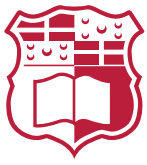
Contextualised Word Prediction System

Liam Bugeja Douglas

Supervisor: Prof. Alexiei Dingli

May, 2023

Submitted in partial fulfilment of the requirements
for the degree of B.Sc. ICT in Artificial Intelligence (Hons.).



L-Università ta' Malta
Faculty of Information &
Communication Technology

Abstract

In recent years language modeling has become an important concept in natural language processing applications. An area which is extensively researched in natural language processing is word prediction, which is a process that involves suggesting the most probable next word in a given text based on the previous context of the words. This technique is used in many text-related applications and allows users to save time whilst typing, leading to faster and easier communication between individuals. Whilst state-of-the-art language models have been rapidly improving in word prediction due to model optimisation and better training techniques, these models often struggle to predict the correct word if they are given limited text input.

This study aims to investigate the potential improvement in word prediction performance by enriching language models with contextual data, by using image classification and speech recognition. For image classification, four different classification models were evaluated including VGG-16, VGG-19, and Inception V3 to predict five indoor classes (bathroom, bedroom, dining room, kitchen, and living room) from a house room image dataset. For speech recognition, Google Cloud Speech-to-Text was employed to transcribe spoken words into text. Large language models, including RoBERTa, ELECTRA, and BERT were then used to evaluate the effectiveness of the image classification and speech recognition by integrating the predicted indoor room and the information obtained from speech transcription before the user input. To evaluate the models a customised multimodal dataset was created with indoor rooms, recorded speech, and text input. To ensure the models were tested on new data, a separate language model was used to generate the text and speech input.

The study revealed a noticeable enhancement in word prediction accuracy across all the language models when the additional context is used. Moreover, the system showcased an improvement of 10% in terms of word prediction accuracy, with the speech recognition data giving the most substantial impact.

Acknowledgements

First of all, I would like to start by expressing my sincere gratitude to my parents, who have supported me and guided me throughout my educational journey. My gratitude also extends to my friends whose constant support and encouragement have been instrumental to keep me focused on my goals and without them, I would not be half the man I am today. I dedicate this FYP to my other half, her constant reassurance and support has made me achieve one of my proudest moments in my life. Finally, I would like to thank my supervisor Prof. Alexiei Dingli, without his constant guidance and feedback the work found in this FYP would not be possible.

Contents

List of Figures

List of Tables

List of Abbreviations

AAC Augmentative and Alternative Communication.

AI Artificial Intelligence.

ANN Artificial Neural Network.

CNN Convolutional Neural Network.

DL Deep Learning.

DNN Deep Neural Networks.

HMM Hidden Markov Model.

KS Keystroke Savings.

LM Language Models.

ML Machine Learning.

NLP Natural Language Processing.

RF Random Forest.

RNN Recurrent Neural Network.

STT Speech To Text.

SVM Support Vector Machines.

WER Word Error Rate.

WPR Word Prediction Rate.

1 Introduction

1.1 Problem Definition

In recent years Language Models (LM) have become increasingly sophisticated, and models such as BERT and RoBERTa, have shown impressive results in Natural Language Processing (NLP) tasks. An important problem in the domain of NLP is next-word prediction since it simplifies the process of typing by suggesting the next word to a user based on the previous words. Whilst research has been done to capture more context from the previous words [1], LMs still struggle to predict the intended word with limited text input. This is especially problematic since most communication done in the real world uses short sentences with little context. Furthermore, individuals who rely on assistive technology to communicate with others are the most affected as it reduces the overall effectiveness of the technology if most of the predictions are incorrect. Research has been done to improve Augmentative and Alternative Communication (AAC) devices for affected people, with most studies emphasising that improvement in word prediction accuracy would impact the quality of life for such individuals since it leads to easier communication with others.

1.2 Motivation

The primary motivation of this study is to shift the attention to capturing context from the environment for LMs rather than relying on increasing the training time and size of the corpora to improve LMs, mainly in the task of word prediction. By the end of this study, a working prototype will be created in which a language model will be able to use additional models to capture the current context of the environment. Whilst research on incorporating LMs with speech data [2] and image data [3] is not a new idea, there has yet to exist a solution that leverages the two to enhance word prediction. However, most LMs which are incorporated with speech recognition models are only used to improve real-time speech recognition accuracy. Furthermore, most vision-based LMs are used to tackle image domain problems such as visual captioning and spatial manipulation.

1.3 Aims and Objectives

The main aim of this study is to improve the word prediction rate of LMs through the use of speech recognition and image classification. These two methods are used to capture the context within the environment which is useful to the LMs. To achieve this aim the following objectives have been set together with their respective research questions, which includes the chapter in which the objective is being addressed.

Aim: To investigate the potential benefit of using speech and image data to improve word prediction in language models.				
Objective		**Research Question**	**Method**	**Chapter**
O1	Identify the current indoor room the user is in by using a deep learning model.	What are the methods identified in the literature review which are able to accurately predict the environment?	Literature Review and Methodology	3.3 and 4.4
O2	Identify user speech using a deep learning model, which converts the speech into text as output.	What are the models identified in the literature review which are able to convert speech into text?	Literature Review and Methodology	3.2 and 4.3
O3	Create a multimodal dataset containing images of indoor rooms, recorded speech, and text data.	How can we create a multimodal dataset containing images, speech, and text data?	Methodology	4.6
O4	Evaluate the effectiveness of deploying objectives 1 and 2 to a language model to improve word prediction.	How much is the word prediction rate of a language model affected by objectives 1 and 2?	Evaluation	5.2

Table 1.1 Research design of the study (Self).

1.4 Proposed Solution

The proposed solution for this project is to gain additional context for LMs to improve their word prediction accuracy. This will be done by using a deep learning model which will be able to classify the current environment the user is in and transform the prediction into a meaningful sentence that will give further context. Pre-trained models and transfer learning will be used to achieve the best results possible and experiments will be carried out to determine the best classification model. Furthermore, a speech recognition model will be used to capture the current conversation into textual data which will also be used with the derived sentence made by the image classification model. The effectiveness of the added context will be tested on customised data containing image, audio, and text data and will be compared to the same dataset using the text data alone.

1.5 Document Structure

The remainder of this document is split into the following chapters. In Chapter 2, an overview of the techniques that will be employed is discussed, along with a brief description of multimodal LMs. In Chapter 3, the approaches taken by previous studies are thoroughly reviewed and analysed. In Chapter 4, we delve into the implementation of the system and the design choices are analysed. In Chapter 5, the effectiveness and usability of the system is evaluated. This section also presents the results of the experiments conducted and are analysed as well. In Chapter 6, a comprehensive summary of the results and possibilities of future work is provided.

2 Background

In this chapter, we provide an essential overview of the foundational concepts that form the basis of this study. An introduction to Artificial Neural Network (ANN) and Deep Learning (DL) technology is discussed which are an integral component of the study. Additionally, we discuss the fundamentals of word prediction, speech recognition, and image classification techniques which are the key areas of research in this study.

2.1 Artificial Neural Network

An ANN is a Machine Learning (ML) method that evolved from the concept of simulating the human brain to solve complex problems [4]. The basic structure of an ANN comprises interconnected nodes in an input layer, a single hidden layer, and an output layer. These layers are linked to one another, enabling the representation of complex functions through self-learning with data sets.

2.2 Deep Learning

Unlike simplistic neural networks, which typically contain two to three hidden layers, DL allows a Deep Neural Networks (DNN) to have more processing layers [5]. DL gained popularity due to the advancement in computer technology, which allowed for more complex neural networks to be created and solving domains with large and high dimensional data that ANNs were incapable of solving [6]. The use of DL has drastically improved state-of-the-art Artificial Intelligence (AI) technologies such as LMs, speech recognition, and image classification.

2.3 Word Prediction

Word prediction is a widely used technique in NLP that tackles the language domain by predicting the next word based on the context of the previous words. Word prediction aids in faster communication since it significantly reduces the time it takes to type and is commonly used in various applications. One common technique which is used to tackle word prediction is the n-gram language model, which analyses the frequency of word combinations of length n on a trained corpus. However, recently with the introduction of LMs such as the RoBERTa model [7], word prediction accuracy has improved since these models are able to model long-range dependencies and capture relationships between words.

2.4 Speech Recognition

Speech recognition is a ML technique that processes human speech into text, it is also one of the most researched areas in speech processing [8–10]. The most popular algorithms that have been applied to speech recognition include the Hidden Markov Model (HMM), Support Vector Machines (SVM), DNN, and hybrid approaches which will be further explained in Section 3.2. Furthermore, open-source Speech To Text (STT) services are readily available online, such systems include Google Cloud and Microsoft Azure STT services which have achieved state-of-the-art results.

2.5 Image Classification

Image Classification is a ML model tasked with labelling images to their respective class, these models require an image input and return a list of predictions of the categories the image belongs to. Convolutional Neural Network (CNN) is the most widely used DL technique to solve image-related issues such as detecting objects, face recognition, and classifying images. Recent research also suggests that CNNs have drastically improved in the classification of images due to fast graphical processing units and techniques utilizing parallel and distributed computing [11].

2.6 Summary

This chapter gives an overview of the techniques which will be used in this study. A summarised explanation of ANNs, DL, and word prediction is discussed. Furthermore, a brief description of image classification and speech recognition is also highlighted. In the following chapter, a literature review of current word prediction techniques will be presented. Furthermore, solutions for image classification and speech recognition techniques are also described and reviewed to achieve the best results. The chapter will end with a section dedicated to multi-modal language models to evaluate the effectiveness of using speech and image data on large language models.

3 Literature Review

This chapter aims to provide a comprehensive overview of the latest research that is relevant to this study. We will explore various state-of-the-art models and techniques used in word prediction, speech recognition, and image classification and how these models improved. Each section will contain an analysis to determine the most effective solution to solve the objectives of this research. Additionally, we will also delve into multimodal language models and examine how these models utilised different techniques to aid in language modeling.

3.1 Word Prediction

As briefly described in Section 2.3 word prediction is a typing assistance tool that aids in faster communication. Early word prediction systems made use of n-gram language models, which predict the probability of a sequence of words based on the frequencies of their constituent n-gram [12]. Recently, there has been an increase in interest in integrating DL with LMs, which has led to a plethora of research to measure the effectiveness of word prediction using these models [13, 14]. However, as these state-of-the-art models have become more accurate in a vast amount of NLP tasks, they have also become computationally expensive to train. Shoeybi et al. [15], present a new technique in which they train a large language model in a model-parallel way. This new training technique allows the model to be trained on multiple hardware devices using data parallelism. The technique also allows models to make use of a cluster of graphical processing units which reduces the training time drastically. The results of this study showed that by using this new training technique, it significantly reduced the training time whilst achieving state-of-the-art results on a range of NLP problems. With the emergence of sophisticated LMs, research on developing appropriate dataset to test and evaluate these models have also been made. Paperno, et al. [16], presented the LAMBADA (LAnguage Modeling Broadened to Account for Discourse Aspects) dataset as a means of evaluating models on their ability to comprehend the long-term context of sentences by predicting the final word of a passage. The authors noted that current state-of-the-art LMs, struggled to achieve satisfactory performance on the LAMBADA challenge at the time of its introduction. They suggested that future work should focus on developing models that are capable of capturing long-term contextual information in order to improve word prediction accuracy.

3.1.1 Language Models for Word Prediction

A recent study was made to measure if LMs are better at word prediction than humans. Buck et al. [17], concluded that LMs are superior to humans in the area of word prediction. The researchers also found that even small LMs had better accuracy than humans. Furthermore, the authors noted that training large LMs on word prediction is the least efficient method of training. However, the authors also mentioned that it has proven to be the most cost-effective method of training with many models adopting it as part of their training. To achieve greater accuracy in word prediction, research has been made on improving current state-of-the-art LMs using different techniques. Song et al. [18], introduced a novel approach to improve the language generation for LMs called MASS (MAsked Sequence to Sequence Pre-training). This works by the sequence-to-sequence learning framework, in which the encoder takes an input sentence with a masked fragment and the decoder predicts the masked fragment based on the encoder representations. Furthermore, the researchers also fine-tuned a variety of zero/low-resource language generation tasks. With these combined, the results showed significant improvements when compared with other baseline models.

Research is also being done on a similar training technique of MASS called Masked Language Modeling (MLM). This technique requires the model to predict a single token instead of a whole sequence, which allows the model to learn the relationships between words leading to better word prediction accuracy. Salazar et al. [19], evaluate different MLMs via their pseudo log likelihood. This is a method in which it estimates the likelihood of a binary variable given the values of other variables in a model. This method is often used to score language modeling, where the aim is to predict the probability of the next word in a sentence based on the preceding words. The authors made use of three popular MLMs called BERT, RoBERTa, and GPT-2. The results from the study showcase that the RoBERTa language model had superior results in the pseudo log likelihood scoring when compared to the other mentioned models. Yu et al. [20], experimented with compressing a Recurrent Neural Network-based Language Model (RNN-LM) by using shared matrix factorisation. By utilising this new approach, the authors claim that they achieved an approximate compression of around 8-fold with negligible losses with regard to the model's performance. Furthermore, due to the compression technique, the model was able to be used on a mobile device and delivered superior results in terms of KS and WPR as shown in Table 3.1. The results were also compared with other word prediction systems used by mobile phones. KS is a measure of the percentage of keystrokes saved by using a language model's predictions compared to a keyboard without prediction capabilities. WPR is the percentage of accurately predicted words in a dataset.

Developer	**KS**(%)	**WPR**(%)
RNN-LM	65.11	34.38
iOS	64.35	33.73
Swiftkey	62.39	31.14
Samsung Galaxy S6	59.81	28.84
G-board S6	58.89	28.02

Table 3.1 Comparing different mobile word prediction systems in terms of KS and WPR [20].

Clark et al. [21], introduced a new pre-training approach called ELECTRA (Efficiently Learning an Encoder that Classifies Token Replacements Accurately). ELECTRA is trained by distinguishing between ”fake” and ”real” tokens in a text corpus. the fake tokens are created by replacing a certain amount of original tokens and replacing them with a random token. Results from this study demonstrate that the ELECTRA model outperforms BERT on several NLP tasks whilst requiring less computational power and training time.

From the reviewed literature, it can be concluded that before the integration of DL with language modeling, research was conducted on the use of n-gram language models to solve word prediction. However, by introducing DL in this field, research has shifted from n-gram models to large language models. Current research is focusing on improving these large language models by introducing new training techniques and optimising the models for specific tasks. Moreover, MLMs are being heavily researched due to their capability of achieving state-of-the-art results in various NLP tasks.

3.2 Speech Recognition

As discussed in Section 2.4 numerous techniques have been used to enhance speech recognition systems. Karpagavalli and Chandra [8], delved into the main components and processes of a typical speech recognition system which can be seen in Figure 3.1. The authors emphasised using the appropriate feature extraction technique in the Acoustic Front-end, which should be able to differentiate between similar-sounding speech without requiring large amounts of training data.

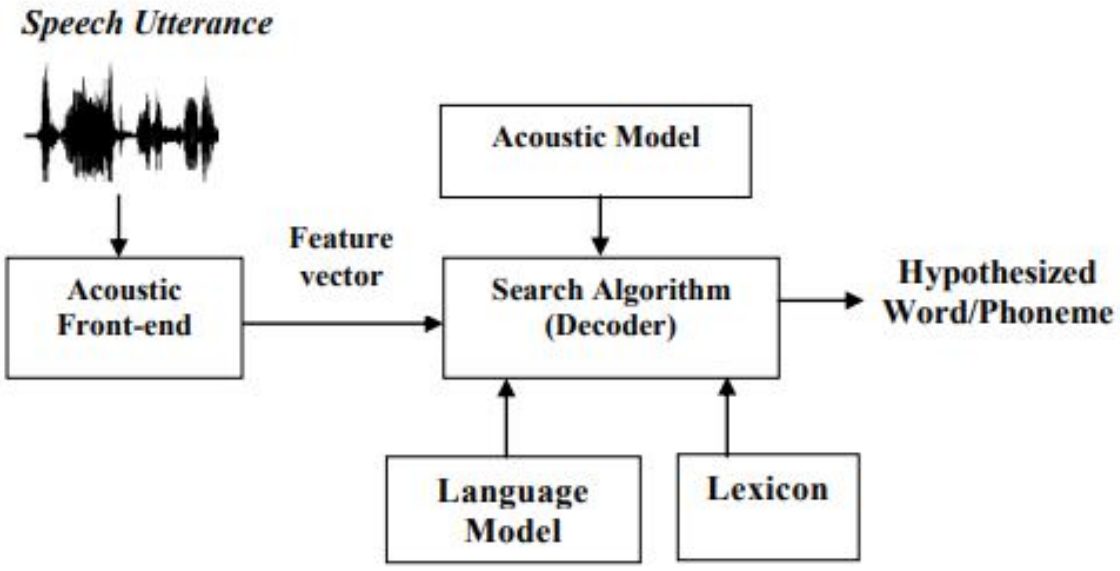


Figure 3.1 Speech Recognition Architecture [8].

The paper also mentions that of all the different feature extraction techniques used, the most popular is the Mel-frequency cepstral coefficient (MFCC) feature set. Ayvaz et al. [22], propose that MFCCs are popular due to their ability to process voice audio signals with high accuracy. After extracting the features and the speech is pre-processed, the acoustic model is used to identify and transcribe the information into text. Haridas et al. [23], mentioned the use of dynamic time warping (DTW), Fuzzy Logic, HMM, SVM and Wavelets as possible techniques which can be used for acoustic modeling. Whilst the authors mention that HMMs are the most widely used acoustic model in speech recognition systems, they suffer from two major drawbacks. The first issue is that HMMs assume that the features can be represented as a mixture of Gaussian distributions. The downfall to this is that if the representations are incorrect the features do not match with actual speech data leading to a significant decrease in the model's accuracy. The second issue is that HMM contain a significant amount of parameters and a single parameter can easily affect the model's accuracy. Therefore, it requires a lot of experimentation and fine-tuning to find the optimal number of parameters and their values. With the introduction of deep learning, the authors in [24] investigated the feasibility of using an HMM-Deep Neural Network (HMM-DNN) model, and compared it to a traditional HMM-Gaussian Mixed Model (HMM-GMM). The result from the experiments carried out in the study showcases that (HMM-DNN) models achieved greater performance when compared to HMM-GMM. However, the authors also noted that HMM-DNN models have certain limitations inherited from HMMs which include data-forced segmentation alignment, independent hypothesis, and multi-module individual training. Furthermore, the authors concluded by reviewing end-to-end models for speech recog-

nition and emphasised that future work should focus on building and fine-tuning these models. Nassif et al. [25], analysed different techniques used for speech recognition. The study concluded that the use of DNN models using HMM or GMM hybrids gave the best results. The authors also highlighted that there is a lack of research done on Recurrent Neural Network (RNN) models for speech recognition. They recommended that future work should focus on the use of Deep RNN models, more specifically Long Short Term Memory (LTSM) models. Radford et al. [26], experimented with the use of scaling weakly supervised pre-training on speech recognition models since hardly any research was done in this area. In this study, the authors made use of a large dataset containing 680,000 hours of labelled audio. The results showed that the proposed technique achieved high-quality results without the need for dataset-specific fine-tuning. Instead, by focusing on zero-shot transfer, the model was able to generalise well to other datasets without requiring extensive fine-tuning.

3.2.1 Speech Recognition Models

There are many companies offering STT models such as Google Cloud and Microsoft Azure which can be easily accessed by corporations or individuals seeking to use and experiment with these services. The STT models offered by these companies use state-of-the-art techniques and frequently add new models to accommodate new languages and accents that are currently not available. Furthermore, they often update their models based on current research and experiments to improve the accuracy of their models. A recent study tested three popular companies offering these STT models which include IBM Watson, Wit, and Google Cloud, with results showing that Google Cloud STT had better overall results [27]. The results also showed that among the three test speakers used in the study, Google's STT service had the smallest average error of 20.63% for WER. Another study also compared different STT models which included DeepSpeech, Google Cloud, IBM Watson, Microsoft Azure, and Kaldi. Alibegović et al. [28] concluded that overall Microsoft Azure had the best baseline model for general English as well as for its adapted model which was tested on an in-domain dataset. However, when using speech recognition systems in real-time, noisy environments are expected to be common and many speech recognition systems are vulnerable to noisy data, thus degrading the accuracy as stated in[29]. Xu et al. [30], experimented with various STT models which are previously mentioned, in their ability to accurately predict words when noisy data is introduced. Overall, the study concluded that Google Cloud and Microsoft Azure are more robust to environmental noise when compared to the other STT services. Moreover, the results also show that Microsoft Azure outperformed Google Cloud by a small margin.

From the reviewed literature, it can be concluded that MFCCs are still being used for feature extraction and HMMs are seeing a significant decline as acoustic models due to the introduction of DL. Hybrid models and DNNs are being proposed as the leading acoustic models due to their robustness and accuracy being proposed in studies. Furthermore, companies such as Microsoft are also conducting research to build speech recognition models capable of handling noisy environments. This continuous research has helped to build state-of-the-art speech recognition models.

3.3 Image Classification

As described in Section 2.5 image classification has drastically improved with the introduction of DL, with most researchers focusing on improving classification accuracy by optimising CNN architectures. The use of CNN models for image classification started with Krizhevsky et al. [31] when they proposed AlexNet, which won the competition in the ImageNet challenge. The architecture of a typical CNN can be split into two components called Feature Extraction and Classification as described in [32]. In the feature learning component, the CNN architecture learns about the image data by passing each image through different layers, that is the convolution layer and the average pooling layer. These layers extract the features through mathematical functions on the image data. Afterwards, the transformed data is sent to the classification component. The classification component receives the transformed data in a matrix form which needs to be flattened for the final layers. The flattened data is then passed through a softmax or sigmoid function which are used for multi-classification and binary classification respectively. Other algorithms such as SVMs and Random Forest (RF)s have been introduced to the field of image classification [33–35].

3.3.1 Image Classification Techniques

In [36], Wang et al. compared and analysed a SVM with a CNN for image classification problems. The study revealed that when there is limited training data, the SVM tended to outperform the CNN in accuracy. However, by increasing the size of the dataset the results show that the CNN gained an advantage and achieved an accuracy of 98% whilst the SVM achieved an accuracy of 88%. Furthermore, the SVM took longer to train than the CNN with the larger dataset, with this we can conclude that SVMs are viable when the training data is limited. Recent research indicates that data augmentation can provide additional training data, whilst also combating the problem of overfitting. O'Gara and McGuinness [37], researched the effect data augmentation has on deep image classification models. The results of the study showed that the model's accuracy drastically

improved by using data augmentation. Furthermore, the best data augmentation technique was found to be random erasing, the authors concluded that this is due to the technique being able to simulate occlusions. Additionally, the authors also found that even if a dataset contains a sufficient amount of images, carefully implementing data augmentation would still slightly improve the classification accuracy. Salman et al. [38], studied the effectiveness of transfer learning when it is used in image classification. Transfer learning is a commonly used technique in deep learning, where a pre-trained model is fine-tuned to a new image dataset. The idea behind transfer learning is to leverage the knowledge and learned features from a pre-trained model, which was trained on a large dataset, and apply it to a new and typically smaller dataset [39]. The results from the study indicate that fine-tuning pre-trained models using transfer learning techniques can yield superior performance compared to training models from scratch.

3.3.2 Pre-trained Models using Transfer Learning

Many pre-trained models are readily available for personal use such as VGG 16 and 19 [40], and Inception V3[41]. A recent study [42], investigated the use of an indoor room classification system using these pre-trained models and focused on classifying the following: bathroom, bedroom, dining room, kitchen, and living room. The authors made use of a scene dataset containing 11,600 images for each class, however, they excluded a number of images from the dataset that were irrelevant to the category. Results from the study, which can be viewed in Table 3.2, show that cleaning the dataset improved the accuracy of each model. Furthermore, it was concluded that the VGG19 outperformed the other models in this experiment, achieving an accuracy of 93.61%.

CNN Models	**All Data Accuracy (%)**	**Clean Data Accuracy (%)**
VGG16	87.78	93.29
VGG19	90.30	93.61
Inception-V3	79.11	84.05

Table 3.2 Comparison of performance of different transfer learning methods for indoor room classification with models using all data and cleaned data [42].

Another study focused on using pre-trained image classification models for product classification to help optimize pricing comparison. Mascarenhas et al. [43], made use of VGG16, VGG19, and ResNet50 to solve the classification problem. The results of the study showed that all three models had a high accuracy achieving 96.67%, 97.07%, and 97.33% for VGG16, VGG19, and ResNet50 respectively. Furthermore, each model was trained for twenty epochs showing that they achieved accurate results with a small number of training steps.

From the reviewed literature, it can be concluded that the CNN architecture has dominated the field of image classification. Whilst studies are being made on other algorithms such as SVMs and RFs, CNNs outperform these algorithms due to the advancement in DL, computer technology, and optimisation techniques. Moreover, current research also suggests that creating models from scratch is insufficient. Instead, by making use of pre-trained models and implementing transfer learning the models achieve a higher accuracy and greater robustness. Therefore, by leveraging the knowledge gained from large-scale datasets, pre-trained models can provide better results when needed for a specific task.

3.4 Multimodal Language Models

LMs have demonstrated excellent capability in solving complex tasks when it comes to the language domain. Models such as BERT and XLNet have revolutionised the field of NLP, however, these models lack the ability to interact with the world due to their incapability of processing different types of sensor data, such as speech and text. Research has been conducted on applying different sensor data to LMs to see if they are capable of processing real-time world problems. Anil et al. [44], introduced PaLM-2 which is an embodied multimodal language model capable of processing textual and visual inputs to better understand real-world problems. The authors showed that PaLM-2 is capable of image captioning, visual question answering, environment navigation, and language translation. The authors concluded that by introducing visual information to LMs, they can better understand the context of the input and generate more accurate and detailed responses. Furthermore, these novel LMs are able to perform tasks that require spatial reasoning and navigation. Another recent study investigated how visual information can aid LMs in text summarisation of textual data [45]. The authors introduced the Vision Enhanced Generative Pre-trained Language Model (VEG-LM), which is trained on a dataset of visual and textual data and focuses mainly on text summarisation. The model showed better results when compared to traditional LMs and the authors noted that with the use of appropriate visual information, the VEG-LM is capable of simulating human summarisation capabilities. Bapna et al. [2], focused on incorporating speech and

text data to improve the performance of speech recognition and language modelling. The authors introduced the Speech-Text Joint pre-training with Latent Alignment and Mixture of Experts (SLAM), which is a unified encoder for speech and language modelling. The SLAM model is trained to predict masked speech and text inputs, similar to the masked language modelling task used in other pre-training methods such as BERT. The results show that SLAM outperforms other pre-training methods such as BERT and Wav2Vec on a range of speech and language understanding tasks.

From the reviewed literature, it is evident that multimodal LMs outperform traditional LMs due to their capability of training and gathering a wider range of data. Furthermore, due to their ability to process different data using their respective sensor, these LMs can interact and solve real word problems. Whilst the concept of multimodal language models is not entirely new, recent advancements have popularised these approaches such as the introduction of GPT-4 [46].

3.5 Summary

This chapter illustrates a review of word prediction, speech recognition, image classification, and multimodal LMs. From the findings, it can be concluded that multimodal language models are superior to current language model systems from the results found in [2, 45], since they gather much more context than traditional LMs. Therefore, by including speech recognition and image classification data in LMs as textual information, we can improve word accuracy for current word prediction systems by exploiting the additional context. From the review carried out on different STT models, it was noted that Google Cloud provided superior results when the input does not contain noisy data. However, as stated in [30], Microsoft Azure is the most robust system due to the noise suppression capability it has in noisy environments. For image classification, it was noted that CNNs demonstrated superior accuracy when compared to other algorithms in image classification. Moreover, the review also highlighted that leveraging transfer learning on pre-trained models is the most effective approach to achieve state-of-the-art performance. Transfer learning enables consumer-grade computers to utilise these pre-trained models for specific tasks, which would not be feasible to train from scratch due to the computational power needed to achieve the same results. In the following section, we will outline the design decisions and implementation details of the contextualised word prediction system. Each decision made will be presented alongside its reasoning to provide a comprehensive understanding of the system.

4 Methodology

In this chapter, we will outline the methodology utilised to implement the system based on the findings discussed in the previous section. We will also delve into how the various AI techniques are used and how they work with each other to achieve the desired results.

4.1 System Overview

As previously discussed, the main goal of the contextualised word prediction system is to provide additional context to language models with the user text input. Therefore, the added context from the speech recognition and image classification models must be transformed into textual data so that the language model is able to process the information. Furthermore, the transformed data must be meaningful to impact the accuracy of the predicted word. In this study, we limited the image classification model to an indoor room dataset containing various rooms a typical house would have. Based on the prediction of the classification model the output will be transformed into a coherent sentence which will be combined with the output of the speech recognition model and the input text of the user. For the speech recognition model, we will be utilising a pre-trained speech recognition model capable of transforming speech data into textual data for the English language. The entire system, including testing, training, and results, can be accessed on GitHub[1].

4.2 System Architecture

The data flow of the system is illustrated in Figure 4.1, in it we can identify three main components. The first part of the data flow diagram (DFD) labelled Image Classification, features the image classification system that makes use of a deep learning model to predict the classification of the indoor room. It also showcases the transformation of the predicted class to meaningful textual data. The second part of the DFD named Speech Recognition highlights the speech recognition system which makes use of a deep learning model which transforms audio data into text data. Finally, the third part labelled Language Model showcases the combined text input that will be inputted into the language model. With this it creates a prediction based on the input text of the user which will be further improved by the added context.

[1] https://github.com/LiamBugejaDouglas/Contextualised-Word-Prediction-System

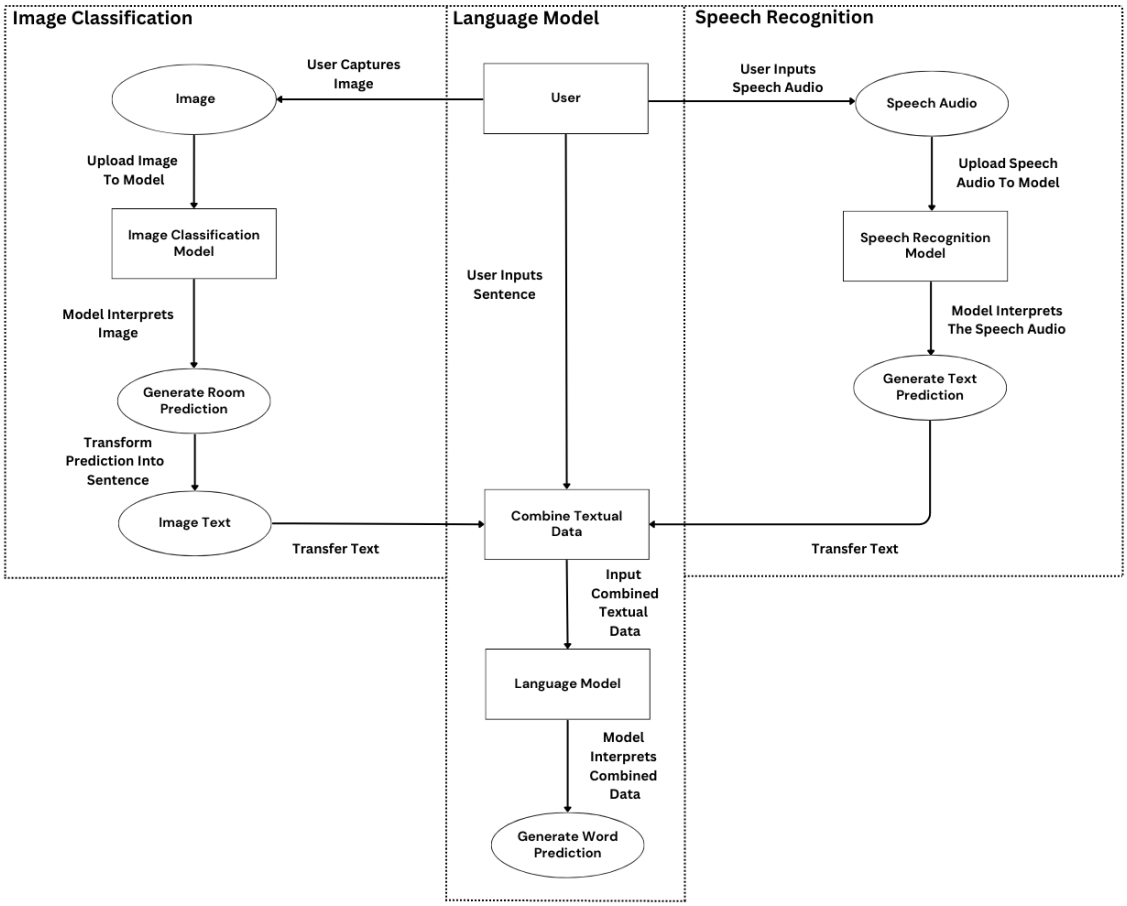


Figure 4.1 Data-flow diagram of the proposed system (Self).

4.3 Speech Recognition

As outlined in Section 3.2, STT services offered by large corporations offer the best results when it comes to speech recognition. In addition, these companies offer STT service compatibility with a multitude of programming languages. This is typically achieved through various methods such as providing libraries, APIs, or offering a cloud-based service. From all the available services, it was determined that two in particular, can be easily integrated into the Python environment, which includes Microsoft Azure Speech service and Google Speech service. Both SST services offer continuous streaming and timed streaming. Furthermore, the research done by Xu et al. [30], shows that Microsoft's STT service is more robust in noisy environments when compared to Google's solution. However, the STT service offered by Microsoft has a limited amount of free usage per account. Consequently, this would hinder the system as a crucial part would stop working due to the trial period. Fortunately, Google's STT offers a generous sixty-minute limit per account each day on its STT models. Although these companies do not

offer a completely free version of their STT services, it is essential to acknowledge that their existence is based on the profit motive, and providing state-of-the-art services for free is not a wise business decision. Additionally, in this study, we will be making use of a speech recognition model capable of transforming speech audio to text for the English language. Both Microsoft and Azure offer different models capable of recognising different accents including American, British, Australian, and Indian English accents. Whilst neither offer a specific model for the Maltese accent, it is worth noting that the Maltese accent is similar to the British variant. Taking all of the above into consideration, the system will make use of the Google Cloud STT service, more specifically we will leverage the British variant model.

Deploying Google's STT service involves a number of steps as shown in Figure 4.2. The system will first start by asking the user for a specific key input to start the recording of the speech audio. Next, the 'pyaudio' library is used to capture the input audio and will be saved as a waveform audio format. The recorded audio will be stopped once the same key input is pressed again. Afterwards, once the system has the audio saved it will utilise the 'speech_recognition' library to employ the Google STT API. This will transform the saved audio into the textual format, which is saved to be used later when combined with the other captured data. After the audio file is used the system deletes the audio format to save space. When implementing this process, it was noted that the audio being recorded had poor quality. As a result, we optimised two key parameters which were affecting the audio quality; the sampling rate and the frames per buffer. We increased the sampling rate to 44100 Hz and the frames per buffer to 1024 frames, with this the audio quality significantly improved and the accuracy of the Google STT service was enhanced as well.

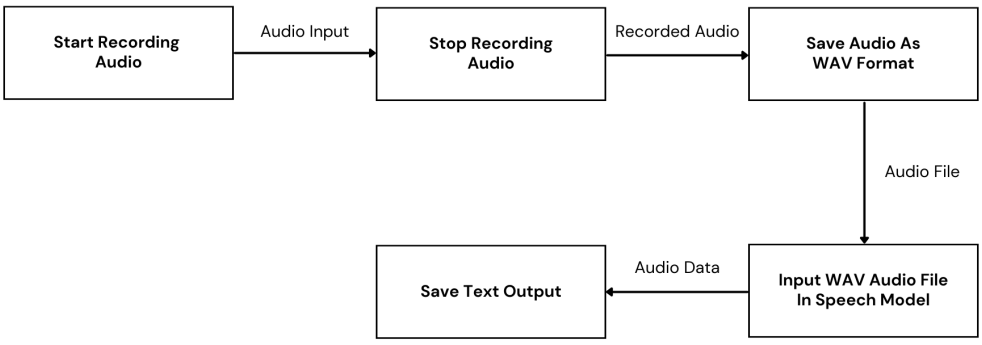


Figure 4.2 Block Diagram for the Speech Recognition System (Self).

4.4 Image Classification

As previously mentioned, we will be using an image classification model to classify different indoor rooms to gain additional context of the environment. We identified the two following deep learning frameworks to deploy the model: TensorFlow and PyTorch. Both frameworks offer similar features to enhance machine learning models such as GPU acceleration, integration of deep learning libraries such as Keras, and offer support for popular neural network architectures including but not limited to CNNs and RNNs. However, TensorFlow has a slight advantage over PyTorch when it comes to wider adoption in the field. This means that there are more tutorials and documentation available to access, making it much easier to use and debug as a framework. Therefore, due to easier access and the resources available on the TensorFlow framework we adopted to use it for this system.

Training and deploying the image classification model requires numerous steps as shown in Figure 4.3. We first have to obtain and clean an appropriate dataset for the model which will be used to train, validate, and test the model. Next, we need to build a model using the TensorFlow framework and the Keras API using the appropriate layers. In this study, we built a single model which was then compared with three pre-trained models. The model contains multiple convolutional layers, which capture local patterns and features. The convolutional layers apply 16 or 32 filters of size 3x3 to capture low and high level features. Furthermore, each convolutional layer contains a ReLU activation function, which is used to introduce non-linearity enabling the model to learn more complex

representations. A max pooling layer was also added after each convolutional layer to reduce spatial complexity whilst retaining the important features. Furthermore, to prevent overfitting and improve the model's ability to generalise, dropout layers were also added which in this case we used a dropout value of 0.5. After the convolutional layers, a flattening layer is used to connect the convolutional layers to the final layers. A dense layer with 128 units and ReLU activation is added to capture complex patterns from the flattened features. The output layer consists of 5 units with a sigmoid activation function. This enables the model to calculate the probabilistic value associated with each class and allows us to sum up the probabilistic values to 1 making it easier to interpret the predictions. Afterwards, the model is trained using the cleaned dataset and is fine-tuned based on the accuracy and loss values outputted by the model during training. Finally, the model is saved and utilised in the proposed system where it takes an image as input and generates its corresponding prediction as output. A visual representation of the model can be viewed in Appendix A.1.

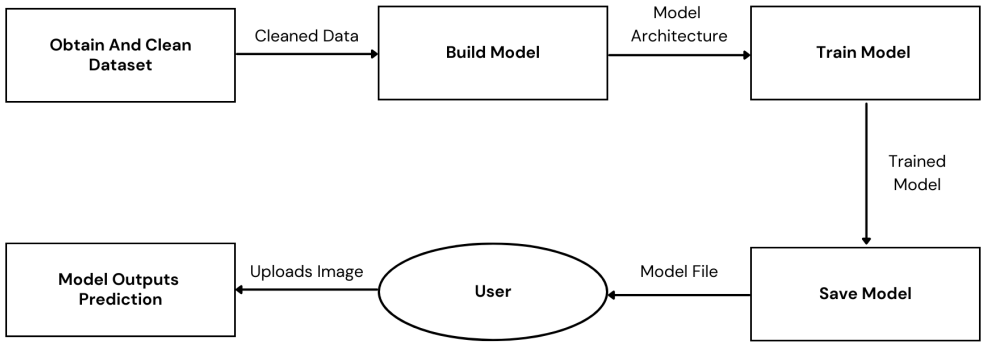


Figure 4.3 Block Diagram of Training and Deploying the Image Classification Model (Self).

The findings of Salman et al. [38], show that it is inefficient to build a model for a specific task. The authors mention that by using pre-trained models and leveraging the power of transfer learning, the models are more robust and yield superior performance. Furthermore, this was also concluded in [42], in which the researchers also used the technique of transfer learning to achieve greater results. Additionally, by using transfer learning the models require less training data and less time to train. Figure 4.4 depicts how transfer learning is leveraged on a pre-trained model.

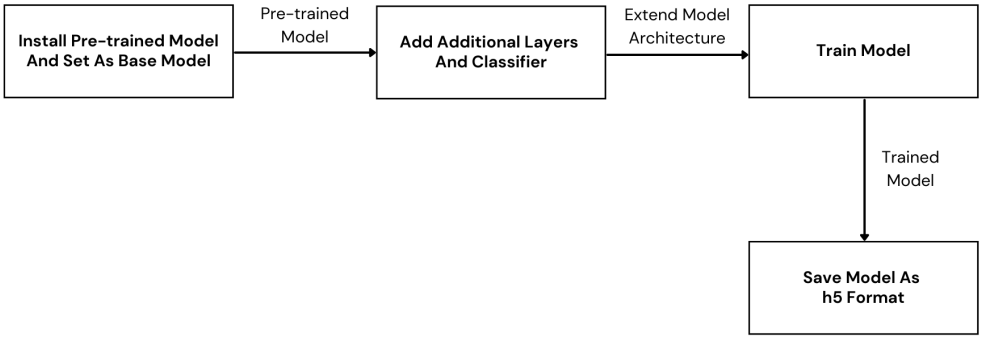


Figure 4.4 Block Diagram of Deploying a Pre-trained Model (Self).

Therefore, the following approach was taken to train the pre-trained image classification models. First, the dataset needs to be pre-processed to prepare it for training. This was done by checking that the images are separated with their corresponding label and also by removing any ambiguous images from the dataset. Next, the images are resized to the required size requested by the models. Moreover, resizing images to a smaller size will require less computational power making the dataset more efficient and faster to train. Afterwards, the pre-trained models are deployed into the environment using the Keras library. These image classification models are made of two main components: the base model and the classifier. The base model consists of the pre-trained model whilst the classifier contains the classification layers. The Keras library offers a vast number of pre-trained models. However, since we will be using the same models used in [42] the chosen models are the VGG16, VGG19, and Inception_V3. Selecting the best model requires analysing the accuracy and latency of the model's predictions and a balance between the two must be found. Moreover, these models were trained using supervised learning on the ImageNet dataset, which consists of 1.2 million labelled images split up into 1000 different classes. In the classification section, we made use of two dense layers which were added after the base model. Furthermore, we also included a dropout layer to help the models overcome overfitting. The final dense layer, which can also be referred to as the output layer consists of 5 neurons with a sigmoid activation function. This enables the model to calculate the probabilistic value associated with each class for our dataset.

When implementing the models it was noted that the models were using a lot of computational resources, resulting in long training times. We noticed that we were updating the weights of the added layers as well as the base model. Whilst updating all the layers to fit the dataset is the most optimal training method, it requires a lot of computational power and results in long training times. Therefore, we opted to only update the added layers and freeze the layers of the base model. Early-stopping was also used on the models during training to stop them from overfitting to the dataset. This was implemented by using the Keras early stopping function which monitors the validation accuracy during training. If the validation accuracy of the model does not improve after 2 epochs the model will stop training and save the weights of the last epoch. Finally, each trained model was then saved as an h5 file which we later used to evaluate them.

4.4.1 Classifying Images

The chosen model based on the evaluation will classify an uploaded image if the image has been pre-processed to fit the model specifics. Therefore, before inputting the image in the model it must first be resized and normalized and only then can it be classified. Furthermore, a list containing the class names is required, this will be used to get the respective labels after the model outputs the list of probabilistic values. Once the image is processed and the model outputs the list of probabilistic values for all the classes, the class with the highest value will be saved. The predicted class name will be later transformed into textual data as previously mentioned. This process can be seen in Figure 4.5

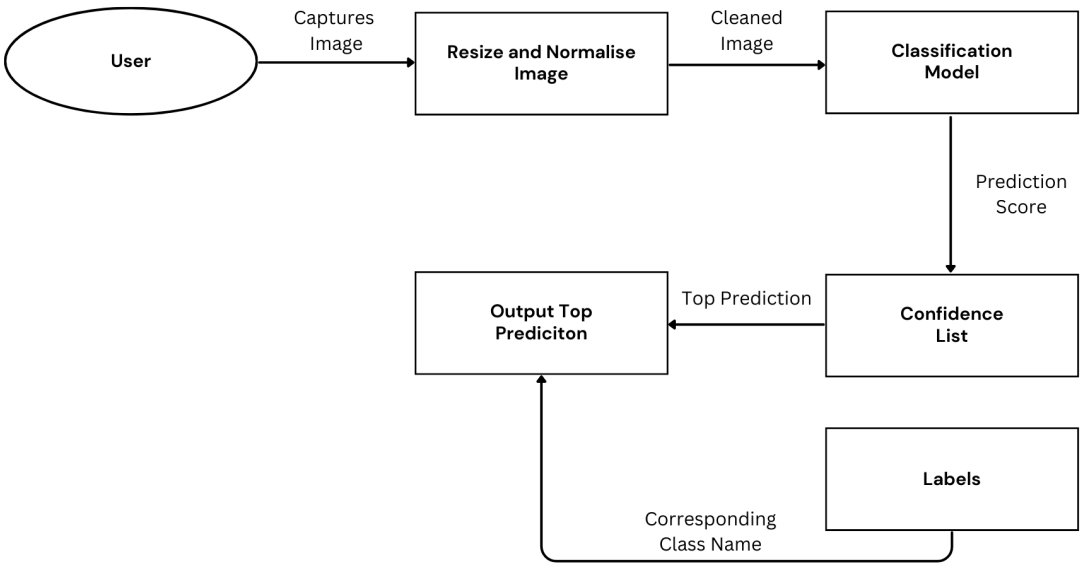


Figure 4.5 Block Diagram for Image Classification System (Self).

4.4.2 Image Dataset

The most used dataset when it comes to indoor room classification is the Monk House classification dataset which was introduced by Quattoni and Torralba in [47]. The dataset contains around 45436 images divided into 7 different categories, exterior, bedroom, kitchen, living_room, interior, bathroom, and dining_room. Another indoor room dataset is the House Rooms Image Dataset [48], containing 5250 images and is divided into 5 different categories, Bathroom, Bedroom, Dining, Kitchen, and Living Room. Furthermore, the House Rooms Image Dataset can be installed locally on a computer, this allows the removal of ambiguous images to be removed much more easily. Therefore, due to the ease of use, the House Rooms Image Dataset offers, and the fact that it is already split as desired, this dataset was chosen to be used on the image classification model. After installing the dataset it was noted that each class had a different amount of images. These could easily affect the accuracy of the models since the models would be biased toward the classes that have a higher amount of images. Therefore we lowered each class to the smallest amount found, which amounted to 606 images and subtracted a further 100 images to be used in Section 4.3.4, bringing the total amount of images in the dataset to 2530. These are then divided into 70% training data, 20% validation data, and 10% testing. Appendix A.2 includes some images from the dataset.

4.5 Word Prediction

As mentioned in Section 3.1.1, currently state-of-the-art word prediction systems make use of LLMs which also give out the best prediction accuracy. Currently, MLMs are popular with researchers due to their capability and accessibility. Four different MLMs have been outlined to fit perfectly for this study which are: BERT, RoBERTa, GPT and ELECTRA. These four models are available in the Python environment using the TensorFlow or PyTorch framework. Focusing on the research done in [19] the results showed that the RoBERTa model had better accuracy on a vast amount of different NLP tasks when compared to GPT and BERT. Another study [21], introduced a new model called ELECTRA which in it the authors found similar results to the RoBERTa model in various NLP issues. However, due to the reasons mentioned in Section 4.4.4, it would be unwise to use the GPT model and compare it with other language models. Therefore, due to the mentioned reasons, we will be testing the word prediction accuracy on the BERT, RoBERTa, and ELECTRA models.

Deploying the Language Models with the speech recognition and image classification model requires numerous steps as can be seen in Figure 4.6. First, we have to set up the language model and its tokeniser, this was achieved by using the TensorFlow framework

since it contains the required installations. Next, the user needs to input an image of the current environment, record the current conversation spoken with another person, and input the text. The image will first be processed as shown in Figure 4.3 and once the prediction is given we will transform the prediction into textual data as shown in Sentence 4.1. Afterwards, the recorded speech will be processed as shown in Figure 4.2 and will be inserted with the image classification textual data as shown in Sentence 4.2. Finally, the user input will also be combined with the transformed textual data and a <mask> token will be added to the end of the sentence as shown in Sentence 4.3. In the context of word prediction, the mask token is used to task the model with predicting the missing word. Therefore, the final sentence which will be inputted into the LM would take the form as shown in Sentence 4.3. Finally, the model outputs the word with the highest probability of occurring based on the previous words. When implementing this process it was noted the LMs would sometimes output punctuation marks and other special tokens such as the end-of-sequence token. To combat this we started checking if the output of the language models consisted of these punctuation marks or any of the special tokens. If this was the case we retrieved the next highest probabilistic token until a word token was found.

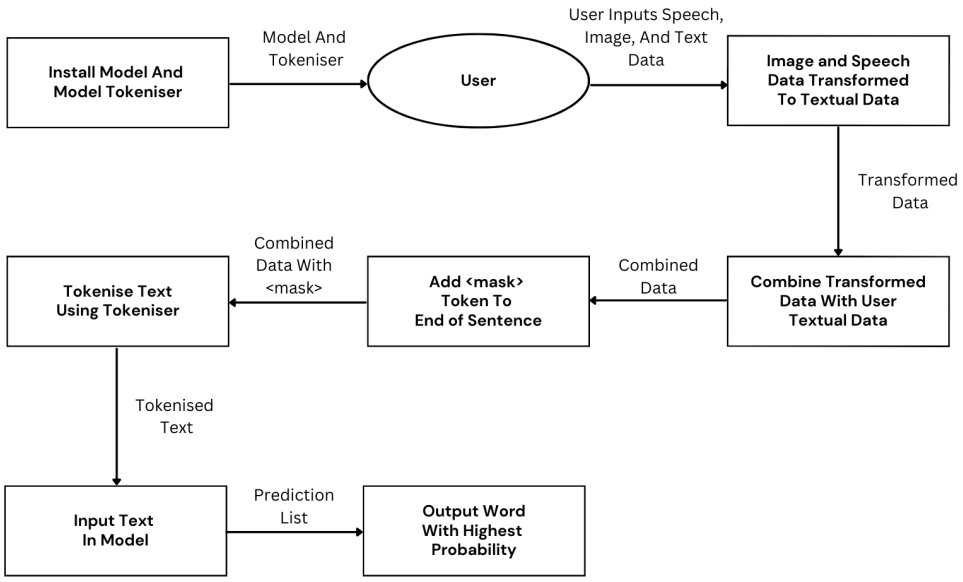


Figure 4.6 Block Diagram of the Word Prediction System (Self).

Input sentence structure:

(4.1) The User is currently in *<predicted class>*

(4.2) The User is currently in <predicted class> + *<speech text>*

(4.3) The User is currently in <predicted class> + < speech text> + *<user text>* + *<mask>*

4.6 MultiModal Dataset

Since the system makes use of speech, image, and text data, a customised dataset containing the mentioned data was created since no similar dataset was found. In [20], the authors also created a customised dataset to evaluate and compare their word prediction system with other solutions, which contained around 100 sentences. Therefore, we decided to mimic this dataset by creating 100 sentences for each scenario corresponding to the indoor rooms, bringing the total size of our dataset to 500 rows. Furthermore, we created two separate files containing the image and speech data which were further separated into different sub-folders for the 5 different scenarios. The text data was only separated into five different text files for each scenario and the sentences in the text file are separated by a newline sequence. Finally, each row in the dataset would contain three different values, the first column value would contain the location of the image, the second column value would contain the location of the speech audio and the third and final value would contain the text data directly.

To gather the values for the multimodal dataset we first made use of the House Rooms Image Dataset. We decided to subtract 100 images from each category bringing the total value of the image dataset to 506 images per category. This was done since our classification models utilise transfer learning, which enables them to predict with high accuracy even when a limited amount of data is available. As stated above, the images were then organised into different sub-folders. The speech and text data was created by making use of OpenAI's ChatGPT model. ChatGPT was asked to simulate 100 conversations between two people for each category of the indoor room, we also asked that the conversations being held in the room relate to topics that are normally discussed in that environment. Since the ChatGPT model is being used to create the speech and user input, it would be unwise to use the model or a previous version of it when comparing the effects of added context. This was done since the GPT model may have prior knowledge of the data it is being tested on. Furthermore, we also asked the model to output the conversations in the following format to segment the data much easier: Person 1: *Sentence*, Person 2: *Sentence*. After obtaining the text data, the Windows Voice Recorder application was used to convert the "Person 1" text into audio, this was done by manually inputting the speech using a headset microphone and saving the file as a waveform audio file format. After the speech audio was obtained we finally organised the text, image, and audio data as mentioned beforehand.

4.7 Summary

This chapter provides a comprehensive explanation of the implementation process for the contextualized word prediction system. The chapter also includes an overview of the architecture with a detailed DFD. Additionally, each component of the proposed solution is further analysed by having a dedicated section with block diagrams and the decisions taken based on the reviewed literature. In the next chapter, we will discuss the evaluation strategies to evaluate the classification model and effectiveness of the word prediction system, using the added context. Finally, a discussion on the results obtained will also be highlighted and results will be compared to existing studies.

5 Evaluation

In this section, we will first evaluate the classification models and the effects of transfer learning. Furthermore, we will also compare the results of two research papers by analysing three different speech recognition models. Afterwards, we will compare the use of speech recognition and image classification on improving word prediction accuracy. The results obtained will be discussed and compared to similar studies. Table 5.1 shows the methodology for the evaluation of the techniques used in this study.

Evaluation Methodology	
Image Classification	The image classification models will be evaluated by comparing the training and validation set values, whilst also assessing the model's performance using the F1-score values gained from the test set.
Speech Recognition	For speech recognition the different models will be evaluated from two different research papers using the WER.
Word Prediction	The word prediction will be evaluated using the WPR on each language model. Additionally, the WPR will be assessed while considering the contextual information provided by the speech and image data.

Table 5.1 Evaluation Methodology (Self).

5.1 Image Classification

Since our main aim for the classification model is to be able to distinguish between five different indoor rooms, we did not require a vast amount of training data. As we concluded in the literature review, the best strategy found was of fine-tuning pre-trained models instead of building one from scratch. Nonetheless, a model was still built to showcase the strength of transfer learning. Therefore, experimentation was carried out to seek the best model that predicts with the highest accuracy, and the VGG16, VGG19, and Inception-V3 were used. In Appendix B, supplementary figures are provided, offering a comprehensive overview of the accuracy values for each model in both the training and validation sets, along with the corresponding loss values for the train and validation sets.

5.1.1 Evaluation

To evaluate the performance of the models we will first start by comparing the model's testing and validation accuracy. The most important component is the validation accuracy which tells us the accuracy of the model on unknown data. Additionally, we also made use of the Keras early stopping function, which was used to minimise the models from overfitting by checking the validation loss value in each epoch. If the validation loss does not improve after 2 epochs, the training stops, and the weights of the model are saved.

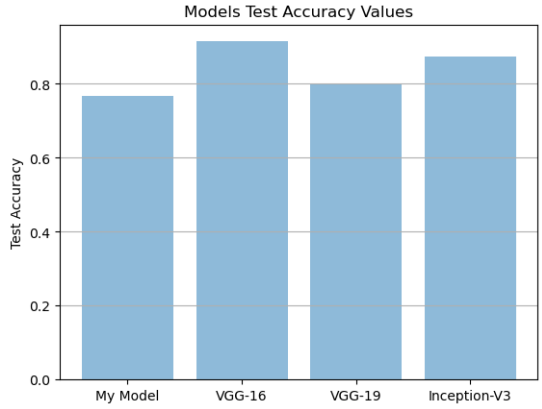


Figure 5.1 Test Accuracy (Self).

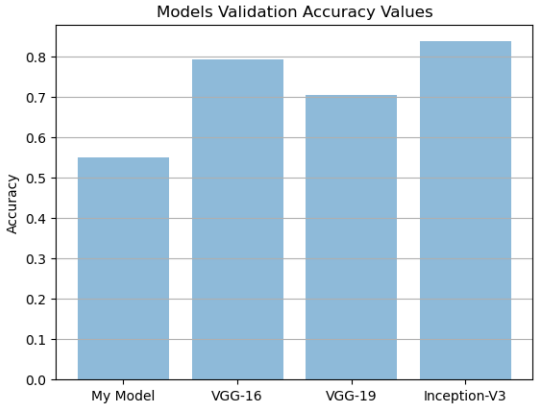


Figure 5.2 Validation Accuracy (Self).

As we can see in Figure 5.1 the VGG-16 outperformed the other models by reaching an accuracy well over 80% on the training data, whilst the VGG-19 and Inception-V3 also reached a respective accuracy close to the VGG-16. The lowest accuracy on the training data can be seen on the model which we built from scratch, whilst the accuracy is close to reaching 80% when compared to the other models it still is a bit behind. When comparing the validation data in Figure 5.2 with the testing data, we can see that the Inception-V3 model has relatively the same accuracy. This means that the model did not overfit to the training dataset at all. Furthermore, the VGG-16 and VGG-19 differed by a slight margin but it is not unusual for image classification models to slightly have lower accuracy on the validation set. However, the model that we built shows a significant difference of around 20% which tells us that the model is overfitting to the training dataset. We will also evaluate the performance of the models using confusion matrices, which summarise the number of correct and incorrect predictions made on our test data set. In multi-class classification problems, the confusion matrix can be represented by an N x N dimension matrix, where N is the number of classes. In this case N=5. Between Figure 5.3 and Figure 5.6 we are able to see the confusion matrices of all the models, with these values we can also calculate the overall Precision(5.1), Recall(5.2) and F1-score(5.3) of the models. Where TP is the number of true positives, FP is the number of false positives, and FN is the number of false negatives.

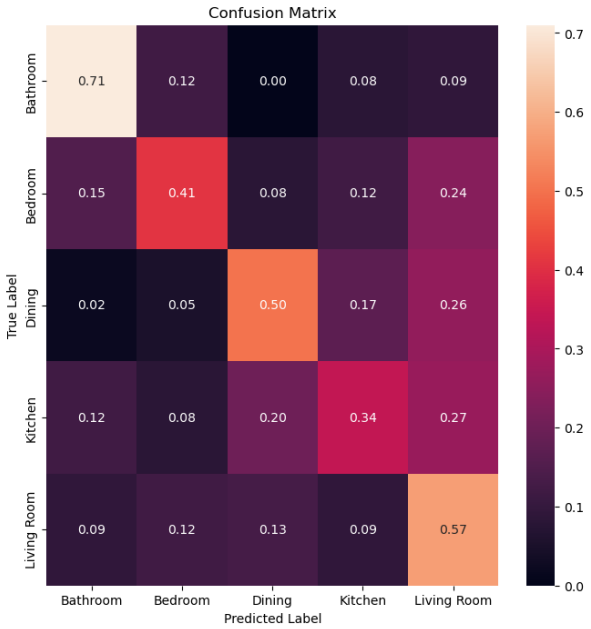


Figure 5.3 My Model Values (Self).

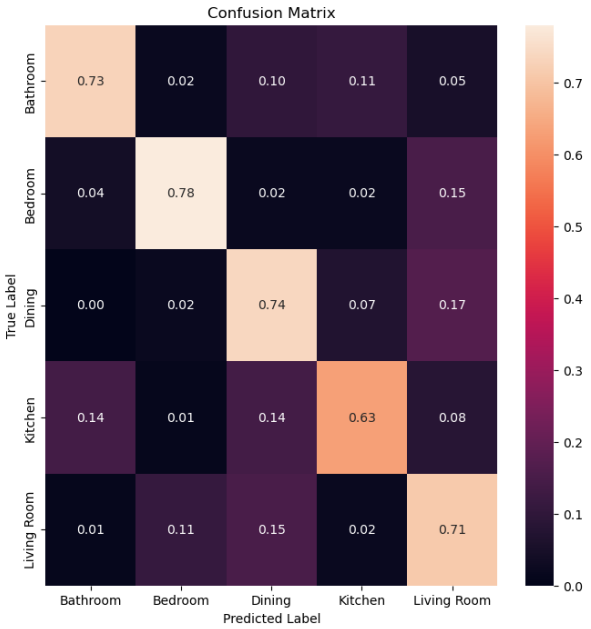


Figure 5.4 VGG16 Values (Self).

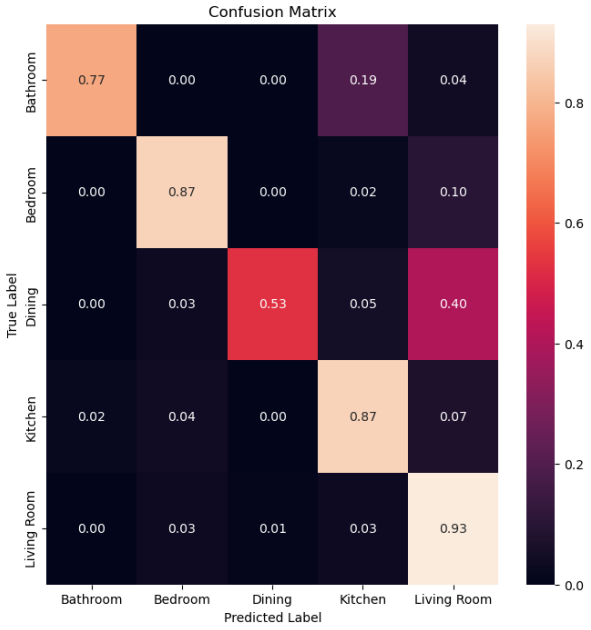


Figure 5.5 VGG19 Values (Self).

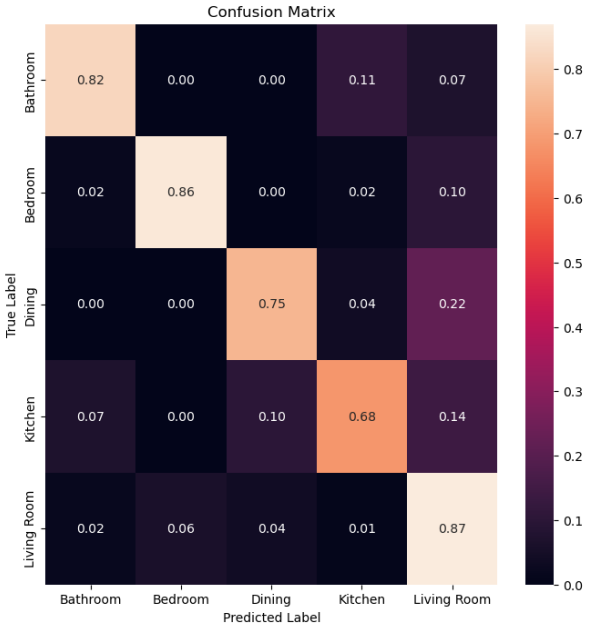


Figure 5.6 InceptionV3 Values (Self).

$$Precision = \frac{TP}{(TP + FP)} \tag{5.1}$$

$$Recall = \frac{TP}{(TP + FN)} \tag{5.2}$$

$$F1 - score = \frac{2 * Precision * Recall}{(Precision + Recall)} \tag{5.3}$$

Model	Precision	Recall	F1-Score
My Model	0.506	0.506	0.498
VGG16	0.626	0.718	0.663
VGG19	0.824	0.736	0.767
Inception-V3	0.773	0.796	0.782

Table 5.2 Comparison of Precision, Recall and F1-Score of Models (Self).

From the results shown in Table 5.2, we also see Inception-V3's ability to achieve the best result in the F1-Score, which is the balance between the precision and recall metrics. Therefore, with these results, we can further conclude that the Inception-V3 is the best model since it can correctly identify positive and negative instances better than the other models making it more robust.

When comparing our results to the study made in [42], we can conclude that the authors received better accuracy prediction, which can be seen in Table 3.2. The authors received better results since they were able to re-train all the layers of the model including the base model, they also made use of a larger dataset. This was not possible from our end since the authors made use of a powerful computer cluster specifically built for large workloads. Furthermore, in their study, the Inception-V3 obtained the least accurate predictions, whilst our results showcased that it was the best model. This is due to the fact that the Inception-V3 architecture is more robust than VGG16 and VGG19 when containing a smaller dataset. However, as mentioned in Section 4.3.2 we must not only consider the accuracy prediction but also the latency of the models. We tested the latency for all the pre-trained models by making use of the test dataset, the results we obtained showed that the Inception-V3 offered the fastest prediction roughly half the time it takes for the VGG19 model. Moreover, the latency of the VGG16 was in between the other two models. From the results obtained and considering the prediction accuracy and latency, the Inception-V3 model outputted the best results.

5.2 Speech Recognition

Our main objective for the speech recognition model is to achieve the highest possible accuracy in identifying the user's speech. Therefore, our study focused on employing a pre-trained model exclusively developed for the English language. As outlined in Section 3.2.1, the Azure model demonstrated superior performance in terms of WER. However, in Section 4.3, we explored alternative models for potential integration into the system.

5.2.1 Evaluation

Whilst the primary focus of the studies lies in word prediction, in this section different results from various papers will be evaluated to provide a comprehensive overview of speech recognition. The different speech models were evaluated using the WER. Where S is the number of word substitutions, I is the number of word insertions, D is the number of word deletions, and N is the total number of words (5.4).

$$WER = \frac{S + I + D}{N} \tag{5.4}$$

Model	**Word Error Rate**		
	Xu Binbin		**Besim Alibegovic**
	Normal Data(%)	**Noisy Data(%)**	**Normal Data (%)**
Google	14.29	20.00	13.85
Microsoft	9.29	11.11	6.67
IBM	14.81	29.63	19.49

Table 5.3 Comparing WER results of [28] and [30].

Based on the findings reported in [28] and [30] which can be seen in Table 5.3, we can conclude that the Microsoft Azure model constantly shows the lowest WER across both normal and noisy data. This indicates that the model is more robust in handling noisy data and recognises words better. Nevertheless, as mentioned in Section 4.3, the Azure model offers limited usage per account, whilst Google's speech recognition model offers free limited usage per day. To ensure that the system works for an extended amount of time it was decided to adopt Google's model despite its relative inferior performance when compared to the Azure model.

5.3 Word Prediction

Since our main aim for the contextualised word prediction system is to have the best LM for word prediction, the models will be tested on word prediction accuracy by using the WPR as done in [20]. Furthermore, as we concluded in the literature review MLMs will be used in this study and will carry experimentation on the following models; RoBERTa, ELECTRA, and BERT.

5.3.1 Evaluation

To evaluate the word prediction accuracy of the LMs we will be making use of the multimodal dataset which we specifically created to evaluate these models. Furthermore, to test the accuracy of the models we will be using the WPR(5.5) which measures the percentage of times the model is able to correctly predict the next word in a sentence.

$$WPR = \frac{total\ number\ of\ correct\ word\ predictions}{total\ number\ of\ words\ predicted} * 100 \tag{5.5}$$

To fully exploit the dataset and to also mimic how the system would be used in real-life scenarios, the models were tested by inputting one word at a time of the user's input text and predict the next word. By utilising this approach, the model would be tested as if it was being used in real-time where in certain scenarios it would have little information on the local context. Furthermore, to better understand the effects of the contextual information on the WPR, the models were tested using the different possible combinations as can be seen in Table 5.4.

Model	**Context**			
	WPR Without (%)	**WPR Image (%)**	**WPR Speech (%)**	**WPR Speech and Image (%)**
RoBERTa	8.00	11.57	14.65	18.60
ELECTRA	2.90	3.90	4.20	4.50
BERT	1.07	1.25	1.34	1.60

Table 5.4 Comparison of RoBERTa, ELECTRA, and BERT on word prediction using additional context (Self).

From the results obtained in this study, we can conclude that the LMs had a hard time predicting words. However, these results can be explained. First of all, the LMs are sometimes being tested to predict words with little local context. Let us take a normal input using both speech and image context. If the LM encounters: "User currently in the kitchen. Are you hungry? <mask>", it would certainly have a hard time predicting the next word since there is not much local context. However, if the LM encounters: "User currently in the kitchen. Are you hungry? I would like something to <mask>", it would be much easier for the model to predict the word *eat* since it has both local and broad context. Furthermore, the models were not fine-tuned on the dataset and we wanted to test if the models are able to perform well on unseen data.

Nevertheless, the results from the study still show positive results. All three models showed improvements in WPR when making use of the added context especially when using both speech and image context. Additionally, the speech data had the greatest impact on the WPR, as it contains more meaningful context that can significantly enhance the WPR. From the results obtained, we can conclude that the RoBERTa model achieved the best results and by using the speech and image context it gained a WPR of 18.60%. RoBERTa's superior performance is due to its training process, in which it is trained on a larger amount of text data for a longer period of time and uses a more sophisticated masking strategy during training when compared to the other models. These factors contribute to the results obtained from this study.

5.4 Summary

In this section, an overview of the results obtained from this study are mentioned and compared with related studies. Furthermore, the evaluation methods used to test the different models are also briefly explained. In the next section, we will discuss if the results obtained and the methodology used accomplished the different objectives. Moreover, we will also mention future work that can be done to improve the system.

6 Conclusion

In this study, we identified the lack of context in word prediction systems. We introduced a novel approach that makes use of speech and image data to enhance word prediction by introducing additional context using the mentioned data. From the reviewed literature, similar systems exist however they focus on improving other NLP tasks such as image captioning and text summarisation. As stated in Section 1.3 the following objectives were identified:

O1 Identify the current indoor room the user is in by using a deep learning model.

O2 Identify user speech using a deep learning model, which converts the speech into text as output.

O3 Create a multimodal dataset containing images of indoor rooms, recorded speech, and text data.

O4 Evaluate the effectiveness of deploying objectives 1 and 2 to a language model to improve word prediction.

The first objective (**O1**), was to develop a deep learning model capable of identifying indoor rooms. This was accomplished by implementing a DNN model capable of classifying an image to its corresponding label. Different pre-trained models were tested, mainly the VGG16, VGG19, and Inception-V3 models as done in [42]. From the results, we concluded that the Inception-V3 model gave the best results in accuracy as well in F1-Score. This could be due to the architecture of the Inception-V3 model being more robust when compared to the VGG16 and VGG19 models. Furthermore, a model was built from scratch to see the effectiveness of transfer learning, we concluded the same as Salman et al. [38], in which pre-trained models give better accuracy with less data and training time. Whilst we did not achieve the same results as Othoma et al. [42] we still obtained an accuracy of 84% on the Inception-V3.

The second objective (**O2**), was to deploy a deep learning model capable of identifying user speech and transforming it into text. This was accomplished by implementing Google's STT API and using the py_audio library. The py_audio library records and saves the speech which is then used by Google's STT to transform it into textual data.

The third objective (**O3**), was to create a multimodal dataset containing images of indoor rooms, recorded speech, and text data. The objective was accomplished by first obtaining images from the House Room Image Dataset which were separated into different folders according to their class. Next, ChatGPT was used to create conversations between two individuals based on topics related to the indoor rooms. The text of the second person was saved into different text files according to the label. Whilst, the text of the first person was transformed into audio using the Windows Voice Recorder application, each audio file was saved in the appropriate folder based on the label. Moreover, the mulitmodal dataset was built by having the location of the image and audio in each row with the corresponding user text input.

The fourth objective (**O4**), was to test the effectiveness of word prediction accuracy by adding objectives O1 and O2 and testing it on the multimodal dataset created in O3. The objective was accomplished by utilising three different language models and testing whether adding additional context improves their WPR by using the multimodal dataset created in the third objective. Before the data is inputted into the language model the speech and image data are also transformed into textual data using the first and second objectives. Furthermore, once the classifier predicts the label for an image, the prediction is transformed into a meaningful sentence. The additional context was tested on the RoBERTa, ELECTRA, and BERT models. The results of the experiment showed that the RoBERTa achieved the highest accuracy of 18.60% whilst using the additional context gained from speech and image data. Furthermore, we can also conclude that both the speech and image data improved the WPR for all the models. However, the most significant improvement was observed as a result of utilising the speech data. When comparing our results to Yu et al. [20], we achieved lower WPR, however, our results still showed improvement in WPR when adding additional context which is the main aim of this study.

6.1 Future Work

In order to provide additional context, the image classification model can be enhanced by adding more labels. Future research should aim to deploy a model capable of identifying additional environments, with this the system can gain further context of the surrounding. Additionally, future work could also incorporate object detection to gain additional context by identifying the objects currently being used by the user or the individual with whom the user is communicating. This added context should lead to further accurate predictions since it leads to a deeper understanding of the situation. The usability of the system can be improved by developing a mechanism capable of continuously capturing image data instead of interrupting the user to capture images. This allows the system to be more convenient for the user and also allows for faster communication. Future research should focus on enhancing the system by allowing continuous speech input. The system should be able to automatically cease the recording once the speech has ended similar to virtual assistants such as Siri. This feature eliminates the need for the user to stop the recording manually and thus enhances the system in terms of user experience. To increase the word prediction accuracy of the system, future work could incorporate a personalised knowledge base specific to each user. The knowledge base should contain information about the user's routine and calendar. Furthermore, the knowledge base should also contain the input text of the user which can be used to train the system. By further training the system on the user input text, the system is able to find patterns in the writing style and display predictions which closely align with the user. Moreover, the system's capability can be expanded by implementing it into an augmentative and alternative communication device. Future work should analyse the effectiveness of the system in terms of keystroke savings and communication efficiency when integrated with such devices. The research would provide valuable data on the benefits of utilising such systems and further justify the research done in this study.

References

[1] U. Khandelwal, H. He, P. Qi, and D. Jurafsky, "Sharp nearby, fuzzy far away: How neural language models use context," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 284–294. DOI: 10.18653/v1/P18-1027. [Online]. Available: https://aclanthology.org/P18-1027.

[2] A. Bapna *et al.*, "Slam: A unified encoder for speech and language modeling via speech-text joint pre-training," *arXiv preprint arXiv:2110.10329*, 2021.

[3] S. Shaikh, S. M. Daudpota, A. S. Imran, and Z. Kastrati, "Towards improved classification accuracy on highly imbalanced text dataset using deep neural language models," *Applied Sciences*, vol. 11, no. 2, p. 869, 2021.

[4] J. Zou, Y. Han, and S.-S. So, "Overview of artificial neural networks," in *Artificial Neural Networks: Methods and Applications*, D. J. Livingstone, Ed. Totowa, NJ: Humana Press, 2009, pp. 14–22. DOI: 10.1007/978-1-60327-101-1_2.

[5] A. Mathew, P. Amudha, and S. Sivakumari, "Deep learning techniques: An overview," *Advanced Machine Learning Technologies and Applications: Proceedings of AMLTA 2020*, pp. 599–608, 2021.

[6] C. Janiesch, P. Zschech, and K. Heinrich, "Machine learning and deep learning," *Electronic Markets*, vol. 31, no. 3, pp. 685–695, 2021.

[7] L. Zhuang, L. Wayne, S. Ya, and Z. Jun, "A robustly optimized BERT pre-training approach with post-training," in *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, Huhhot, China: Chinese Information Processing Society of China, Aug. 2021, pp. 1218–1227. [Online]. Available: https://aclanthology.org/2021.ccl-1.108.

[8] S. Karpagavalli and E. Chandra, "A review on automatic speech recognition architecture and approaches," *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 9, no. 4, pp. 393–404, 2016.

[9] S. Alharbi *et al.*, "Automatic speech recognition: Systematic literature review," *IEEE Access*, vol. 9, pp. 131 858–131 876, 2021.

[10] S. Feng, O. Kudina, B. M. Halpern, and O. Scharenborg, "Quantifying bias in automatic speech recognition," *arXiv preprint arXiv:2103.15122*, 2021.

[11] S. Tripathy and R. Singh, "Convolutional neural network: An overview and application in image classification," in *Proceedings of Third International Conference on Sustainable Computing: SUSCOM 2021*, Springer, 2022, pp. 145–153.

[12] M. Suzuki, N. Itoh, T. Nagano, G. Kurata, and S. Thomas, "Improvements to n-gram language model using text generated from neural language model," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2019, pp. 7245–7249.

[13] P. Budzianowski and I. Vulić, "Hello, it's GPT-2 - how can I help you? towards the use of pretrained language models for task-oriented dialogue systems," in *Proceedings of the 3rd Workshop on Neural Generation and Translation*, Hong Kong: Association for Computational Linguistics, Nov. 2019, pp. 15–22. DOI: 10.18653/v1/D19-5602. [Online]. Available: https://aclanthology.org/D19-5602.

[14] T. Brown *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[15] M. Shoeybi, M. Patwary, R. Puri, P. LeGresley, J. Casper, and B. Catanzaro, "Megatron-lm: Training multi-billion parameter language models using model parallelism," *arXiv preprint arXiv:1909.08053*, 2019.

[16] D. Paperno *et al.*, "The LAMBADA dataset: Word prediction requiring a broad discourse context," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 1525–1534. DOI: 10.18653/v1/P16-1144. [Online]. Available: https://aclanthology.org/P16-1144.

[17] B. Shlegeris, F. Roger, L. Chan, and E. McLean, "Language models are better than humans at next-token prediction," *arXiv preprint arXiv:2212.11281*, 2022.

[18] K. Song, X. Tan, T. Qin, J. Lu, and T.-Y. Liu, "Mass: Masked sequence to sequence pre-training for language generation.," in *ICML*, K. Chaudhuri and R. Salakhutdinov, Eds., ser. Proceedings of Machine Learning Research, vol. 97, PMLR, 2019, pp. 5926–5936. [Online]. Available: http://dblp.uni-trier.de/db/conf/icml/icml2019.html#SongTQLL19.

[19] J. Salazar, D. Liang, T. Q. Nguyen, and K. Kirchhoff, "Masked language model scoring," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online: Association for Computational Linguistics, Jul. 2020, pp. 2699–2712. DOI: 10.18653/v1/2020.acl-main.240. [Online]. Available: https://aclanthology.org/2020.acl-main.240.

[20] S. Yu, N. Kulkarni, H. Lee, and J. Kim, "On-device neural language model based word prediction," in *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, 2018, pp. 128–131.

[21] K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, "Electra: Pre-training text encoders as discriminators rather than generators," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*, OpenReview.net, 2020. [Online]. Available: https://openreview.net/forum?id=r1xMH1BtvB.

[22] U. Ayvaz, H. Gürüler, F. Khan, N. Ahmed, T. Whangbo, and A. Bobomirzaevich, "Automatic speaker recognition using mel-frequency cepstral coefficients through machine learning," *CMC-COMPUTERS MATERIALS & CONTINUA*, vol. 71, no. 3, 2022.

[23] A. V. Haridas, R. Marimuthu, and V. G. Sivakumar, "A critical review and analysis on techniques of speech recognition: The road ahead," *International Journal of Knowledge-Based and Intelligent Engineering Systems*, vol. 22, no. 1, pp. 39–57, 2018.

[24] D. Wang, X. Wang, and S. Lv, "An overview of end-to-end automatic speech recognition," *Symmetry*, vol. 11, no. 8, p. 1018, 2019.

[25] A. B. Nassif, I. Shahin, I. Attili, M. Azzeh, and K. Shaalan, "Speech recognition using deep neural networks: A systematic review," *IEEE access*, vol. 7, pp. 19 143–19 165, 2019.

[26] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," *arXiv preprint arXiv:2212.04356*, 2022.

[27] F. Filippidou and L. Moussiades, "A benchmarking of ibm, google and wit automatic speech recognition systems," in *Artificial Intelligence Applications and Innovations: 16th IFIP WG 12.5 International Conference, AIAI 2020, Neos Marmaras, Greece, June 5–7, 2020, Proceedings, Part I 16*, Springer, 2020, pp. 73–82.

[28] B. Alibegović, N. Prljača, M. Kimmel, and M. Schultalbers, "Speech recognition system for a service robot-a performance evaluation," in *2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, IEEE, 2020, pp. 1171–1176.

[29] B. Shi, W.-N. Hsu, and A. Mohamed, "Robust self-supervised audio-visual speech recognition," *arXiv preprint arXiv:2201.01763*, 2022.

[30] B. Xu, T. Chongyang, Y. Raqui, and S. Ranwez, "A benchmarking on cloud based speech-to-text services for french speech and background noise effect," in *Conférence Nationale sur les Applications Pratiques de l'Intelligence Artificielle (APIA 2021)*, Bordeaux, France: hal-03874256, Jul. 2021, pp. 102–107.

[31] L. Cai, J. Gao, and D. Zhao, "A review of the application of deep learning in medical image classification and segmentation," *Annals of translational medicine*, vol. 8, no. 11, 2020.

[32] M. Khoshdeli, R. Cong, and B. Parvin, "Detection of nuclei in h&e stained sections using convolutional neural networks," in *2017 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*, IEEE, 2017, pp. 105–108.

[33] M. Sheykhmousa, M. Mahdianpari, H. Ghanbari, F. Mohammadimanesh, P. Ghamisi, and S. Homayouni, "Support vector machine versus random forest for remote sensing image classification: A meta-analysis and systematic review," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 6308–6325, 2020.

[34] S. Koda, A. Zeggada, F. Melgani, and R. Nishii, "Spatial and structured svm for multilabel image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 10, pp. 5948–5960, 2018.

[35] W. Man, Y. Ji, and Z. Zhang, "Image classification based on improved random forest algorithm," in *2018 IEEE 3rd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*, IEEE, 2018, pp. 346–350.

[36] P. Wang, E. Fan, and P. Wang, "Comparative analysis of image classification algorithms based on traditional machine learning and deep learning," *Pattern Recognition Letters*, vol. 141, pp. 61–67, 2021.

[37] S. O'Gara and K. McGuinness, "Comparing data augmentation strategies for deep image classification," in *IMVIP 2019: Irish Machine Vision & Image Processing*, Technological University Dublin, Dublin, Ireland, Aug. 2019. DOI: 10.21427/148b-ar75.

[38] H. Salman, A. Ilyas, L. Engstrom, A. Kapoor, and A. Madry, "Do adversarially robust imagenet models transfer better?" *Advances in Neural Information Processing Systems*, vol. 33, pp. 3533–3545, 2020.

[39] N. Tripuraneni, M. Jordan, and C. Jin, "On the theory of transfer learning: The importance of task diversity," *Advances in neural information processing systems*, vol. 33, pp. 7852–7862, 2020.

[40] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015.

[41] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[42] K. M. Othman and A. B. Rad, "An indoor room classification system for social robots via integration of cnn and ecoc," *Applied Sciences*, vol. 9, no. 3, 2019. DOI: 10.3390/app9030470.

[43] S. Mascarenhas and M. Agarwal, "A comparison between vgg16, vgg19 and resnet50 architecture frameworks for image classification," in *2021 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON)*, IEEE, vol. 1, 2021, pp. 96–99.

[44] R. Anil *et al.*, "Palm 2 technical report," *arXiv preprint arXiv:2305.10403*, 2023.

[45] L. Jing, Y. Li, J. Xu, Y. Yu, P. Shen, and X. Song, "Vision enhanced generative pre-trained language model for multimodal sentence summarization," *Machine Intelligence Research*, pp. 1–10, 2023.

[46] OpenAI, "Gpt-4 technical report," *arXiv*, 2023.

[47] A. Quattoni and A. Torralba, "Recognizing indoor scenes," in *2009 IEEE conference on computer vision and pattern recognition*, IEEE, 2009, pp. 413–420.

[48] R. Reni, *House rooms image dataset*, Aug. 2020. [Online]. Available: https://www.kaggle.com/datasets/robinreni/house-rooms-image-dataset.

Appendix A

A.1 Model Diagram

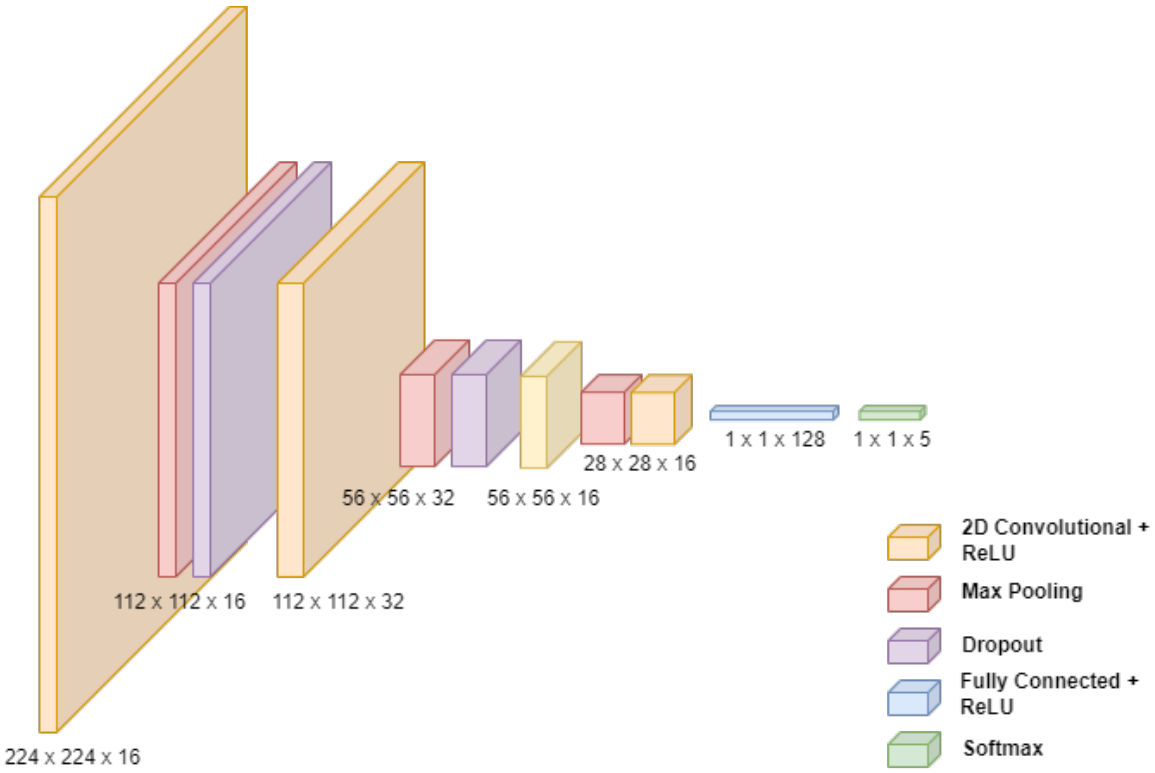


Figure A.1 Model Architecture (Self).

Architecture for the image classification model created for room classification.

A.2 Image Dataset

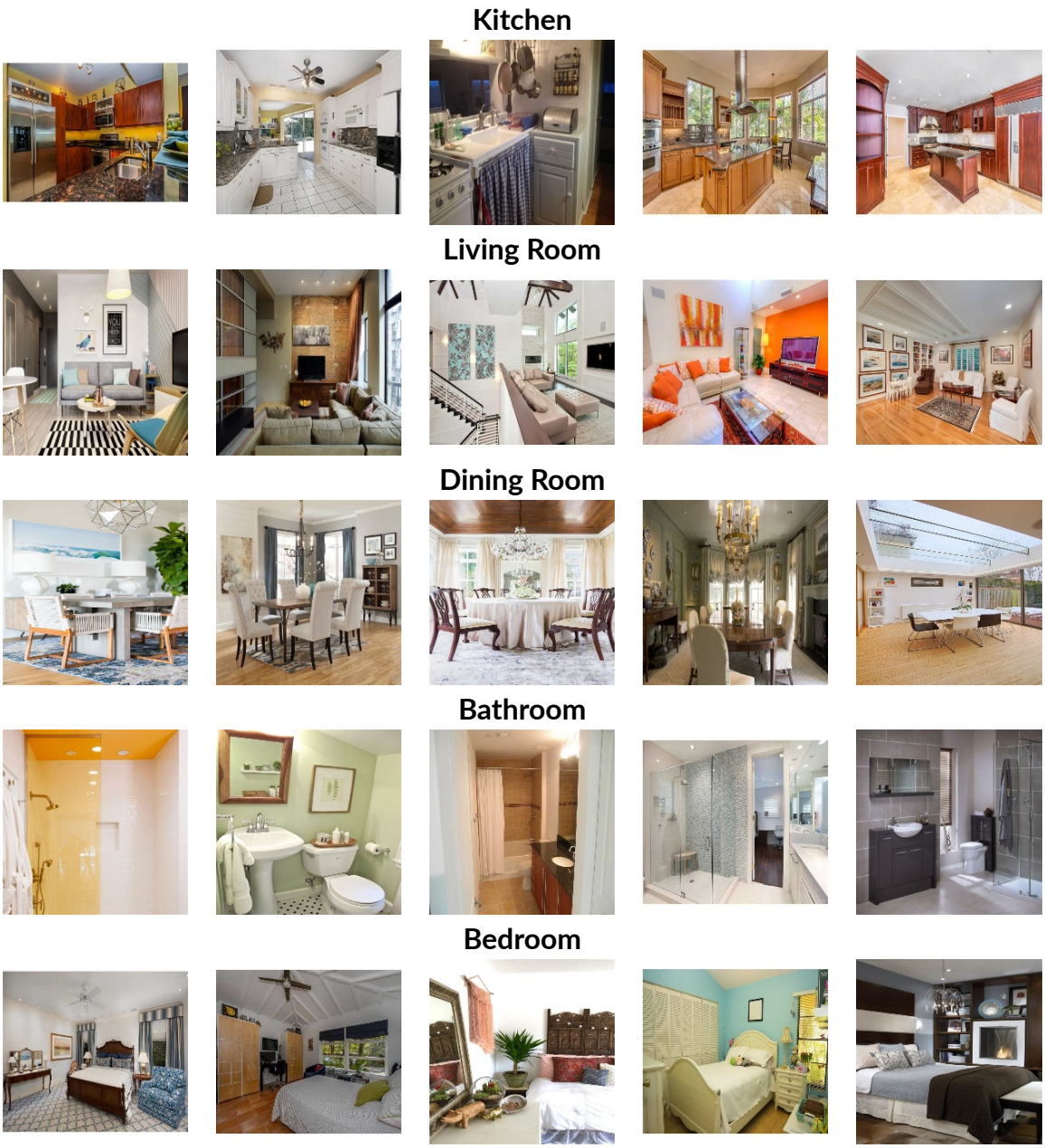


Table A.1 Image Categories [48].

Images from the House Room Image Dataset [48], showcasing the following categories: Kitchen, Bathroom, Bedroom, Living Room, and Dining Room.

Appendix B

B.1 Models Accuracy Values

Accuracy values of the models used for this study including VGG16, VGG19, Inception-V3 and the model built from scratch.

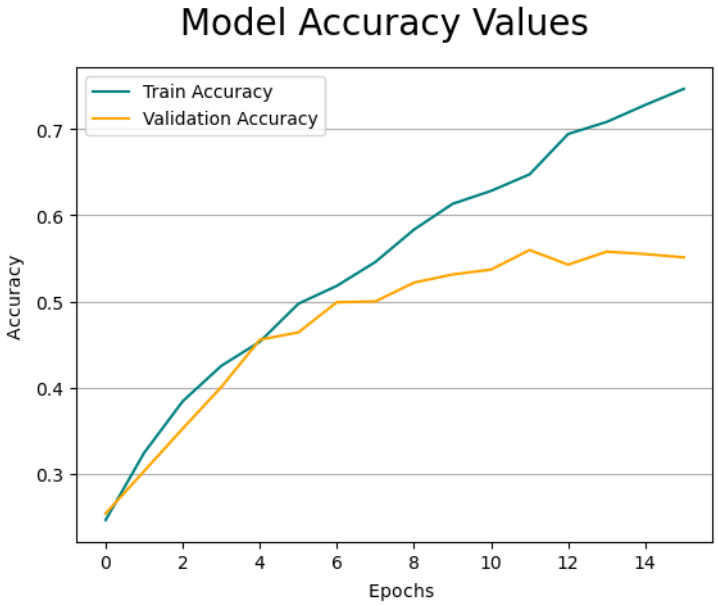


Figure B.1 Model Test and Validation Accuracy Values (Self).

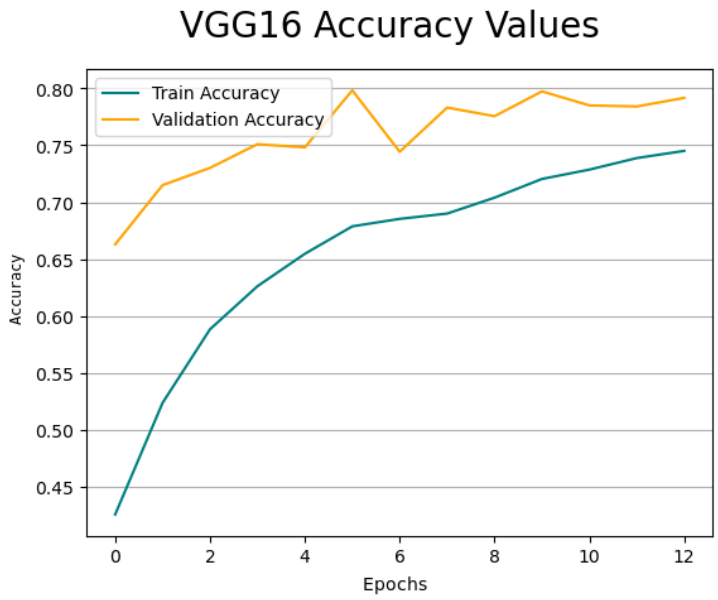


Figure B.2 VGG16 Test and Validation Accuracy Values (Self).

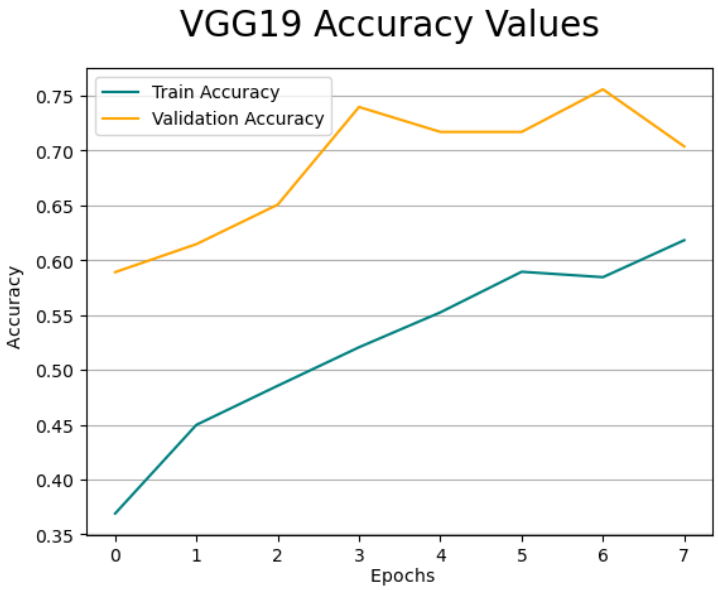


Figure B.3 VGG19 Test and Validation Accuracy Values (Self).

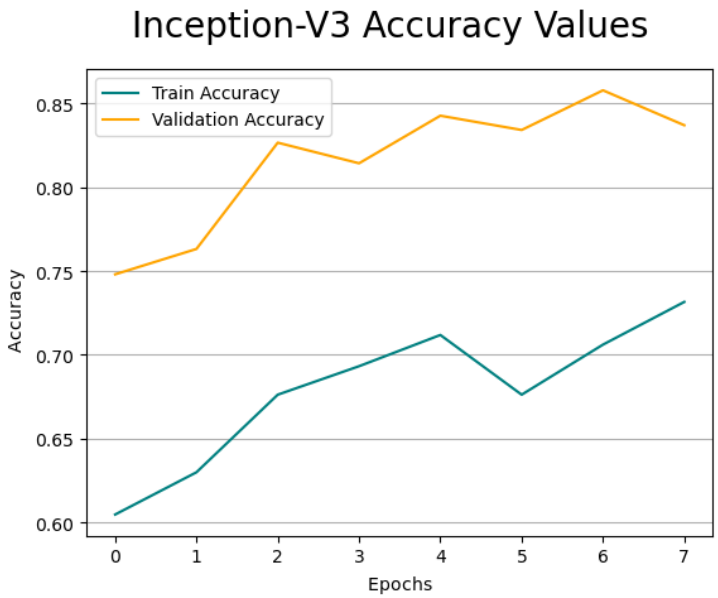


Figure B.4 Inception V3 Test and Validation Accuracy Values (Self).

B.2 Models Loss Values

Loss values of the models used for this study including VGG16, VGG19, Inception-V3 and the model built from scratch.

Figure B.5 Model Test and Validation Loss Values (Self).

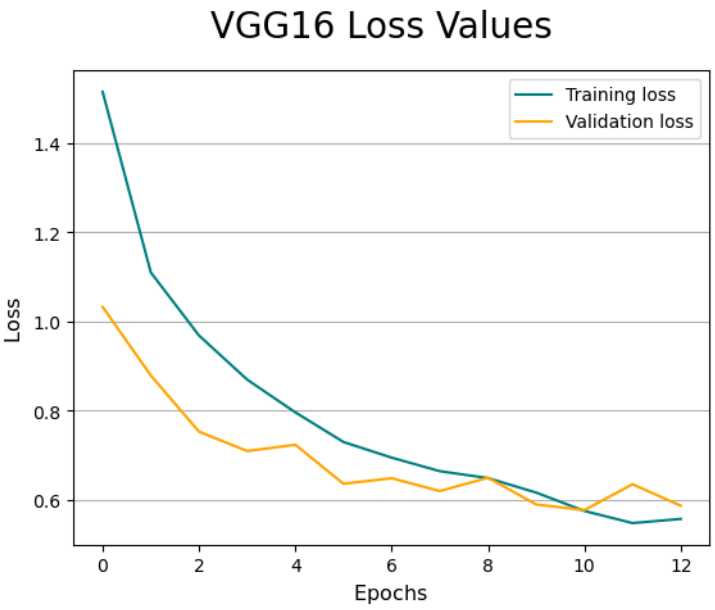


Figure B.6 VGG16 Test and Validation Loss Values (Self).

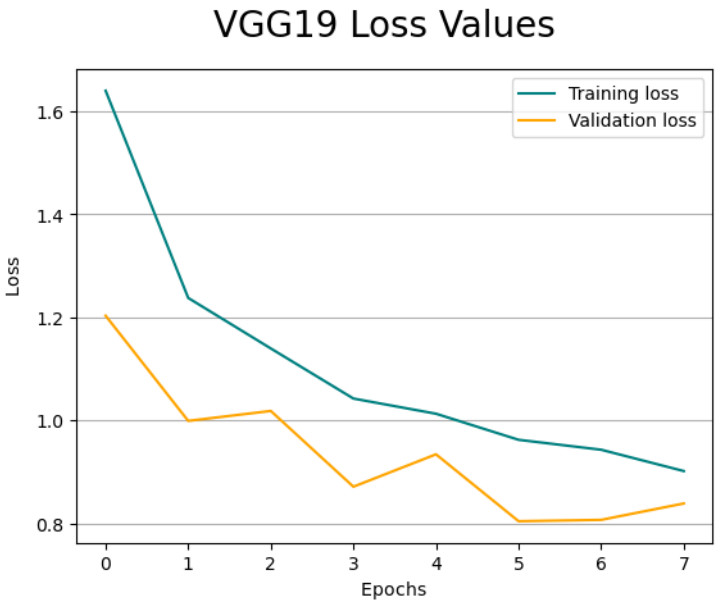


Figure B.7 VGG19 Test and Validation Loss Values (Self).

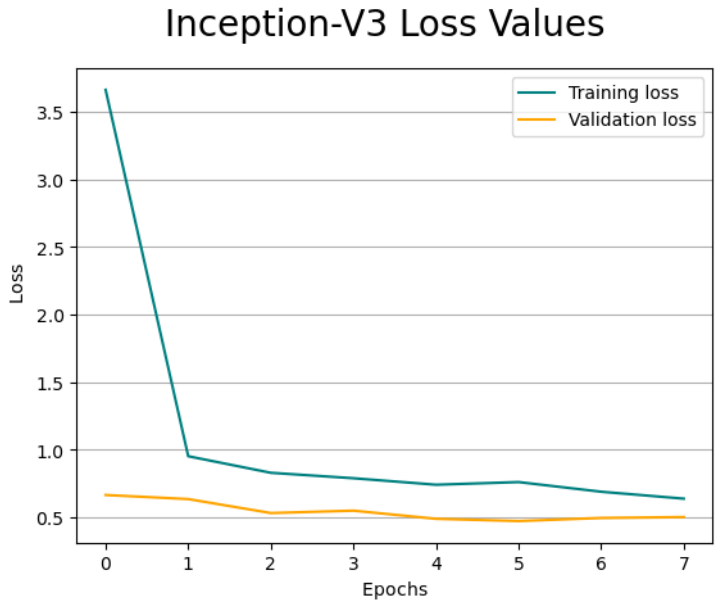


Figure B.8 Inception V3 Test and Validation Loss Values (Self).