

1 A Appendix / supplemental material

2 A.1 Prompt Examples

3 Following the Fast-Chat [1], the prompt used in the data selection process is as Table 1.

Table 1: Prompt of data selection for LLM

System Prompt: Please act as an impartial judge and evaluate the quality of the responses provided by three AI assistants to the user question displayed below. You should choose the assistant that follows the user’s instructions and answers the user’s question best. Your evaluation should consider factors such as the helpfulness, relevance, accuracy, depth, creativity, and level of detail of their responses. Begin your evaluation by comparing the two responses and provide a short explanation. Avoid any position biases and ensure that the order in which the responses were presented does not influence your decision. Do not allow the length of the responses to influence your evaluation. Do not favor certain names of the assistants. Be as objective as possible. After providing your explanation, output your final verdict by strictly following this format: “[A]” if assistant A is the bset, “[B]” if assistant B is the bset, and “[C]” for a tie.
Prompt Template: [User Question] <question> [The Start of Assistant A’s Answer] <answer_a> [The End of Assistant A’s Answer] [The Start of Assistant B’s Answer] <answer_b> [The End of Assistant B’s Answer] [Final Verdict]:

4 A.2 Case Study

5 The case of the single-turn dialogue and multi-turn dialogue are presented in Table 2 and Table 3.

Table 2: A comparison case on Single turn

Instruction: A tech startup invests \$8000 in software development in the first year, and then invests half of that amount in software development in the second year. What’s the total amount the startup invested in software development over the two years?	
Llama-2-7B-evol_instruct: The total amount the startup invested in software development over the two years is \$12,000.	Llama-2-7B-star_instruct: To calculate the total amount the startup invested in software development over the two years, we need to find the total investment in software development for each year and then add them together. In the first year, the startup invested \$8000 in software development. In the second year, the startup invested half of that amount, which is \$4000. So, the total investment in software development for the two years is: \$8000 (first year) + \$4000 (second year) = \$12000 Therefore, the total amount the startup invested in software development over the two years is \$12000.

Table 3: A comparison case on Multi turn

<p>Instruction: Act as a math teacher. I will provide some mathematical equations or concepts, and it will be your job to explain them in easy-to-understand terms. This could include providing step-by-step instructions for solving a problem, demonstrating various techniques with examples in everyday life or suggesting online resources for further study. My first request is "I need help understanding how probability works."</p>	
<p>Llama-2-7B-evol_instruct: Sure, I'd be happy to help! Probability is the likelihood of something happening, and it's a fundamental concept in math and science. To explain it simply, probability is the chance of something happening out of all possible outcomes. For example, if you flip a coin, the probability of getting heads is 1 out of 2, or 50%. Probability can be used to make predictions and calculate risks, and it's important in fields like finance, statistics, and data analysis.</p> <p>To calculate probability, you can use different formulas depending on the type of problem you're solving. One common method is to use a formula called the probability mass function, which tells you the likelihood of getting a certain outcome. Another method is to use the probability density function, which tells you the likelihood of getting a certain range of outcomes.</p> <p>There are many resources available online to help you learn more about probability, including video tutorials, interactive simulations, and practice problems. Some popular websites for learning probability include Khan Academy, Mathway, and Coursera. I hope this helps!</p>	<p>Llama-2-7B-star_instruct: Sure, I'd be happy to help explain probability! Probability is a measure of how likely an event is to occur. It's often expressed as a number between 0 and 1, where 0 represents a very unlikely event and 1 represents a very likely event. To calculate probability, you need to know the total number of possible outcomes and the number of favorable outcomes for the event you're interested in. For example, if you're flipping a fair coin, there are two possible outcomes (heads or tails), and the probability of getting heads is $1/2$ or 0.5, because there's only one way to get heads out of two possible outcomes. In everyday life, probability can be used to calculate the likelihood of certain events, such as the chance of rain on a particular day or the probability of getting a certain grade on a test. It's also used in decision-making situations, like calculating the risk of a particular investment or the likelihood of success in a business venture. There are many resources available online to help you learn more about probability, such as video tutorials, practice problems, and interactive simulations. Some popular websites include Khan Academy, Coursera, and edX, which offer free courses and resources on probability and other mathematical concepts.</p>
<p>Instruction: What are the differences between Riemannian geometry and euclidean geometry?</p>	
<p>Llama-2-7B-evol_instruct: Riemannian geometry and Euclidean geometry are two different types of geometries. In Euclidean geometry, the distance between two points is always a straight line, and the angles of a triangle always add up to 180 degrees. In Riemannian geometry, the distance between two points is not always a straight line, and the angles of a triangle can add up to more than 180 degrees.</p> <p>Another key difference between the two geometries is the way they handle curved surfaces. In Euclidean geometry, a flat surface is a rectangle with parallel sides, while in Riemannian geometry, a surface can be curved in any way. This means that Riemannian geometry is better suited for describing the geometry of the universe, as it can handle the curvature of space-time.</p> <p>Overall, Riemannian geometry is a more complex and advanced form of geometry, and it has many practical applications in fields such as physics and engineering.</p>	<p>Llama-2-7B-star_instruct: Great question! Riemannian geometry and Euclidean geometry are two different types of geometry that are used to describe and analyze the properties of spaces. The main differences between the two are as follows: 1. Metric: In Euclidean geometry, the distance between two points is defined by the Pythagorean theorem, which states that the sum of the squares of the two sides of a right triangle is equal to the square of the length of the hypotenuse. In Riemannian geometry, the distance between two points is defined by a metric, which is a function that describes how much the shortest path between two points is curved. 2. Curvature: Euclidean geometry deals with flat spaces where the curvature of the Earth's surface is negligible. In contrast, Riemannian geometry deals with curved spaces, such as the surface of the Earth, where the curvature plays a significant role in determining the properties of the space. 3. Geodesics: In Euclidean geometry, the shortest path between two points is a straight line. In Riemannian geometry, the shortest path between two points is called a geodesic, which can be curved and depends on the metric. 4. Tensor Fields: Riemannian geometry involves the use of tensor fields, which are mathematical objects that describe the geometry of a space. These fields are used to calculate the curvature of the space and the behavior of objects moving through it. Overall, Riemannian geometry and Euclidean geometry are two different approaches to understanding the properties of spaces, with Riemannian geometry being more applicable to curved spaces and Euclidean geometry being more applicable to flat spaces.</p>

6 A.3 Data Examples

7 The example of diverse data is shown in Table 4.

Table 4: A comparison on Diversity Data Examples

<p>Instruction: Given some binary classification data, identify a suitable machine learning model that can be applied to this task.</p>
<p>Evol Instruct: The suitable machine learning models for binary classification tasks include:</p> <ol style="list-style-type: none"> 1) Logistic regression: It is a statistical method that uses a logistic function to model a binary dependent variable. It is a relatively simple and interpretable model that works well with linearly separable datasets. 2) Support Vector Machines (SVMs): It is a popular classifier that is used for binary classification tasks. SVMs can handle high-dimensional data and work well with datasets that are not linearly separable. 3) Random Forests: It is an ensemble learning method that works by constructing a multitude of decision trees at training time and outputs the class that is the mode of the classes output by the individual trees. 4) Neural networks: It is a type of machine learning model that is widely used for binary classification tasks. Neural networks can handle non-linear relationships between variables and are capable of learning complex patterns in the data. choice of the model depends on the particular characteristics of the data and the specific requirements of the project."
<p>Agent-Pair A: There are several machine learning models that can be applied to binary classification tasks, depending on the nature of the data and the problem at hand. Here are a few popular ones:</p> <ol style="list-style-type: none"> 1. Logistic Regression: This is a simple and effective model that can be used for binary classification problems. It works by estimating the probability of the positive class and using it to make predictions. 2. Decision Trees: Decision Trees are a popular machine learning model that can handle both categorical and numerical features. They work by recursively splitting the data into subsets based on the values of the input features. 3. Random Forest: Random Forest is an ensemble learning method that combines multiple decision trees to improve the accuracy of the model. It works by creating a set of decision trees and then averaging their predictions. 4. Support Vector Machines (SVM): SVM is a powerful model that can handle high-dimensional data. It works by finding the hyperplane that maximally separates the classes in the feature space. 5. Neural Networks: Neural Networks are a set of algorithms that can learn and classify from data. They can handle complex and nonlinear relationships between the input features and the output labels. <p>The choice of the model depends on various factors such as the size and complexity of the dataset, the type of features, the computing resources, and the desired level of accuracy.</p>
<p>Agent-Pair B: For binary classification tasks, several machine learning models are suitable, depending on the size of the dataset, complexity of the problem, and the availability of features. Here are some popular models that can be applied:</p> <ol style="list-style-type: none"> 1. Logistic Regression: A simple and interpretable model that works well for small to medium-sized datasets. It's linear and can handle both categorical and continuous features. 2. Decision Trees and Random Forest: These are easy to interpret and can handle both numerical and categorical data. Random Forest is an ensemble method that combines multiple decision trees, reducing overfitting. 3. Support Vector Machines (SVM): Effective for high-dimensional data and can handle non-linear relationships through kernel functions. SVMs are robust to outliers and can be used for both linear and non-linear classification. 4. Naive Bayes: A probabilistic model that assumes features are independent. It's fast, easy to implement, and works well for text classification or when features are not highly correlated. 5. K-Nearest Neighbors (KNN): A simple model that classifies based on the majority class of its nearest neighbors. It's non-parametric and can be useful for small datasets. 6. Neural Networks (e.g., Feedforward, Convolutional, or Recurrent Neural Networks): Deep learning models that can capture complex patterns in large datasets. They are powerful but require more data and computational resources. 7. Gradient Boosting Machines (GBM) or XGBoost: Ensemble methods that iteratively build weak learners (e.g., decision trees) to create a strong classifier. They are often used for structured data and can handle high dimensionality. 8. AdaBoost: Another boosting algorithm that combines weak learners to create a strong classifier. It's particularly useful when dealing with imbalanced datasets. 9. Neural Network Ensembles (e.g., stacking, blending): Combining predictions from multiple models can improve performance and robustness. choose the best model, it's essential to perform exploratory data analysis, split the data into training and testing sets, and evaluate the models using appropriate metrics (e.g., accuracy, precision, recall, F1-score, or AUC-ROC for imbalanced datasets). Cross-validation can help in selecting the optimal model and hyperparameters
<p>Agent-Pair C: A suitable machine learning model for this task would be a Random Forest Classifier. This model is effective in handling both categorical and numerical data, and can handle large datasets with high dimensionality.</p>
<p>Agent-Pair D: One suitable machine learning model for binary classification tasks is the Logistic Regression model. Logistic Regression is commonly used for binary classification as it provides the probability of an instance belonging to a particular class. It is a simple and efficient algorithm that works well with linearly separable data. popular machine learning models for binary classification tasks include Support Vector Machines (SVM), Decision Trees, Random Forest, Gradient Boosting Machines (GBM), and Neural Networks. The choice of the model depends on the complexity of the data, size of the dataset, interpretability requirements, and computational resources available.</p>

8 **References**

- 9 [1] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin,
10 Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena.
11 Advances in Neural Information Processing Systems, 36, 2024.