# Experiment 1 – testing different few shot training datasets

## Numbers vs 4 Numbers

|  |  |
| --- | --- |
| greater than symbol | 0.5 |
| semi-colon | 0.5 |
| question mark | 1 |
| exclamation point | 0 |
| number 9 | 1 |
| number 5 | 1 |
| number 7 | 1 |
| number 0 | 1 |
| letter A | 1 |
| letter R | 0 |
| letter B | 1 |
| letter O | 1 |
| Triangle | 0.5 |
| Rectangle | 1 |
| Skull | 0 |
| Heart | 0 |
| Tree | 0 |
| Face | 0 |
| chess board | 0.5 |
| Animal | 0 |
| **Total** | **11** |

## 3x letters, 3x punctuation, 3 objects

|  |  |
| --- | --- |
| greater than symbol | 1 |
| semi-colon | 1 |
| question mark | 0 |
| exclamation point | 0 |
| number 9 | 1 |
| number 5 | 0.5 |
| number 7 | 0 |
| number 0 | 1 |
| letter A | 1 |
| letter R | 0 |
| letter B | 0 |
| letter O | 1 |
| Triangle | 0 |
| Rectangle | 1 |
| Skull | 0 |
| Heart | 0.5 |
| Tree | 0 |
| Face | 0 |
| chess board | 0 |
| Animal | 0 |
| **Total** | **8** |

## 4 Numbers, 3x letters, 3x punctuation,

|  |  |
| --- | --- |
| greater than symbol | 0 |
| semi-colon | 1 |
| question mark | 1 |
| exclamation point | 1 |
| number 9 | 1 |
| number 5 | 0.5 |
| number 7 | 0.5 |
| number 0 | 1 |
| letter A | 1 |
| letter R | 0.5 |
| letter B | 0 |
| letter O | 1 |
| Triangle | 1 |
| Rectangle | 0 |
| Skull | 0 |
| Heart | 0 |
| Tree | 0 |
| Face | 0 |
| chess board | 1 |
| Animal | 0 |
| **Total** | **10.5** |

## 2 Numbers, 2x letters, 2x punctuation,

|  |  |
| --- | --- |
| greater than symbol | 1 |
| semi-colon | 0.5 |
| question mark | 0 |
| exclamation point | 0 |
| number 9 | 0.5 |
| number 5 | 1 |
| number 7 | 1 |
| number 0 | 1 |
| letter A | 1 |
| letter R | 0 |
| letter B | 0.5 |
| letter O | 1 |
| Triangle | 1 |
| Rectangle | 1 |
| Skull | 0 |
| Heart | 1 |
| Tree | 0 |
| Face | 0.5 |
| chess board | 1 |
| Animal | 0 |
| **Total** | **12** |

# Experiment 2 testing

Text, letter

Description automatically generated

Images displayed on a computer screen are made up of tiny dots of color called pixels. These pixels are arranged in a grid, called a pixel matrix, to form an image. The size of the matrix determines the resolution of the image, with larger matrices providing more detail. In this project, we will focus on pixel matrices of size 8x8, which are small enough to easily generate using GPT-3.

To create an image with a pixel matrix, each pixel must be assigned a value of 0 (black) or 1 (white). The resulting matrix is a sequence of 0s and 1s that represents the image. For example, a simple 8x8 pixel matrix of a square could be represented by the following sequence of 64 numbers:

11111111

10000001

10000001

10000001

10000001

10000001

10000001

11111111

In the following examples, we will show how to use GPT-3 to generate pixel matrices for simple objects like numbers and shapes. Keep in mind that the examples given are not strict guidlines and other solutions exist. The important thing is that each matrix should have 8 rows, as this is the size of the display.

Graphical user interface, text, application

Description automatically generated