# Blind Evaluation for Error Explanations

Evaluator Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Instructions:**

**\*\*Firstly, make a copy of this file and edit that. This ensure that other evaluators cannot see your scores. \*\***

There are 10 errors below.  
For each error please score the 6 options based on how well they explain the corresponding error.

Each option has the “Score” field at the end of it. A score of 6 is the highest (or best) and score of 1 is the lowest (or the worst). Please enter your score to the right of the “Score” field.

Imagine that you are faced with this error and are trying to understand it. And then ask these questions to yourself.

1. Which option clearly explains the error first?
2. Which option would be easiest and quickest to understand?
3. Does the option provide any hints or potential solutions that can be quickly tried?

Carefully score the options. **Do not hurry. Please Take your time and score them.**

Do not cheat or discuss with other evaluators.

And finally, thank you for participating in this evaluation. It would help our research in many ways.

All the best.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

## Error Message 1

**ValueError You are trying to load a weight file containing 12 layers into a model with 11 layers**

Option 1: 12 layers are loaded into the model, but the model only has 11 layers, indicating that the model is missing a layer. Score:

Option 2: The error occurs because the model you are trying to load has 12 layers, but the model you are trying to load the weights into only has 11 layers. This is because the model you are trying to load has 12 layers, but the model you are trying to load the weights into only has 11 layers Score:

Option 3: The TypeError occurs because an attempt is made to iterate over an integer value, which is not supported Score:

Option 4: 12 layers in the weight file do not match the 11 layers in the model. Score:

Option 5: The `ValueError` occurs because the number of layers in the weight file does not match the number of layers in the model; ensure both the model architecture and the weight file correspond in layer count before loading. Score:

Option 6: The `ValueError` indicates that the number of layers in the model does not match the number of layers in the weight file; ensure that the model and weight file are compatible. Score:

## Error Message 2

**‘s’ was not declared in this scope**

Option 1: This error occurs because the variable `s` is not declared in the scope of the function where it is used. Ensure that the variable is declared in the correct scope. Score:

Option 2: The error occurs because the variable 's' is not defined in the scope of the function; to resolve this, ensure the variable is defined in the scope of the function. Score:

Option 3: This error indicates that the variable s is being used without being declared within the accessible scope; ensure all variables are declared before use. Score:

Option 4: s’ was not declared in this scope  
 This error occurs when you try to use a variable that has not been declared.  
 To fix this, you can declare the variable before using it. Score:

Option 5: The error occurs because the variable ‘s’ is not declared in the scope of the function, indicating that the variable is not defined in the function. Score:

Option 6: The error message indicates that a variable named ‘s’ is being used in a scope where it has not been declared or initialized, suggesting that the variable is being used before it is defined or initialized. Score:

## Error Message 3

**AttributeError: 'functools.partial' object has no attribute '\_\_code\_\_'**

Option 1: The error occurs because the script named 'functools.py' is shadowing the built-in Python 'functools' module; renaming your script file to a different name should resolve the issue. Score:

Option 2: The AttributeError occurs because the script attempts to access the `\_\_code\_\_` attribute of a 'functools.partial' object, which does not exist; check for any modifications being made to `\_\_code\_\_` of 'functools.partial' objects and ensure such operations are valid and necessary.  
  
## Score:

Option 3: The error occurs because the code attempts to access the `\_\_code\_\_` attribute of a `functools.partial` object, which does not have this attribute and is not meant to be used as a function. Score:

Option 4: The error occurs because the shapes of two matrices being multiplied in a linear layer do not align, specifically, a matrix with shape (128x3072) cannot be multiplied with another matrix of shape (1024x256), indicating a mismatch in the expected dimensions for the operation. Score:

Option 5: The `AttributeError` happens because a `functools.partial` object lacks a `\_\_code\_\_` attribute, which is necessary for operations that require inspecting the function's code, such as serialization or some forms of introspection; use the original function object instead for such operations. Score:

Option 6: The 'functools.partial' object has no attribute '\_\_code\_\_'.  
 This error occurs when you try to access the \_\_code\_\_ attribute of a partial object.  
 The \_\_code\_\_ attribute is used to get the code object of a function.  
 The \_\_code\_\_ attribute is used to get the code Score:

## Error Message 4

**TypeError BatchDataset' object is not subscriptable**

Option 1: The error message indicates that the object is not subscriptable. This means that the object does not support indexing.  
   
 The most common cause of this error is when you try to index a list or a dictionary.  
   
 The solution is to check the documentation to see if the object supports indexing. Score:

Option 2: This error occurs because the `BatchDataset` class is not subscriptable, which means that it cannot be indexed. This is because the `BatchDataset` class is not designed to be indexed, and instead should be iterated over. Score:

Option 3: The `TypeError` indicates that the `BatchDataset` class does not support subscripting, typically resolved by ensuring that the class definition includes this argument or correcting the argument name if it was misspelled. Score:

Option 4: The `TypeError` indicates that a 'BatchDataset' object does not support indexing directly, which is common for dataset objects in machine learning frameworks; instead, iterate through the dataset or use dataset-specific methods to access elements. Score:

Option 5: The error occurs because the shapes of two matrices being multiplied in a linear layer do not align, specifically, a matrix with shape (128x3072) cannot be multiplied with another matrix of shape (1024x256), indicating a mismatch in the expected dimensions for the operation. Score:

Option 6: 1. The error occurs because the code attempts to access a batch of data using subscript notation, but the dataset object is not iterable or subscriptable  
 2. The solution is to ensure that the dataset object is correctly loaded and formatted for use in the code  
</s> Score:

## Error Message 5

**TypeError can only concatenate str (not "datetime.datetime") to str**

Option 1: The error "TypeError can only concatenate str (not "datetime.datetime") to str" occurs because the `+` operator is being used to concatenate a string with a `datetime` object; use the `strftime` method to convert the `datetime` object to a string before concatenating. Score:

Option 2: This error arises when attempting to concatenate a string with a datetime object directly in Python; convert the datetime object to a string using a suitable format method before concatenating. Score:

Option 3: The error occurs because the shapes of two matrices being multiplied in a linear layer do not align, specifically, a matrix with shape (128x3072) cannot be multiplied with another matrix of shape (1024x256), indicating a mismatch in the expected dimensions for the operation. Score:

Option 4: The TypeError exception is raised when an operation or function receives an argument of an unsupported type. Score:

Option 5: This error occurs because the script is attempting to concatenate a string with a datetime object, which is not supported. Score:

Option 6: 1. The TypeError occurs because Python expects a string to be concatenated with another string, but it received a datetime object instead, which cannot be concatenated with strings.  
 2. To fix the TypeError, ensure that the object being concatenated with the string is a string as well.  
</s> Score:

## Error Message 6

**control reaches end of non-void function**

Option 1: The error occurs when a function is defined with a return type but does not return a value at the end of its execution. Score:

Option 2: The function does not return a value. Score:

Option 3: This error occurs because the function does not return a value, which is required for all non-void functions. Score:

Option 4: This error means that a function expected to return a value does not have a return statement on all control paths, which is required to provide an output for each possible execution path. Score:

Option 5: This error occurs because the function is not returning a value, and the return type is not void. Score:

Option 6: The error occurs because the shapes of two matrices being multiplied in a linear layer do not align, specifically, a matrix with shape (128x3072) cannot be multiplied with another matrix of shape (1024x256), indicating a mismatch in the expected dimensions for the operation. Score:

## Error Message 7

**OSError Not found: "/weights/LLaMA/tokenizer.model": No such file or directory Error #2**

Option 1: The error occurs because the script named 're.py' is shadowing the built-in Python 're' (regular expression) module; renaming your script file to a different name should resolve the issue. Score:

Option 2: The `OSError` occurs because the file at "/weights/LLaMA/tokenizer.model" cannot be found, likely due to an incorrect path or the file not being present at the specified location; verify the file path and ensure the file exists in the directory you are accessing. Score:

Option 3: The error occurs because the script is trying to access a file that does not exist, likely due to a typo in the file path or a missing directory. Score:

Option 4: The error "OSError Not found: "/weights/LLaMA/tokenizer.model": No such file or directory Error #2" occurs when the system cannot find the file "tokenizer.model" in the specified directory,possibly because it is being used by another process Score:

Option 5: The file tokenizer.model was not found in the directory /weights/LLaMA. This file is required for the model to run. Score:

Option 6: This error occurs because the specified file '/weights/LLaMA/tokenizer.model' could not be found, indicating that the path to the file is incorrect or the file does not exist at the specified location. Score:

## Error Message 8

**ValueError Sample larger than population or is negative**

Option 1: The error occurs because the shapes of two matrices being multiplied in a linear layer do not align, specifically, a matrix with shape (128x3072) cannot be multiplied with another matrix of shape (1024x256), indicating a mismatch in the expected dimensions for the operation. Score:

Option 2: The error is caused by the sample being larger than the population or negative.  
 The solution is to make sure the sample is smaller than the population and is positive.  
 The solution is to make sure the sample is smaller than the population and is positive. Score:

Option 3: The `ValueError` occurs when attempting to sample more items than are available in the population or using a negative sample size, often with functions like `random.sample`; ensure the sample size is positive and does not exceed the population size. Score:

Option 4: The `ValueError` indicates that the sample size is larger than the population size or is negative; ensure that the sample size is not negative and is smaller than the population size. Score:

Option 5: The `ValueError` occurs when the size of the sample is larger than the population or is negative, indicating that the sample size is invalid. Score:

Option 6: This error occurs because the sample size is larger than the population size or is negative. Score:

## Error Message 9

**ValueError can only convert an array of size 1 to a Python scalar**

Option 1: This error happens when trying to convert an array with a size greater than 1 into a scalar, commonly caused by mistakenly treating an array as a scalar; to fix, ensure the operation is compatible with array inputs. Score:

Option 2: The error occurs because the shapes of two matrices being multiplied in a linear layer do not align, specifically, a matrix with shape (128x3072) cannot be multiplied with another matrix of shape (1024x256), indicating a mismatch in the expected dimensions for the operation. Score:

Option 3: ValueError can only convert an array of size 1 to a Python scalar Score:

Option 4: The error occurs because the `ValueError` is raised when attempting to convert an array of size greater than 1 to a Python scalar; ensure that the input is a valid scalar before attempting to convert it. Score:

Option 5: The error occurs because the script named 're.py' is shadowing the built-in Python 're' (regular expression) module; renaming your script file to a different name should resolve the issue. Score:

Option 6: The `ValueError` occurs when a NumPy array with more than one element is attempted to be converted to a Python scalar, which is not possible. Score:

## Error Message 10

**TypeError Layer.\_\_init\_\_() takes 1 positional argument but 2 were given**

Option 1: The `TypeError` occurs because the `\_\_init\_\_()` method of a `Layer` class was called with two arguments, but it only accepts one; ensure that you are passing the correct number of arguments as specified in the class definition or library documentation. Score:

Option 2: The `Layer.\_\_init\_\_()` method takes 1 positional argument but 2 were given. Score:

Option 3: 1. The TypeError occurs when a function is called with more arguments than expected, in this case, the \_\_init\_\_ method of the Layer class is expecting one argument but receives two.  
 2. The solution is to ensure that the number of arguments passed to the \_\_init\_\_ method matches the number of parameters defined in the class. Score:

Option 4: This error occurs because the constructor of `Layer` is called with more arguments than it is defined to accept; to resolve this, ensure that the correct number of arguments is passed when instantiating `Layer`. Score:

Option 5: The error occurs because the shapes of two matrices being multiplied in a linear layer do not align, specifically, a matrix with shape (128x3072) cannot be multiplied with another matrix of shape (1024x256), indicating a mismatch in the expected dimensions for the operation. Score:

Option 6: The `TypeError` occurs because the `Layer` constructor is used incorrectly with multiple arguments; to create a layer with multiple arguments, the arguments should be enclosed in parentheses as a single argument, like `Layer((arg1, arg2))`. Score: