

Wire2Code

Parag Khuman

Department of Computer
Engineering Vidyavardhini's
College of Engineering and
Technology Vasai, India.

Gauri Bodke

Department of Computer
Engineering Vidyavardhini's
College of Engineering and
Technology Vasai, India.

Moh. Saif Mundhekar

Department of Computer
Engineering Vidyavardhini's
College of Engineering and
Technology Vasai, India.

Dr. Swapna Borde

Department of Computer Engineering
Vidyavardhini's College of Engineering and Technology
Vasai, India.

Abstract : The major obstacle in developing any web based product is the prototyping of the desired wire-frame. Rather on core logical features, programmers tend to focus and spends large amount of time on repetitive task of UI implementation. Thus, we can automate the task of UI prototyping just by feeding a snapshot of designed wire-frame to the model. Image recognition through machine learning models automates the process of implementing wire-frames into HTML or styled HTML codes. The precision and recall factors have been calculate as per the element classes. The image class has the highest precision factor succeeded by input and button classes.

Keywords—Wire-frames, CNN, UI

I. INTRODUCTION

Fundamental step in creating any web based application is the HTML code. Basic Prototyping of the whole project starts designing the User Interface on paper, and then implementing the prototype to get basic insight of the design. This creates a wide gap between the designed model and basic usable interface. The basic steps are usually repetitive, thus we eliminate the repetitive process. This overcomes a huge gap between a common user and a professional web developer. The process of project developing starts from drawing a rough sketch of a wire-frame depicting outlines and sections of the webpage. This project creates an application which transforms the rough wire-frames sketches into basic HTML codes with some bootstrap styling using Machine Learning and Artificial Intelligence. This project involves major challenges such as

- Developing a machine learning model which performs:
 - Detection of the wireframe elements drawn on the paper
 - Eliminating human errors from designs
 - Transforming wire-frame into actual code
 - Simplified output to the user

- Creating Datasets involving actual design and their respective tokens
- Measuring the accuracy and performance of the model

II. PROGRAMMING METHODOLOGY

A. Overview

Implementing client-side interface depending on the design mock ups i.e. the wire-frame created by the developer. It is very time consuming as the major time is wasted on designing the user interface rather than focusing on actual logic and methodology of the application. Most of the languages are domain specific languages (DSL) like programming languages, markup languages. They are developed for specialized usage. Therefore, using domain specific languages can limit the complexity of the language which needs to be modeled. The project incorporate with the convolutional neural networks, long short term memory model and recurrent neural networks.

B. Basic Model Architecture

The first phase is the training of the vision model and with the help of tokens generated, Sampling is performed which encompasses the second phase. For the training phase, the context and the wire-frame is provided to the LSTM and convolutional layers then the output vectors are provided as input to the second LSTM layer which actual acts as a decoder.

The sequence of tokens which are encoded related to the domain specific language code is encoded by the LSTM language model. The process is repeated on a stack of LSTM layers. The sampling is performed with the help of softmax layer. The previous predicted output is used with the current input for accurate predictions. The output of the training phase is the DSL generated tokens. The sampling phase takes the same input as the training without prior contexts.

III. FRAMEWORK

HTML markup is a hierarchy of objects known as DOM(Document Object Model). It has containers like div, footer, header, sections, paragraphs, images, buttons, etc. All these elements are relatively arranged in a webpage.

The major types of elements can be listed as:

- Style and CSS(Cascading Style Sheets) : This contains some tags which determines the certain position of an element on the webpage. Eg. Align, width, height, focus, hover.
- Unlabelled elements : These are most functional or structural elements, they drag the engine to the specific path. Eg. Header, footer, div

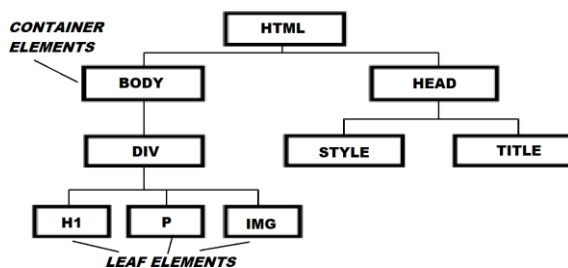


figure 1 : Document Object Model Tree

A. Preprocessing Process

1. Image Noise Reduction

The project captures real world images as input as it contain noise content due to variations of camera sensor. Noisy

<images causes inaccurate edge detection which gradually leads to inaccurate element detection. The project uses median filter as a noise reduction filter to get a smooth image. Gaussian filter often blurs the edges and edges are important in edge detection.[2]

2. Edge Detection

Vital part is detecting the element from the wire-frame. Therefore, detecting edges is most crucial. We are using techniques like Canny operator for filtering purposes. As it efficient to detect black pen on white paper. [2]

3. Segmentation

This is used to detect boundaries of the element. This project uses color based segmentation which uses color to differentiate boundaries. With algorithms such as histograms and region growing.[2]

B. Machine Learning

1. Artificial Neural Networks(ANN)

ANN is computational unit similar to biological neural network.. It has input layers and output layers. The input layers is passed through an activation function and the output is present.

2. Convolutional Neural Network(CNN)

Mainly used in image related processing. Multi Layer perceptron can also be used but they the problem of full connectivity. Therefore, Convolutional Neural Networks proves to be efficient than other techniques. They perform convolution operation on the input images. [2] It has three layers: Input, Output, one or more hidden layers. Further, the hidden layer can be divided into multiple layers like

- Convolutional Layers which perform convolution based on the input based on receptive fields.
- Pooling Layers takes output from previous neurons and feeds it to the next neuron. This helps to reduce the parameters and the size of representation which eventually reduce overfitting.
- Fully Connected Layers has fully connected planar graph structure. Each neuron is connected to adjacent neurons to predict final output which derives the features.

3. Long Short Term Memory Model(LSTM)

LSTMs are feedback networks. It contains input, output and a feedback layers. The error is backpropagated which is obtained from the output layer. The model is trained in supervised learning. LSTM often uses logistic sigmoid function. It is used to get the accurate results and less spatial complexity.[1]

C. Datasets

Datasets are generated such that

- Exploring website and then sketching
- Drawing and matching
- Auto generating mockups

The model predicts five classes which matches the wire-frame elements as follows:

1. Image : <svg>, , <video>

2. Button :

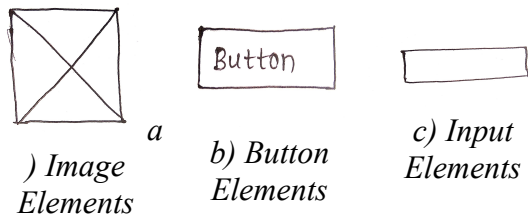
<button>, <a> 3.

Title :

<h1>, <h2>, <h3>

4. Input : textinput, textarea, range, input, slider

5. paragraph : <p>, ,



d) Paragraph Elements e) Title Elements

Figure 2 : Samples of Sketched elements of five elements

D. Algorithm

The project has three phases

1. Snapshot of the image and uploading to the API
2. Extraction of the elements
3. Inferring the structure

Pseudo Code :

Data : Elements arranged in ascending order by area or position

```

for Elements in E do
    contains=[]
    for elements in E2 in sectioned do
        if E1 contains E2 then
            contains=>push(E2)
            section=>remove(E2)
        endif
    e1=>contains=contains
    endforloop
    section=>append(E1)
endforloop

```

IV. RESULTS

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

For each class, the precision and recall factors have been calculated[2].

Where,

TP is True Positive, instances where the bounding box matches

FP is False Positive, instances where there is no corresponding elements

FN is False Negatives, instances where the elements are not detected.

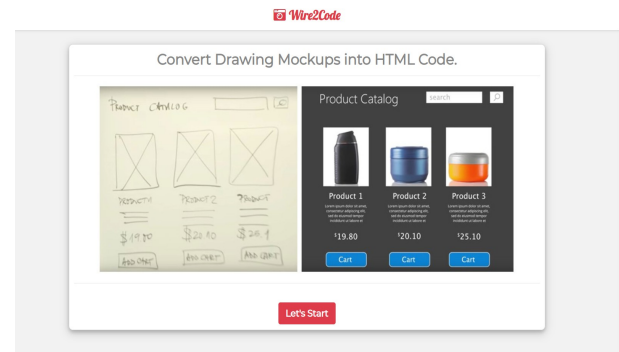


figure 4 : Sample Input Output

V. CONCLUSION

Wire2code, a novel method to generate computer code given a single GUI image as input. While our work demonstrates the potential of such a system to automate the process of implementing GUIs, we only scratched the surface of what is possible. Our model consists of relatively few parameters and was trained on a relatively small dataset. The quality of the generated code could be drastically improved by training a bigger model on significantly more data for an extended number of epochs. Implementing a now-standard attention mechanism could further improve the quality of the generated code.

ACKNOWLEDGMENT

We thank professor Dr. Swapna Borde, Department of Computer Engineering, Vidyavardhini's College of Engineering and Technology, Vasai who provided insight and expertise that greatly assisted throughout this project. Also for her constant encouragement and support throughout the work and helping a lot for the preparation of this paper.

REFERENCES

- [1] Tony Beltramelli, "Pix2code, Generating code from a graphical user interface screenshot", 22 May 2017.
- [2] Alexander Robinson, "Sketch2Code: Generating a website from paper mockup"