecasts, providing a single-number summary that represents the model's accuracy.

Given a set of N predictions, where each prediction i consists of a predicted probability p_i that a certain event (e.g., belonging to a particular class) will occur, and the actual outcome of that event y_i (with $y_i = 1$ if the event occurs and $y_i = 0$ otherwise), the Brier score (BS) is calculated:

$$BS = \frac{1}{N} \sum_{i=1}^{N} (p_i - y_i)^2 \tag{2.8}$$

The Brier score ranges from 0 to 1, where 0 represents a perfect model that always predicts the actual outcomes with 100 % certainty, and 1 represents the worst possible model. A lower Brier score indicates better calibration and reliability of the model's probabilistic predictions. It effectively penalizes both overconfident and underconfident predictions that do not align with the actual outcomes, encouraging models not only to be accurate but also to have well-calibrated probability estimates.

2.5.2 Expected Calibration Error

Expected Calibration Error (ECE) is another post-processing calibration metric used to assess the reliability of probabilistic predictions made by classification models. ECE measures the discrepancy between the predicted probabilities and the actual outcomes, providing an aggregate measure of the difference between the predicted confidence of a model and its actual accuracy.

To compute it, predictions are first grouped into M equally spaced bins based on their predicted probability. Within each bin, the average predicted probability (p_{avg}) and the actual accuracy (a_{avg})—the fraction of correct predictions—are calculated. The ECE is then the weighted average of the absolute differences between the predicted probabilities and the actual accuracies across all bins, with the weights being the number of predictions in each bin. This can be mathematically represented as shown in equation 2.9.

$$ECE = \sum_{m=1}^{M} \frac{|N_m|}{N} |p_{\text{avg},m} - a_{\text{avg},m}| \tag{2.9}$$

where N_m is the number of predictions in bin m, N is the total number of predictions, $p_{\text{avg},m}$ is the average predicted probability in bin m, and $a_{\text{avg},m}$ is the actual accuracy in bin m.

The ECE metric ranges from 0 to 1, where 0 indicates perfect calibration (i.e., the predicted probabilities perfectly match the actual outcomes), and higher values indicate greater discrepancies between the model's confidence and its empirical accuracy.

Differences between Brier Score and ECE

The choice of binning strategy can significantly affect the ECE, potentially leading to different interpretations of model calibration. ECE is straightforward to interpret as it directly reflects the calibration error in a probabilistic model. It is highly sensitive to discrepancies in calibration, making it useful for models where calibration is critical. Brier Score, on the other hand, evaluates both the calibration and the resolution of the predictions, giving a more complete picture of prediction quality. By squaring the differences, the Brier Score is sensitive to the size of the prediction errors, heavily penalizing larger errors. But, while comprehensive, the Brier score does not isolate calibration errors specifically, which can be a drawback when calibration is the primary concern [Fer22].

CHAPTER 3

Theoretical Framework and Methodology

This chapter provides an approach to the research process, outlining the selection of Large Language Models and detailing the methodology employed for questioning, answering, and evaluating responses. It begins by explanation of the reason behind choosing specific LLMs, which are OpenAI's GPT-3.5 and Cohere Co.Generate. Subsequently, it describes the interrogation process using medical multiple-choice questioning datasets, such as MedMCQA and MedMC, along with specialized querying extensions and prompting strategies like Chain-of-Thought and Revision steps. The chapter further elucidates post-processing calibration methods including Platt Scaling, Isotonic Regression, as well as evaluation metrices like the Hosmer-Lemeshow Test, Calibration Curve, ROC Curve, Youden's Index, and Area Under the Curve.

3.1 Large Language Models

In the context of this research, the selection of appropriate Large Language Models is important. The primary criteria for selection was the ability of the models to return the logistic probability of each generated token, indicating the model's confidence level in its own response. This feature is integral to the objectives of my thesis, and hence, models that lack this capability were not considered.

Based on these prerequisites, I have chosen LLMs for my work:

- OpenAI GPT-3.5-turbo-instruct [Ope23]
- Cohere Co.Generate [Coh23]

3.2 Questioning

The methodology for my Master's thesis is primarily based on interrogating the Language Learning Model with questions sourced from the medical multiple choice questioning datasets.

3.2.1 MedMCQA

The MedMCQA is a Multiple Choice Question Answering dataset, specifically curated to address real-world medical entrance exam questions. With over 194,000 questions sourced from institutions like the All India Institute of Medical Science and the National Eligibility cum Entrance Test, both qualifying and ranking examinations in India, for students who wish to study as a postgraduage in the field of medicine [PUS22]. It encompasses both multiple-choice and single-choice questions, with the focus of this thesis being solely on single-choice questions.

For instance, a typical question could be: *"Thiamine deficiency causes lactic acidosis due to defect in the action of?"*, with 4 choices offered:

(a) Alpha - KG dehydrogenase

(b) Sorbitol reductase

(c) Lactate dehydrogenase

(d) **Pyruvate dehydrogenase**

From this dataset, a subset of 100 single-choice questions was extracted for the primary phase of inquiry. To enrich the analysis, a second subdataset comprising 1000 additional single-choice questions was incorporated, ensuring that none of the questions overlapped with those from the initial subset.

3.2.2 MedMC

The MedMC dataset was used as a control dataset on Cohere Co.Generate. Similar to MedMCQA, MedMC is a subset of the United States Medical Licensing Examination (USMLE), a examination for medical licensure in the USA, with around 12000 questions [JPO+20]. It includes single-choice questions, with five choices given. A typical question with its answer possibilities would be:

"An investigator is studying neuronal regeneration. For microscopic visualization of the neuron, an aniline stain is applied. After staining, only the soma and dendrites of the neurons are visualized, not the axon. Presence of which of the following cellular elements best explains this staining pattern?"

(a) Microtubule

(b) **Rough endoplasmic reticulum**

(c) Nucleus

(d) Lysosome

(e) Golgi apparatus

3.2.3 Query Extension

To gain insights into the model's decision-making process, I extend the query to include ranking the possible answers. This is done by appending the phrase:

Arrange the following letters in the order of the correct answer to the least correct answer based on the presented answer possibilities. Provide the sequence starting with the correct answer letter and proceeding to the less probable answers letter, but with a blank between, e.g. a b c d. Question: ...

3.2.4 Chain-of-Thought Step

For the Chain-Of Thought prompting strategy, different pre-questioning phrases are introduced to the model to guide its thought process. These include:

- Let's work this out in a step by step way to be sure we have the right answer.
- Carefully go through each step to make sure we achieve the correct outcome.
- "I'd appreciate it if you could tackle this task step by step for precision and the right outcome.
- Walk through the process step by step to confirm we're on the correct path.

3.2.5 Revision step

Subsequently, a secondary result review of the model's responses is conducted. This involves re-prompting the model with the original question, while also referring to its previous response. The model is then asked to reaffirm or revise its answer, and provide an explanation either way. This is done with the following phrase:

You did say in a previous response, that the correct answer is ... . Are you sure this is the correct answer? If you are, explain to me why, and if not, give me the correct answer.

This secondary evaluation phase is implemented twice. The first instance is performed on the entire dataset, while the second instance is only executed if the likelihood of the stated answer falls below a predetermined threshold. This threshold is discussed further in subsection Youlden's Index 3.4.6.

3.3 Answering

In listing 3.3 there is one response JSON. There are outputs from the LLM to the question, and added stats and calculations from my code. The JSON properties and details are described as following:

- *correct*: Did the LLM answer this question correct the first time?
- *correct_answer*: What would have been the correct answer?
- *stated_answer*: The selected answer in the first process
- *options*: The textual output of the LLM have been only the letters of the answer possibilities in the question. In example 3.3, this would have been "c d a b". Here we have for each token, e.g. each letter, some more characteristics:
  - *token*: The answer possibility
  - *likelihood*: the logistic probability for this token in the sequence "c d a b"
  - *likelihood_perc*: The logistic probability *likelihood* transformed into "normal" probability, so e.g. 0.8833 mean 88,33 %
  - *toplog_percentage*: Explained below in 3.3
  - *calculated_percentage*: Explained below in 3.3
- *input_question*: The input question from the MedMCQA with:
  - *question*: The question
  - *cop*: the correct answer possibility. 1: a, 2: b, 3: c, 4: d
  - *opa/opb/opc/opd*: Single Choice option a/b/c/d
- *restated*: Was the answer restated in the revision step process 3.2.5
- *response*: The response of the revision step process
- *restated_answer*: The restated answer of the revision step process
- *restated_correct*: The selected answer in the revision step process
- *state*: The result revision state, defined in 3.1
- *percentage_new_calc*: The probability of the restated answer from the first step, calculated
- *percentage_new_topl*: The probability of the restated answer from the first step, retrieved from the toplog

Token probabilities

The OpenAI model has the capacity to yield not only the most probable token but also to return the other highly probable tokens at each token position. We can map these 'toplogs' to each subsequent token if they exist. However, as Cohere lacks this feature, we introduce a *calculated_percentage*. This percentage is computed based on the previous likelihoods.

Let's denote the *calculated_percentage* for $options[i]$ as P_i, and the $options[i].likelihood\_percentage$ as L_i.

We can express the relationships between these variables using the following formulae:

1. The first calculated percentage is simply equal to the first likelihood percentage:

$$P_0 = L_0$$

2. For the second token:

$$P_1 = (1 - P_0) \times L_1$$

3. For the third token:

$$P_2 = (1 - (P_0 + P_1)) \times L_2$$

4. For the fourth token:

$$P_3 = (1 - (P_0 + P_1 + P_2)) \times L_3$$

In its general form:

$$P_i = (1 - (P_0 + P_1 + .. + P_{i-1})) * L_i$$

or equivalently:

$$P_i = (1 - \sum_{k=0}^{i-1} P_k) \times L_i$$

The above expressions allude to the computation of the probability of each option considering the likelihood percentage of each option and the probabilities of preceding options. The close alignment of these calculated probabilities to the given *toplog_percentages* permits us to use these as the probability of a token, thus also defining the confidence of the model in terms of the correct answer. Therefore, the *calculated_percentage* could be deemed as the model's confidence score ascribed to each token, making this a potential way of quantifying a model's certainty in this answer.


```
{
    "id": 284,
    "llm": "GPT-3.5",
```

```
    "correct": false,
    "correct_answer": "d",
    "stated_answer": "c",
    "percentage_old_calc": 0.8833222675944602,
    "percentage_old_topl": 0.8833222675944602,
    "options": [
        {
            "token": "c",
            "likelihood": -0.124065176,
            "likelihood_perc": 0.8833222675944602,
            "calculated_percentage": 0.8833222675944602,
            "toplog_percentage": 0.8833222675944602
        },
        {
            "token": "d",
            "likelihood": -0.18756504,
            "likelihood_perc": 0.828975199879996,
            "calculated_percentage": 0.09672294654242705,
            "toplog_percentage": 0.06709673465825658
        },
        {
            "token": "a",
            "likelihood": -0.45635712,
            "likelihood_perc": 0.6335875299128606,
            "calculated_percentage": 0.01264310348494966
        },
        {
            "token": "b",
            "likelihood": -0.00026431592,
            "likelihood_perc": 0.9997357190083754,
            "calculated_percentage": 0.007309750039493698,
            "toplog_percentage": 0.030375773883496462
        }
    ],
    "input_question": {
        "question": "Best X ray view to visualize superior
           orbital fissure is",
        "cop": 4,
        "opa": "Anteroposterior",
        "opb": "Basal",
        "opc": "Towne's",
        "opd": "Caldwell luc view",
        "subject_name": "Ophthalmology",
```

```
        "topic_name": "Diseases of orbit, Lids and lacrimal
           apparatus",
        "id": "6dd7bd1f-4375-4df3-b502-3e19cc4f0305",
        "choice_type": "single"
    },
    "restated": true,
    "response": "The correct answer is not c) Towne's view. The
        best X-ray view to visualize the superior orbital
       fissure is actually d) Caldwell luc view. The Caldwell
       luc view is a specialized X-ray view that is taken with
       the patient's head tilted slightly backward. This view
       allows for better visualization of the superior orbital
       fissure, which is a small opening in the skull located
       behind the eye socket. This view is particularly useful
       for assessing fractures or other abnormalities in this
       area. I hope this clarifies things for you. Thank you
       for pointing out the error, and I appreciate your
       understanding.",
    "restated_answer": "d",
    "restated_correct": true,
    "state": 3,
    "percentage_new_calc": 0.09672294654242705,
    "percentage_new_topl": 0.06709673465825658
}
```


3.3.1 Answer Evaluation

For the result revision, there are different states possible, based on the first and second answer for the question of the LLM. I defined those in Table 3.1.

State number	Correct First Time	Correct Second Time	Description
0			No new answer given
1	1	1	Correct both times, "stood on answer"
2	1	0	Switches from correct to false answer, corrected itself incorrectly
3	0	1	Corrected itself correctly
4	0	0	Stood on answer incorrectly
5	0	0	Recorrected to another false answer

Table 3.1: State definition after revision step

3.4 Post-Processing Calibration

3.4.1 Platt Scaling

Platt Scaling was implemented on the dataset. The true labels of the dataset represent whether the model accurately answered the question, while the scores correspond to the toplog or calculated probabilities. We have returned calibrated probabilities.

3.4.2 Isotonic Regression

In a similar vein to Platt Scaling, Isotonic Regression was also executed.

3.4.3 Hosmer Lemeshow Test

Subsequently, the Hosmer-Lemeshow test was performed on both the uncalibrated and Platt Scaling-calibrated probabilities. This test serves as a measure of the effectiveness of the Platt Scaling method. A low p-value, such as less than 0.005, as a result of the Hosmer Lemeshow test, suggests a poor alignment between the predicted probabilities and the data.

3.4.4 Calibration Curve

The calibration curve is plotted with the predicted probabilities on the x-axis and the actual probabilities on the y-axis. A perfectly calibrated model will have a calibration curve that lies on the 45-degree diagonal line. Any deviation from this line indicates a discrepancy between the predicted and actual probabilities.

3.4.5 Receiver Operating Characteristic

The Receiver Operating Characteristic (ROC) curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings. The area under the ROC curve (AUC) is a measure of how well a parameter can distinguish between two diagnostic groups (diseased/normal) [REC20].

3.4.6 Youden's index

The Youden index is a statistic used to evaluate the performance of diagnostic tests. It is calculated as the difference between TPR and FPR. Essentially, it measures the ability of a test to correctly identify positive cases while avoiding false positives. A higher Youden index therefore means better test performance. Youden's Index is closely related to the ROC curve, as it represents a single summary statistic derived from the ROC curve, maximizing the vertical distance from the diagonal line, indicating the optimal trade-off between sensitivity and specificity [FFR05].

3.4.7 Area Under the Curve

The Area Under the Curve represents the degree or measure of separability. It tells how much model is capable of distinguishing between classes. Higher the AUC, better the model is at predicting 0s as 0s and 1s as 1s. An excellent model has AUC near to the 1 which means it has good measure of separability. A poor model has AUC near to the 0 which means it has worst measure of separability. In fact it means it is reciprocating the result. It is predicting 0s as 1s and 1s as 0s. And when AUC is 0.5, it means model has no class separation capacity whatsoever [REC20].

CHAPTER 4

Results

This results chapter examines the performance of the Language Learning Models in terms of correctness, and confidence, both in their initial responses and after employing the Chain-Of-Thought prompting strategies. A comprehensive overview of the correctness of LLM responses is given, examining how well they answered the medical multiple-choice questions and their capability to either revise or maintain their responses through the Result Revision process. Furthermore, the calibration of the models is evaluated, both before and after applying post-processing calibration techniques such as Platt Scaling, Beta Calibration and Isotonic Regression, to enhance their predictive accuracy.

4.1 Correctness Overview

The correctness comparison is evaluated on the basis of the accuracy of the responses provided by the two Large Language Models. GPT3.5 exhibited an accuracy of 58 % in answering the first dataset of 100 questions, and this slightly declined to 54.2% when the model was subjected to the larger dataset comprising of 1000 questions, see Figure 4.1.

Contrarily, the performance of Cohere was significantly lower in comparison. Within a set of 100 questions, the model was able to correctly answer only 20%. This low performance was consistently reflected in another test conducted using an alternative Medical Question Answering dataset, MedMC. Cohere again scored low with only 26 correct answers out of a set of 100 questions.

In the case of MedMC, with five answer possibilities, random guessing would statistically yield a correct answer 1 out of 5 times, or 20% of the time. This is because there is one correct answer among five choices. Now, in MedMCQA, where there are only four answer possibilities, the probability of randomly guessing the correct answer increases slightly. With one correct answer among four choices, random guessing would yield a correct answer 1 out of 4 times, or 25% of the time.

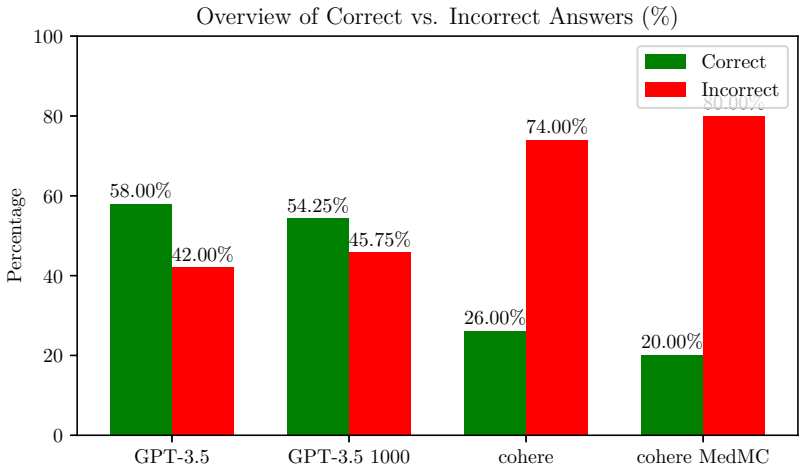


Figure 4.1: Correctness for GPT-3.5 on 100 and 1000 MedMCQA questions, cohere on 100 MedMCQA and 100 MedMC questions.

Therefore, when looking at the scoring percentages alone, it might seem like cohere's responses are guided more by random guessing. This is because the score of 26% in MedMCQA aligns more closely with the probability of random guessing (1 out of 4), same for the score of 20% in MedMC (1 out of 5). Thus, model's responses seemed to be guided more by random guessing rather than a knowledgeable deduction, see also Figure 4.1.

4.1.1 Chain-of-Thought Correctness

The Chain-of-Thought prompting technique did not lead to a significant improvement in the performance of GPT-3.5. In fact, there was a marginal decrement in the correctness of the results generated by the model, see Figure 4.2.

In contrast, Cohere demonstrated no improvement in performance using the Chain-of-Thought prompting.

But strikingly, both models did not demonstrate any notable difference in their response to different Chain-of-Thought prompts. Their performance remained consistent regardless of the specific prompt used. As a result, in the subsequent data presentation, I will only incorporate data obtained through the prompt, "Let's work this out in a step by step way to be sure we have the right answer".

4.2 Confidence overview

The confidence levels extrapolated from the probabilities given by the GPT-3.5 model, either from the top logs or computed, yields a noteworthy observation. There appears to

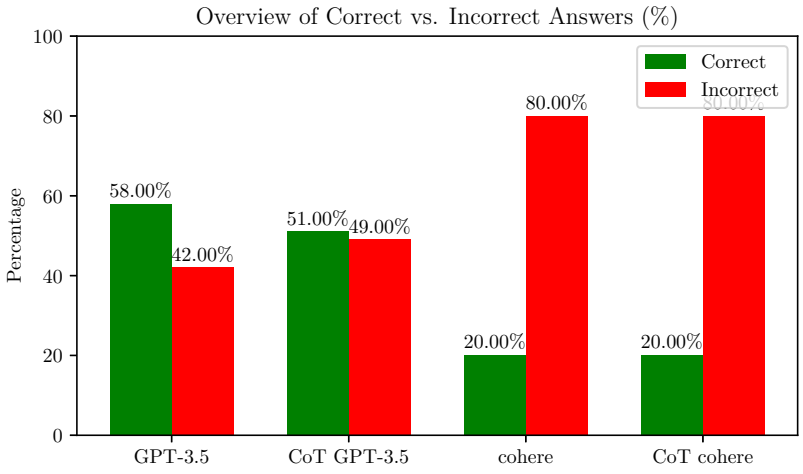


Figure 4.2: Correctness for GPT-3.5 and Cohere on 100 MedMCQA questions, with and without Chain-of-Thought prompting.

be a direct relationship between the model's confidence and the likelihood of the answer being correct: It can be inferred that higher confidence levels of the model correspond to a higher likelihood of the answer being correct, see Figure 4.3.

As we have seen before, Chain-of-Thought prompting did not improve in the correctness of the results, and here we see it also does not affect the confidence of the model in its answers. See Figure 4.4a and Figure 4.4b.

4.3 Result revision

The process of result revision was carried out twice. In the first instance, with the entire dataset, a reduction to only 26 % accuracy for 100 questions and 35.64 % was observed for GPT-3.5, as shown in Figure 4.5a.

Conversely, for cohere, a significant increase in the number of correct responses was noted.

Given the decrease in accuracy for GPT-3.5, a second round of result revision was introduced. This involved the calculation of Youden's index, which was used as a threshold parameter. The model's confidence in its initial response was evaluated against this threshold. If it fell below the threshold, the model was prompted to reassess the question. If the confidence level was above the threshold, the initially provided answer was accepted (state 0).

Post this secondary revision, there was a minor increase in the correct responses provided by GPT-3.5, a finding which can be seen in Figure 4.5b.

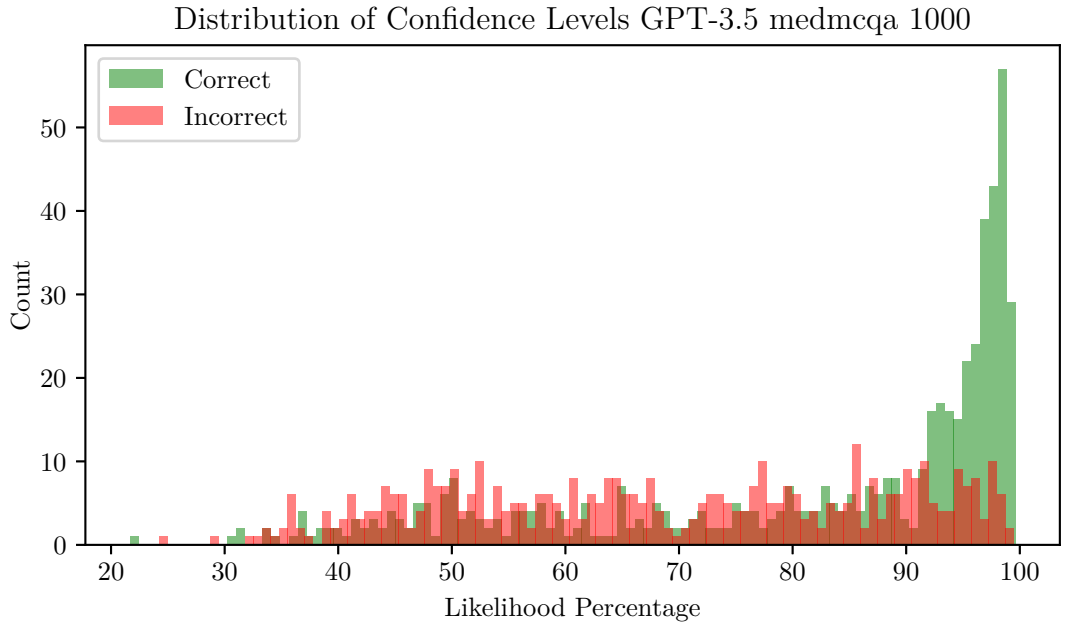
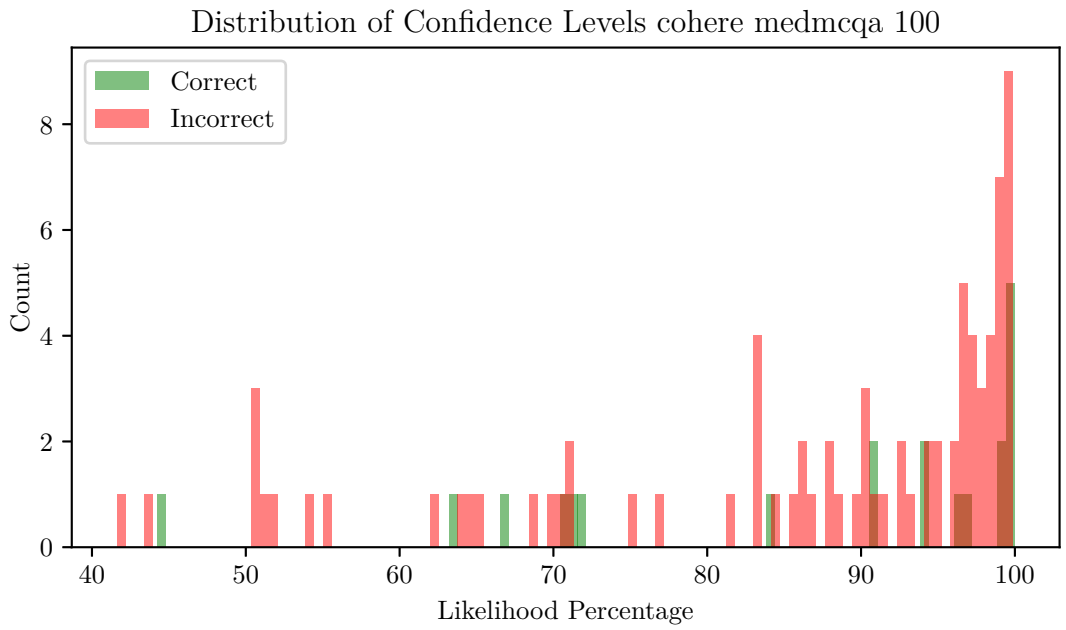


Figure 4.3: Histogram of GPT-3.5's confidence in its answers.

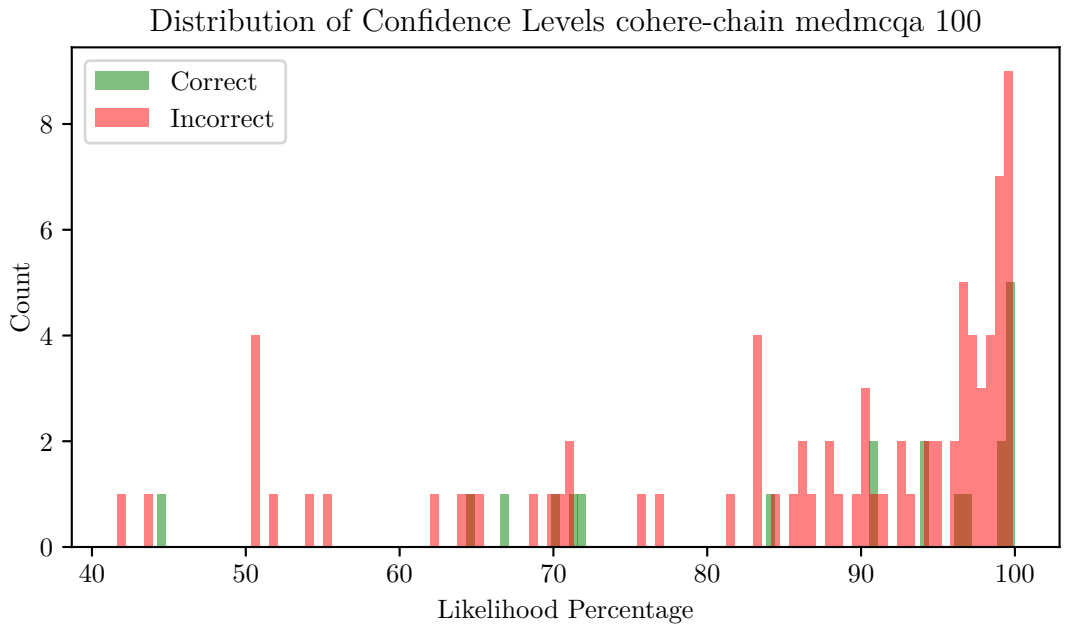
In order to reassess the shift between correct and incorrect answers, I established distinct states, see Table 3.1. The results for the dataset of 100 questions are illustrated in Figure 4.6a. It shows that there were more questions that were initially answered incorrectly but corrected in the revision process (state 3) as compared to questions which were initially correct but revised to incorrect answers (state 2). The difference between these states directly corresponds to the observed increment in accuracy.

To substantiate the observed elevation in accuracy, the same evaluation was performed on the dataset of 1000 questions. Regrettably, the improved accuracy was not maintained in this larger dataset, as illustrated in Figure 4.6b. Expressing this in terms of state numbers, there were more questions in state 2 (initially correct, revised to incorrect) than in state 3 (initially incorrect, corrected upon revision). To enhance accuracy, the situation needs to be reversed - with more questions improving from state 3 than deteriorating from state 2.

The analysis of the stages reveals an intriguing pattern: when confronted with result revision prompts, the model consistently responded with minimal resistance to altering its initial answer. Rarely did it firmly adhere to its original response, whether accurate (state 1) or inaccurate (state 4). Instead, it predominantly transitioned from an incorrect to a correct answer (state 2), vice versa (state 3) or from one incorrect answer to another one (state 5). This observation suggests that the prompt primarily serves to signal the model to modify its response, irrespective of its accuracy.

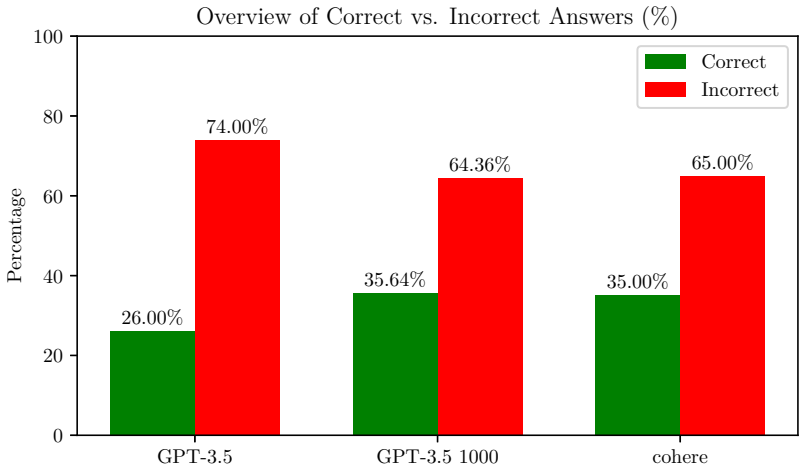


(a) Without Chain-of-Thought prompting

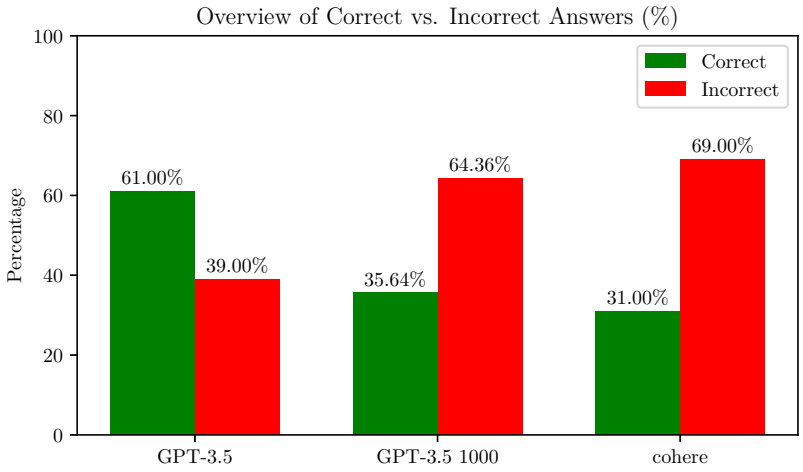


(b) With Chain-of-Thought prompting

Figure 4.4: Histograms comparing cohere's confidence in its answers



(a) Following the initial revision step, there was a significant drop in GPT-3.5's accuracy.



(b) Following the second revision step, there was a small increase in GPT-3.5's accuracy. After the second revision step with 1000 questions, we did not have the same effect we had on 100.

Figure 4.5: Comparison of accuracy changes after revision steps

4.4 Calibration Evaluation

The following analysis was constructed from the data after the previous result revision. Platt Scaling, Beta Calibration and Isotonic Regression were implemented. For each method, a training function was defined to fit the calibration model using the correctness, if the answer was answered correctly, as labels and probabilities of the model's confidence. Subsequently, an application function is used to leverage the trained model for calibrating probabilities.

4.4.1 Hosmer Lemeshow Test

The Hosmer Lemeshow Test yielded limited insights due to the absence of sufficient data points in many of the bins, particularly those representing uncalibrated probabilities. In statistical analysis, "bins" refer to predefined intervals into which data points are grouped based on their values. This suggests an uneven distribution of probabilities across the range of 0 to 1. While it's common to address such issues by adjusting the bins, such as by adding a small constant to prevent division by zero, this approach risks introducing artificial data points into empty bins, potentially skewing the results of the Hosmer-Lemeshow test. Ideally, empty bins should be excluded from calculations to maintain the test's integrity. However, given the significant number of empty bins observed, I opted to omit the Hosmer Lemeshow Test from my analysis. Instead, I focused on alternative indicators such as the Expected Calibration Error , Brier Score, and Calibration Curve for a more robust assessment.

4.4.2 Brier Score

Analysis of the Brier scores provided in Table 4.1 for the models GPT-3.5 (on 100 and 1000 questions) and Cohere, and calibration methods: Uncalibrated, CoT (Calibration over Time), Platt Scaling, Beta Calibration, and Isotonic Regression.

Brier score ranges from 0 to 1, with 0 indicating perfect accuracy (the predicted probabilities match the actual outcomes) and 1 indicating perfect inaccuracy (the predicted probabilities are completely off). So, the closer the Brier score is to 0, the better the predictions.

The uncalibrated models have relatively high Brier scores across the board, indicating suboptimal accuracy in probabilistic predictions. Cohere exhibits the highest Brier score among all models, suggesting the least accurate predictions among the uncalibrated models.

CoT is only available for GPT-3.5 on 100 questions and for Cohere. For GPT-3.5 it shows a slightly worse performance compared to the uncalibrated GPT-3.5 model, with a higher Brier score. Cohere has the same score for CoT as uncalibrated.

Platt Scaling improves the calibration of the predictions for all models compared to their uncalibrated versions. However, Cohere still exhibits the highest Brier score among all models even after Platt Scaling calibration. Beta Calibration further improves the calibration of the predictions, resulting in lower Brier scores compared to Platt Scaling for all models. Among all models, Cohere demonstrates the best performance after Beta Calibration. Isotonic Regression consistently outperforms both Platt Scaling and Beta Calibration for all models, showcasing the lowest Brier scores.

Isotonic Regression consistently provides the best calibration and accuracy improvement for all models, followed by Beta Calibration and Platt Scaling. However, even after calibration, Cohere tends to have higher Brier scores compared to the GPT-3.5 models, indicating potentially less accurate probabilistic predictions.

Model	GPT-3.5	GPT-3.5 1000	Cohere
Uncalibrated	0.205	0.270	0.618
CoT	0.275	-	0.618
Platt Scaling	0.190	0.222	0.160
Beta Calibration	0.164	0.210	0.155
Isotonic Regression	0.144	0.203	0.138

Table 4.1: Brier Score

4.4.3 Expected Calibration Error

As Brier Score, for the Expected Calibration Error applies: the smaller, the better. Let's analyze the Expected Calibration Error (ECE) scores provided in Table 4.2.

Model	GPT-3.5	GPT-3.5 1000	Cohere
Uncalibrated	0.174	0.224	0.660
CoT	0.241	-	0.618
Platt Scaling	0.134	0.073	0.002
Beta Calibration	0.076	0.026	0.074
Isotonic Regression	5.32e-17	2.05e-17	2.19e-17

Table 4.2: Expected Calibration Error

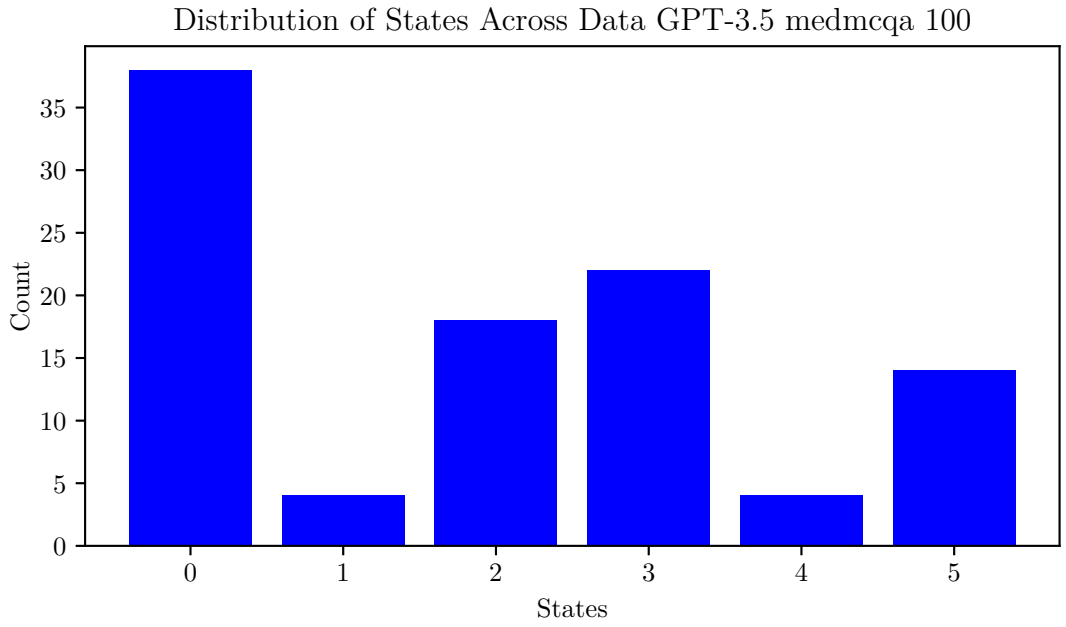
Similar to the Brier scores, the uncalibrated models exhibit relatively high ECE scores, indicating a lack of calibration in their probabilistic predictions. Cohere again demonstrates the highest ECE among all models, indicating the poorest calibration.

CoT for GPT-3.5 shows a higher ECE compared to its uncalibrated version, indicating worsened calibration. Cohere is slightly better for CoT here.

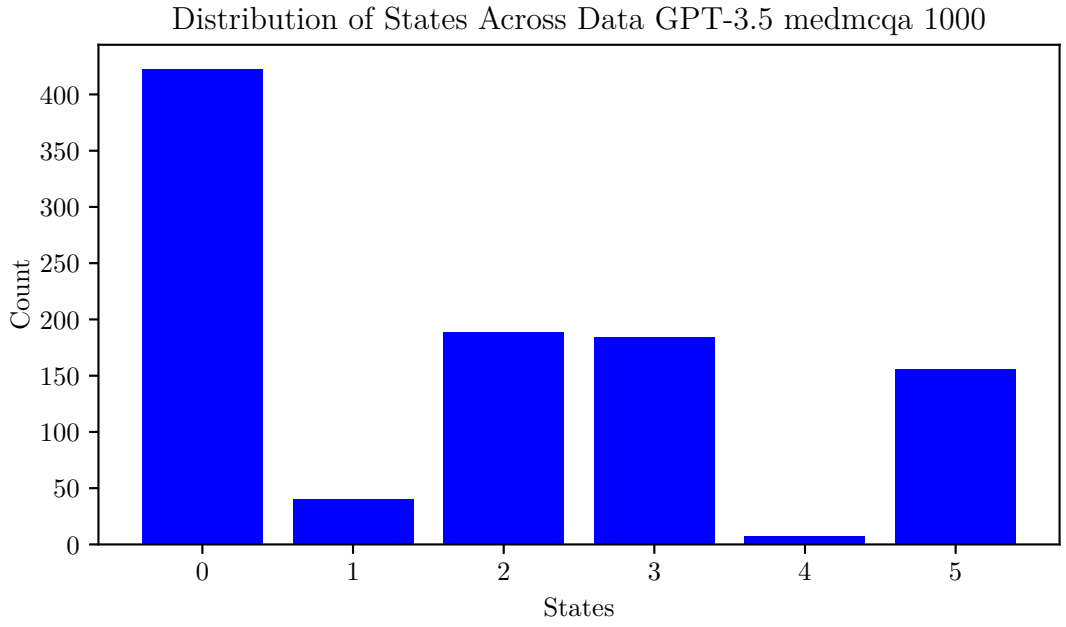
Platt Scaling significantly improves calibration for all models, resulting in substantially lower ECE scores compared to their uncalibrated versions. Beta Calibration further improves calibration, resulting in even lower ECE scores compared to Platt Scaling for all models. Isotonic Regression provides near-perfect calibration, resulting in ECE scores very close to zero for all models. This indicates highly accurate and well-calibrated probabilistic predictions across all models after applying Isotonic Regression.

Among the Calibration methods, Isotonic Regerssion provides the best calibration for all models.

The scientific question that CoT can achieve higher calibration than post-processing calibration methods is wrong based on the measures Brier Score and Expected Calibration Error, and, indeed the opposite seems to be true.



(a) We have more answers in state 3 than 2, meaning we have more answers where the model corrected itself correctly (state 3) than corrected itself incorrectly, meaning switching from a correct to an incorrect answer.



(b) We have more answers in state 2 than 3, thus declining the accuracy.

Figure 4.6: Comparison of state frequency overview

4.4.4 Calibration curve

In the following analysis, the calibration of the models is assessed using calibration curves.

The calibration curve for cohere (Figure 4.7) exhibits substantial deviation from the ideal diagonal line, indicating poor calibration. This suggests that the model's probabilistic predictions are not aligned with the actual outcome frequencies, reinforcing the hypothesis that the cohere model may be operating on a guessing mechanism.

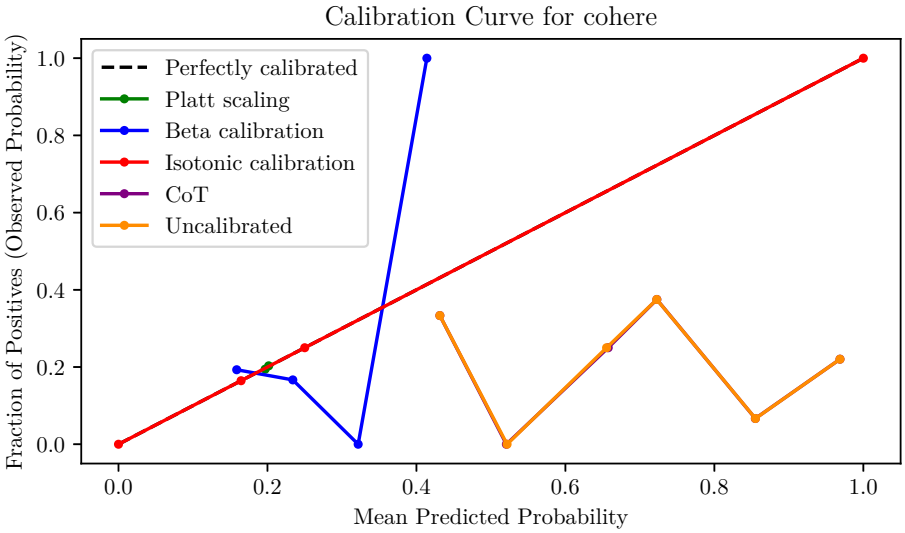


Figure 4.7: Calibration curve for the uncalibrated Coherence demonstrates suboptimal calibration. Isotonic Regression improves calibration immensely.

Contrasting with Cohere, GPT-3.5 demonstrates a notably superior calibration curve when considering the initial uncalibrated data (refer to Figure 4.9 for the Calibration Curve of 1000 sample questions and Figure 4.8 for 100 samples, including Chain-Of-Thought Calibration). The curve of the uncalibrated proabilities tracks the diagonal line, indicative of a positive correlation between predicted probabilities and actual outcomes. This alignment underscores the model's increasing confidence in its predictions corresponding to a higher likelihood of accuracy.

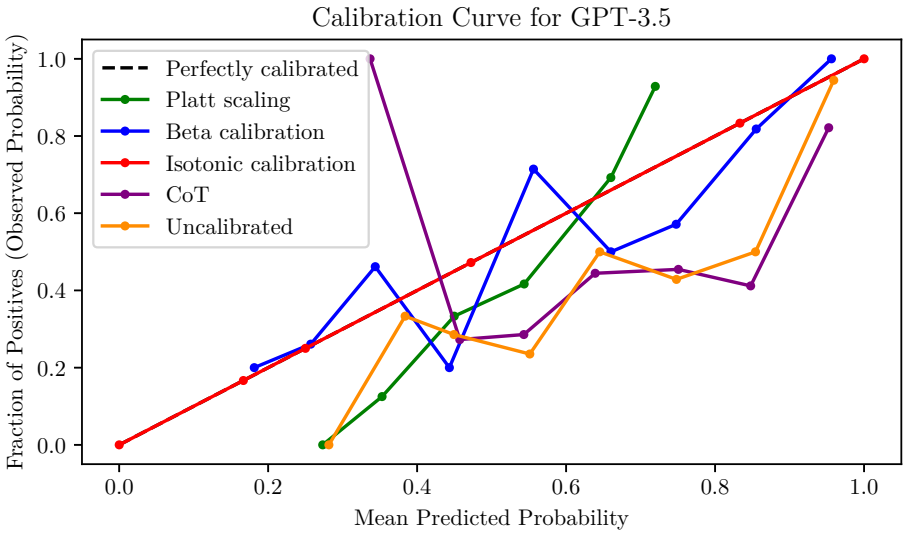


Figure 4.8: Calibration Curve for GPT-3.5 showing fine calibration for uncalibrated, improving after applying the Post-Processing Calibration methods, especially perfect calibration after applying Isotonic Regression.

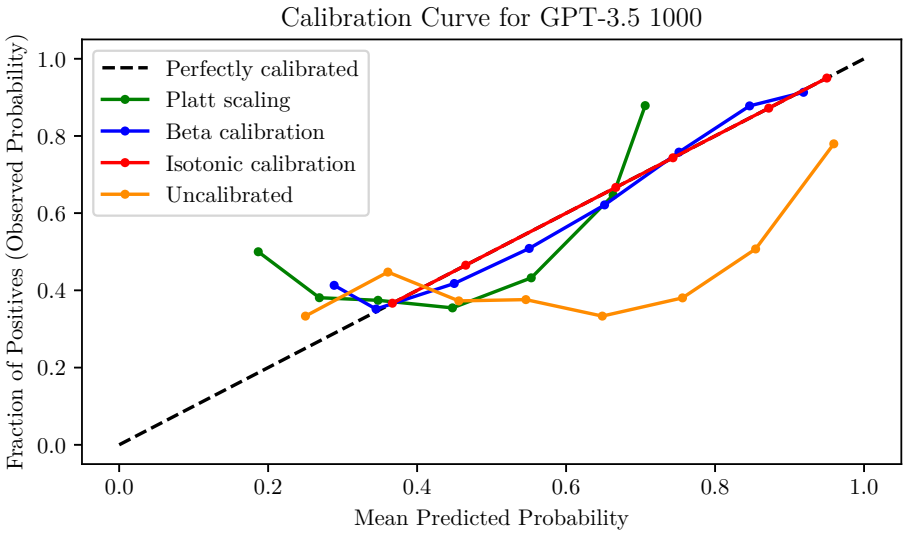


Figure 4.9: Calibration Curve for GPT-3.5 showing good calibration aligning with the ideal diagonal line after applying Post-Processing Calibration methods

In line with observations on Brier Score and Expected Calibration Error, once again, CoT does not significantly impact calibration. However, both Platt Scaling and Beta

Calibration visibly enhance calibration, evidenced by a closer alignment with the diagonal line representing perfect calibration. Notably, Isotonic Calibration precisely matches the line of perfect calibration, suggesting it as the optimal method for refining post-processing calibration for this model.

4.4.5 ROC Curve

The ROC curve depicted in Figure 4.10 reveals a notable deficiency in discriminatory capability for Cohere, as evidenced by its AUC score of 0.55. This outcome strongly suggests that the model's predictions closely resemble random guesses. Notably, the curve closely mirrors the diagonal line, a hallmark of random guessing. Although a slight enhancement is observed upon incorporating Isotonic Regression, the overwhelming evidence pointing to mere guesswork renders the addition of further Post-Processing Calibration methods unnecessary for Cohere.

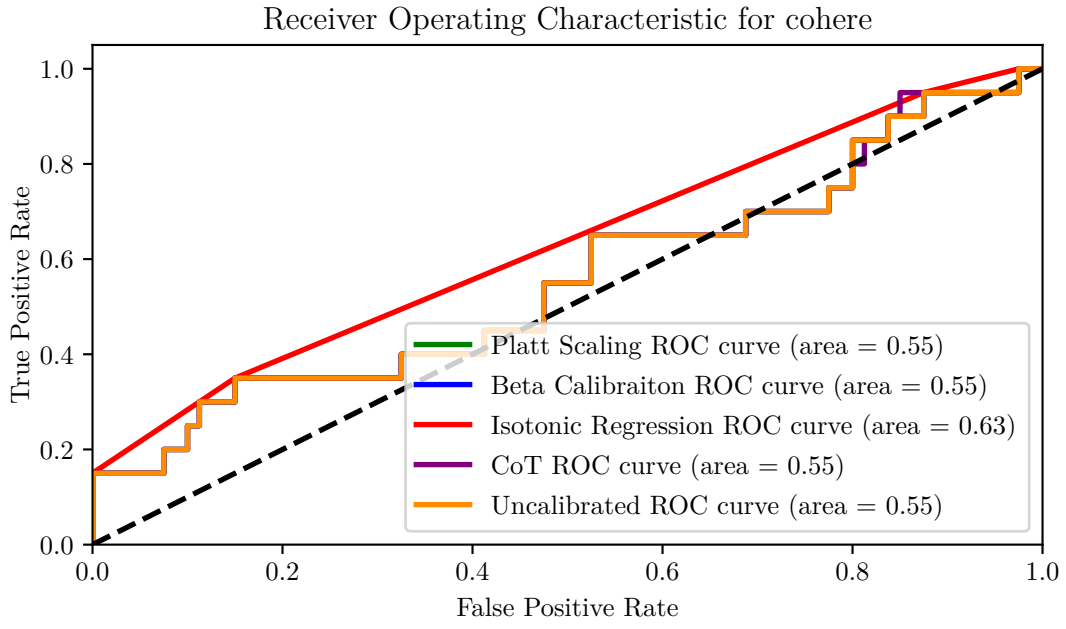


Figure 4.10: ROC for cohere indicating random guessing.

In contrast, the GPT-3.5 model's ROC curves (see Figure 4.9 for the Calibration Curve of 1000 sample questions and Figure 4.8 for 100 samples, including Chain-Of-Thought Calibration) showcases an improvement over Cohere's. Although not flawless, the curve demonstrates a tendency towards the "hugging" direction, implying a more optimal balance between the model's true positive rate and false positive rate. The term "hugging" refers to the curve's proximity to the upper-left corner of the ROC space, where the true positive rate is high and the false positive rate is low. This alignment indicates a more desirable performance of the model in distinguishing between positive and negative cases. Additionally, with an AUC score of 0.72, the model's performance is deemed acceptable.

Post-processing calibration method further enhance the AUC score, with Isotonic Regression yielding a significant improvement. However, Platt Scaling and Beta Calibration did not result in any noticeable enhancements. Notably, there is no discernible calibration

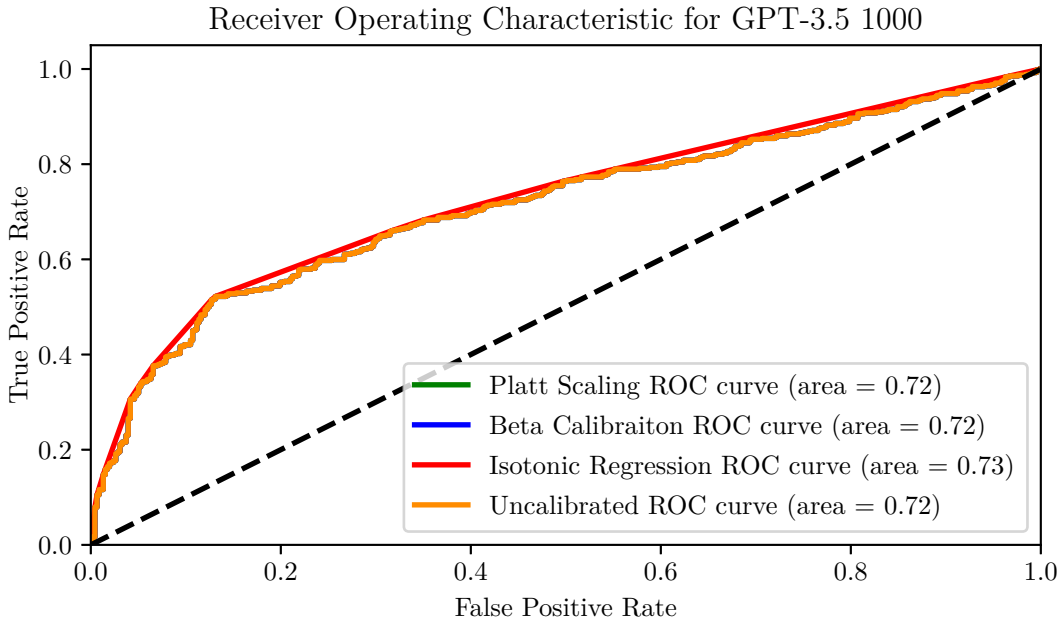


Figure 4.11: ROC curve for GPT-3.5 with 1000 samples demonstrates an acceptable level of calibration, albeit not achieving perfection. Notably, both Platt Scaling and Beta Calibration fail to induce any detectable alterations to the uncalibrated ROC curve, rendering them indistinguishable within the plot.

enhancement observed with Chain-Of-Thought Prompting, as the AUC Score is almost identically to the calibration before CoT, even slightly below, with 0.71 (in Contrast to 0.72).

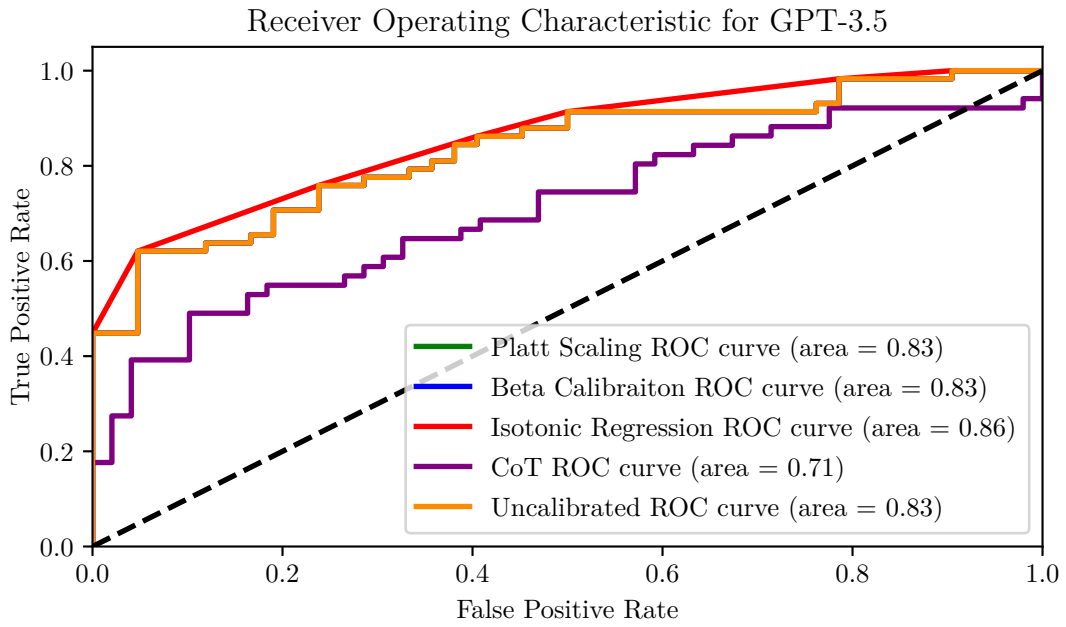


Figure 4.12: ROC curve for GPT-3.5 displays a passable level of calibration, which gets better with Isotonic Regression. Interestingly, both Platt Scaling and Beta Calibration show again no noticeable improvement over the uncalibrated curve. Moreover, the CoT Curve performs even worse than the uncalibrated ROC curve.

CHAPTER 5

Discussion

This thesis focused on evaluating how well Large Language Models, specifically GPT-3.5 and Cohere, can accurately answer medical multiple-choice questions. It looked into how different prompting techniques and adjustments after the initial processing might improve the models' performance, in both correctness and confidence in its responses. The research used the MedMCQA and MedMC dataset for testing. The results provide a clear view of what these advanced AI models can and cannot do when it comes to understanding medical information, highlighting their capabilities and limitations.

5.1 Model Performance and Prompting Strategies

The comparison between GPT-3.5 and Cohere showed a noticeable difference in their performance on answering medical multiple-choice questions, with GPT-3.5 achieving a significantly higher accuracy rate than Cohere. This difference might be due to several factors, including the amount and type of training data each model was exposed to, their architectural differences, and the technologies they're built on. It's possible that GPT-3.5 had access to a broader range of medical content during training, which might explain its superior performance.

Despite expectations, using Chain-of-Thought prompting strategies did not lead to a significant improvement in the accuracy of the models. It was anticipated that CoT prompts would help the models adopt a more systematic approach to solving problems, but the results showed that these prompts had little to no effect on the accuracy of the responses. This suggests that the effectiveness of CoT prompting might be limited to specific contexts or types of questions. In the case of medical multiple-choice questions, which often require direct factual knowledge rather than complex reasoning, CoT prompts seem to offer little benefit.

5.2 Confidence and Calibration

The analysis of model confidence showed that there is a link between how confident GPT-3.5 is in its answers and how often those answers are correct. This suggests that the model's own estimates of how likely it is to be correct could be a useful tool for determining the reliability of its medical advice.

However, when looking at how well the models' confidence matched their actual performance, both GPT-3.5 and especially Cohere showed significant discrepancies. This misalignment means that the models' confidence levels might not always reflect their true likelihood of being correct. It indicates that if GPT-3.5 is very sure about its answer, above a threshold of about 90%, it truly reflects that it knows the question, as in the context that the model was trained on the domain and question.

5.3 Revision Strategy Efficacy

The accuracy of GPT-3.5's responses improved slightly after revising the answers, but the overall impact was not significant. This indicates that while strategies to revise answers have the potential to correct inaccuracies, their effectiveness may be hampered by the model's initial comprehension of the question and its ability to generate improved responses on further consideration.

Additionally, it was observed that if the model is prompted with a suggestion that its first response may be incorrect, it indeed tends to alter its answer. This behavior was noted when all questions were subjected to revision, highlighting the necessity of establishing a threshold for when to apply revisions, to not impair the overall correctness of the results.

5.4 Post-Processing Calibration and Comparison

The assessment of Brier scores reveals the extent of calibration achieved by each method. Lower Brier scores indicate better calibration, with Isotonic Regression consistently outperforming Platt Scaling and Beta Calibration across all models. This highlights the efficacy of Isotonic Regression in improving the alignment between predicted probabilities and actual outcomes, leading to more accurate and reliable predictions.

Similarly, the analysis of ECE scores reaffirms the superior performance of Isotonic Regression in achieving near-perfect calibration. The negligible ECE scores obtained after applying Isotonic Regression indicate minimal discrepancies between predicted probabilities and observed frequencies, signifying highly calibrated models.

The Calibration curves offered a visual representation of model calibration, with the ideal calibration depicted by a diagonal line. Discrepancies from this line indicate deviations in calibration, with Cohere exhibiting substantial deviations, particularly at certain probability thresholds. In contrast, GPT-3.5 models demonstrate superior

calibration, especially after post-processing calibration methods such as Platt Scaling, Beta Calibration, and Isotonic Regression. These methods effectively reduce deviations from the diagonal line, indicating improved calibration.

ROC curves provide insights into the discriminatory capability of models, with higher AUC scores indicating better performance. Cohere's ROC curve reflects poor discriminatory capability, resembling random guessing, whereas GPT-3.5 models exhibit moderate discrimination, with room for improvement. Post-processing calibration methods enhance discrimination, with Isotonic Regression yielding the most significant improvements.

Notably, Chain-Of-Thought Prompting does not significantly impact discrimination, suggesting its limited effectiveness in enhancing model performance in this context.

5.5 Research questions

5.5.1 How can post-processing calibration contribute to rendering LLMs more reliable and trustworthy for applications in clinical settings?

By calibrating LLM predictions, post-processing techniques such as Platt Scaling, Beta Calibration, and Isotonic Regression help align predicted probabilities with actual outcomes, thereby improving the model's calibration and accuracy. In clinical applications, where decisions are often based on probabilistic predictions, well-calibrated models instill confidence among practitioners and patients alike. Additionally, calibrated LLMs are better equipped to provide accurate risk assessments, aid in treatment planning, and support clinical decision-making processes. By leveraging post-processing calibration, LLMs can contribute to more reliable diagnostic outcomes, but there is still a lot of work to do until they ultimately can enhance patient care and safety in clinical settings.

5.5.2 To what extent do various post-processing calibration techniques exhibit comparative advantages and limitations in the calibration of LLMs for diagnostic purposes?

Post-processing calibration has distinct advantages and limitations in calibrating LLM predictions for diagnostic purposes. Platt Scaling is a simple yet effective method that improves calibration by fitting a logistic regression model to LLM scores. While it provides noticeable calibration improvements, it may struggle with extreme probabilities and may not fully capture non-linear relationships between LLM scores and true probabilities. Beta Calibration, on the other hand, addresses these limitations by modeling the calibration curve using a non-parametric approach, resulting in more flexible calibration. However, it may require larger datasets for optimal performance and can be computationally intensive. Isotonic Regression stands out as a highly effective calibration technique, offering near-perfect calibration by modeling the calibration curve as a piecewise constant function. It is robust to outliers and can handle small datasets effectively. However, it may suffer from overfitting if not appropriately regularized.

5.5.3 Exploring the potential of Chain-of-Thought prompting: Can a more finely tuned calibration be achieved compared to traditional post-processing calibration methods? What is the procedural approach to achieving this calibration with the right prompting strategies?

Chain-of-Thought prompting represents an innovative approach to prompting LLMs by guiding the generation process through structured prompts. While Chain-of-Thought prompting may excel in facilitating coherent and contextually relevant responses for logic and procedural questions, its effectiveness in improving calibration and accuracy for LLMs when answering questions based on unfamiliar knowledge is limited. Unlike traditional post-processing calibration methods such as Platt Scaling, Beta Calibration, and Isotonic Regression, which focus on adjusting predicted probabilities to align with ground truth outcomes, Chain-of-Thought prompting does not inherently address calibration issues.

The procedural approach to employing Chain-of-Thought prompting involves designing prompts tailored to guide the LLM through complex reasoning tasks, such as solving intricate calculations by breaking them down into manageable steps. However, when applied to questions requiring knowledge outside the LLM's training data, Chain-of-Thought prompting may not achieve improved calibration or accuracy. In such cases, the LLM's responses may lack grounding in factual information, leading to unreliable predictions.

While Chain-of-Thought prompting holds promise for certain types of tasks, its utility for improving calibration and accuracy in diagnostic settings where knowledge outside the training data is frequently required is limited.

CHAPTER 6

Conclusion

In this thesis, I investigated the capabilities of Large Language Models, specifically GPT-3.5 and Cohere, in answering medical multiple-choice questions and explored various strategies to improve their performance, confidence estimation, and calibration. Through experimentation with the MedMCQA and MedMC dataset, insights were gained into the strengths and limitations of these advanced AI models in comprehending and responding to medical inquiries.

The findings revealed significant differences in the performance of GPT-3.5 and Cohere, with GPT-3.5 demonstrating superior accuracy in answering medical MCQs. We attributed this variance to factors such as the models' training data diversity, architectural variances, and underlying technologies. Despite expectations, employing Chain-of-Thought prompting strategies did not substantially enhance model accuracy, indicating the contextual limitations of this approach in medical question-answering tasks.

Also the relationship between model confidence and correctness of responses was examined, noting a correlation between GPT-3.5's confidence and answer accuracy. However, discrepancies between predicted confidence and actual performance highlighted the need for robust calibration methods to ensure the reliability of model predictions in clinical applications.

Furthermore, my investigation into revision strategies underscored their potential to marginally improve response accuracy, albeit with minimal impact overall. Notably, prompting models with suggestions of potential inaccuracies prompted them to reconsider and revise their initial responses, suggesting the importance of judiciously applying revision strategies to optimize accuracy does not always work. Here it is important to notice, that those revision strategy also change correct responses to incorrect ones.

Post-processing calibration techniques, including Platt Scaling, Beta Calibration, and Isotonic Regression, emerged as pivotal tools for aligning model predictions with ground

truth outcomes. Isotonic Regression, in particular, exhibited remarkable effectiveness in achieving near-perfect calibration, thereby enhancing the reliability and trustworthiness of LLMs for clinical applications.

In conclusion, our study sheds light on the nuanced interplay between LLM performance, prompting strategies, confidence estimation, revision techniques, and calibration methods in the context of medical question-answering. While LLMs like GPT-3.5 show promise in understanding medical content, their optimal utilization in clinical settings necessitates careful consideration of contextual factors, rigorous calibration, and ongoing refinement of strategies to ensure accuracy, reliability, and most, learning and training of more medical context. Moving forward, further research and development efforts are warranted to enhance LLM capabilities and maximize their potential contributions to healthcare delivery and decision-making processes.

List of Figures

List of Tables

Bibliography

[Coh23] Cohere. Cohere models documentation. `https://docs.cohere.com/docs/models`, 2023. Accessed: 26/11/2023.

[CPH23] Alexandre Capone, Geoff Pleiss, and Sandra Hirche. Sharp calibrated gaussian processes, 2023.

[Fer22] Luciana Ferrer. Analysis and comparison of classification metrics. *arXiv preprint arXiv:2209.05355*, 2022.

[FFR05] Ronen Fluss, David Faraggi, and Benjamin Reiser. Estimation of the youden index and its associated cutoff point. *Biometrical Journal: Journal of Mathematical Methods in Biosciences*, 47(4):458–472, 2005.

[HLM+20] Yingxiang Huang, Wentao Li, Fima Macheret, Rodney A Gabriel, and Lucila Ohno-Machado. A tutorial on calibration measurements and calibration models for clinical prediction models. *Journal of the American Medical Informatics Association*, 27(4):621–633, 2020.

[JPO+20] Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? A large-scale open domain question answering dataset from medical exams. *CoRR*, abs/2009.13081, 2020.

[KSFF17] Meelis Kull, Telmo Silva Filho, and Peter Flach. Beta calibration: a well-founded and easily implemented improvement on logistic calibration for binary classifiers. In *Artificial intelligence and statistics*, pages 623–631. PMLR, 2017.

[NC07] Todd G Nick and Kathleen M Campbell. Logistic regression. *Methods in molecular biology (Clifton, NJ)*, 404:273–301, 2007.

[OA24] Kurez Oroy and Jane Anderson. Robustness of large language models: Mitigating adversarial attacks and input perturbations. Technical report, EasyChair, 2024.

[Ope23] OpenAI. Gpt-3.5 documentation. `https://platform.openai.com/docs/models/gpt-3-5`, 2023. Accessed: 26/11/2023.

[Pla00] John Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Adv. Large Margin Classif.*, 10, 06 2000.

[PLSL17] Hao Peng, Jianxin Li, Yangqiu Song, and Yaopeng Liu. Incrementally learning the hierarchical softmax function for neural language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

[PUS22] Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In Gerardo Flores, George H Chen, Tom Pollard, Joyce C Ho, and Tristan Naumann, editors, *Proceedings of the Conference on Health, Inference, and Learning*, volume 174 of *Proceedings of Machine Learning Research*, pages 248–260. PMLR, 07–08 Apr 2022.

[REC20] Nicolás Ricardo Enciso and Jorge E. Camargo. Classification and visualization of web attacks using http headers and machine learning techniques. In Fabián R. Narváez, Diego F. Vallejo, Paulina A. Morillo, and Julio R. Proaño, editors, *Smart Technologies, Systems and Applications*, pages 243–255, Cham, 2020. Springer International Publishing.

[TTE+23] Arun James Thirunavukarasu, Darren Shu Jeng Ting, Kabilan Elangovan, Laura Gutierrez, Ting Fang Tan, and Daniel Shu Wei Ting. Large language models in medicine. *Nature medicine*, 29(8):1930–1940, 2023.

[WWS+22] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed H. Chi, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. *CoRR*, abs/2201.11903, 2022.

[ZE02] Bianca Zadrozny and Charles Elkan. Transforming classifier scores into accurate multiclass probability estimates. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 08 2002.

[ZXG+24] Qinyu Zhao, Ming Xu, Kartik Gupta, Akshay Asthana, Liang Zheng, and Stephen Gould. The first to know: How token distributions reveal hidden knowledge in large vision-language models? *arXiv preprint arXiv:2403.09037*, 2024.