CODE EXPLANATION **10/5/2024**

8.EVALUATION PART1

Here AI generated response

response = "[{'category': 'Computers and Laptops', 'products': ['TechPro Ultrabook', 'BlueWave Gaming Laptop']}, {'category': 'Gaming Consoles', 'products': ['GameSphere X']}]"

Since this string uses single quotes instead of double quotes (which are required for valid JSON) we need to replace single quotes with double quotes:

Next,

json_like_str = response.replace("'", '"')

Next,

Now we can parse this modified string into a Python data structure:

l_of_d = json.loads(json_like_str)

Resulting Parsed Output

After parsing, l_of_d will contain:

```
l_of_d = [
    {'category': 'Computers and Laptops', 'products': ['TechPro Ultrabook', 'BlueWave Gaming Laptop']},
    {'category': 'Gaming Consoles', 'products': ['GameSphere X']}
```

]

Next,

Now that we have our parsed output in l_of_d, we can use it for evaluation against an ideal output:

ideal = {

'Computers and Laptops': ['TechPro Ultrabook', 'BlueWave Gaming Laptop', 'PowerLite Convertible'],

'Gaming Consoles': ['GameSphere X']

}

Next,

correct = 0

This line initializes a variable named correct to 0. This variable will be used to count how many categories in the parsed response (l_of_d) match those in the ideal output.

Next,

cat = d.get('category') # 4. Checking for valid categories and products.

prod_l = d.get('products')

cat: This variable retrieves the value associated with the key 'category' from the dictionary d. It represents the name of the category (e.g., "Computers and Laptops").

prod_l: This variable retrieves the value associated with the key 'products', which is expected to be a list of products belonging to that category (e.g., ['TechPro Ultrabook', 'BlueWave Gaming Laptop']).

Next,

prod_set = set(prod_l) #1

converts the list of products (prod_l) into a set (prod_set).

Using a set allows for easier comparison operations, such as checking for equality, subsets, or supersets.

Next,

prod_set_ideal = set(ideal.get(cat)) #2 # 1 and 2 compare

Next,

if correct = 1

if not correct = 0

    print superset or subset

Next,

Calculating and returning an accuracy percentage based on matches.