

## Readme file

The given problem is based on the multimodal information (Images and texts) for generating the domain specific code (DSL).

### Task 1:

#### **- The dataset analysis carried out in the colab notebook (dataset\_description.ipynb)**

The task one is dataset analysis and description of the dataset based on the different aspects of data such as-

- No. of available images and corresponding labels
- Image formats and labels formats
- Shape of images and size of labels for corresponding images
- The different info in the each images and corresponding labels in each dataset (D1,D2,D3) is extracted in the separate excel file (dataset\_description.xlsx)
- The unique tokens in each dataset (D1,D2,D3) with respective size is extracted in the separate excel file (unique\_tokens\_labels.xlsx)
- The sample images and the corresponding labels are visualized in the colab notebook for understanding the images and the labels

Please refer for following contents for details :

- **Colab Notebook: dataset\_description.ipynb**
- **Excel file : dataset\_description.xlsx**
- **Excel file : unique\_tokens\_labels.xlsx**

**Brief about all the datasets: generated in the notebook**

Dataset	Image_format	Label_format	No.of samples	Unique_tokens
D1	.png	.gui	300	15
D2	.png	.gui	300	28
D3	.png	.gui	300	54

Unique Tokens in the datasets:

Unique_tokens_D1	No.of tokens_D1	Unique_tokens_D2	No.of tokens_D2	Unique_tokens_D3	No.of tokens_D3
btn-inactive quadruple } row btn-green , { btn- active single small-title header double btn-orange text btn-red	15	quadruple radio medium-title row rating , btn-active double icons burger-link btn- inactive { ck-box btn-red } small- title canvas- header btn- orange text sidebar btn-green single cart-link header search-bar dropdn slider page-title	28	quadruple logo radio medium-title row rating logo- img entry , btn- active double field-active table user-img table- row sidebar-left- home-right icons burger-link category btn- inactive user-img- user-name components field- inactive { cards ck-box nav table- col user-name btn-red logo-txt } img-txt-btn-card img-card img-btn- card small-title canvas-header btn-orange text home sidebar orientation btn btn-green single cart-link header btns user search- bar dropdn sidebar-right- home-left slider page-title	54

**Task2 :**

The following approach is followed in designing the model

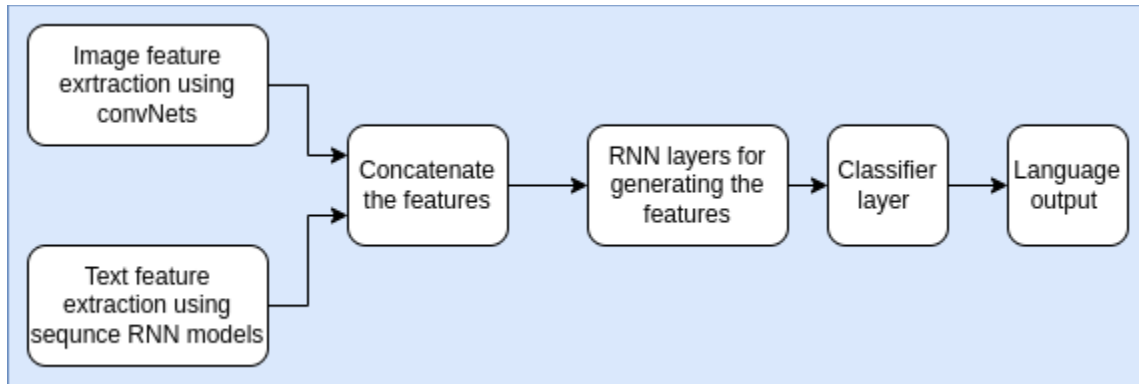


Fig: Approach followed in generation of code from the images

The image features are extracted using the pretrained models for all three datasets

**Dataset D1 : VGG16**

**Dataset D2 : ResNet50**

**Dataset D3 : EfficientNetB0**

The language model and classification model used for all the datasets are

**Dataset D1 : LSTM**

**Dataset D2 : GRU**

**Dataset D3 : LSTM**

The image model ConvNets and language models RNN take the image and language DSL as inputs and their outputs are concatenated and fed to the another RNN model which generates the language tokens as output.

The network details provided in the colab notebook

**D1\_images\_to\_text.ipynb**

**D2\_images\_to\_text.ipynb**

**D3\_images\_to\_text.ipynb**

**Training and Evaluation:**

Training samples 95% (285)

Test samples 5% (15)

Evaluation can be seen using the real text and predicted text

BLEU score calculated to decide overall efficiency

### Training Configuration:

Dataset	No. of Epochs	Optimizer	Learning rate	Train samples	Test samples
D1	1500	Adam	0.001	285	15
D2	1500	Adam	0.001	285	15
D3	1500	Adam	0.001	285	15

### Results:

Dataset	BLEU Score
D1	0.27968146945794126
D2	0.10453286648992965
D3	0.2782158332195104

\*\* Results can be varied by changing hyperparameters and training configuration, Dataset quantity

### Package Dependancies:

Dependencies are generated using the session\_info

Click to view session information

-----

```
cv2          4.6.0
google       NA
keras        2.9.0
keras_preprocessing 1.1.2
matplotlib   3.2.2
nltk         3.7
numpy        1.21.6
pandas       1.3.5
session_info 1.0.0
sklearn      1.0.2
tensorflow   2.9.2
```

-----

Click to view modules imported as dependencies

-----

```
IPython       7.9.0
jupyter_client 6.1.12
jupyter_core   5.1.0
notebook       5.7.16
```

```
-----  
Python 3.8.16 (default, Dec 7 2022, 01:12:13) [GCC 7.5.0]  
Linux-5.10.133+-x86_64-with-glibc2.27  
-----
```

Session information updated at 2022-12-31 13:26

#### **Steps for training:**

Load the dataset and train the model as per the given sequence in the notebook.

#### **Evaluate Model:**

Test dataset provide to evaluate function to generate the text output and evaluate the accuracy using BLEU score.

#### **References :**

Beltramelli, Tony. "pix2code: Generating code from a graphical user interface screenshot." Proceedings of the ACM SIGCHI Symposium on Engineering Interactive Computing Systems. 2018.

<https://towardsdatascience.com/code2pix-deep-learning-compiler-for-graphical-user-interfaces-1256c346950b>

<https://github.com/floydhub/pix2code-template/blob/master/pix2code.ipynb>

<https://medium.com/@srikar.appal/paper-review-3-pix2code-generating-code-from-a-graphical-user-interface-screenshot-6191844dcb07>

<https://theosz.medium.com/pix2code-automating-front-end-development-b9e9087c38e6>  
<https://towardsdatascience.com/foundations-of-nlp-explained-bleu-score-and-wer-metrics-1a5ba06d812b#:~:text=Bleu%20Scores%20are%20between%200,rarely%20achieve%20a%20perfect%20match.>