

Western Kentucky University

TopSCHOLAR®

---

Masters Theses & Specialist Projects

Graduate School

---

Fall 2020

## Video Game Genre Classification Based on Deep Learning

Yuhang Jiang

Western Kentucky University, yuhang.jiang176@topper.wku.edu

Follow this and additional works at: <https://digitalcommons.wku.edu/theses>



Part of the [Artificial Intelligence and Robotics Commons](#), [Other Physical Sciences and Mathematics Commons](#), and the [Probability Commons](#)

---

### Recommended Citation

Jiang, Yuhang, "Video Game Genre Classification Based on Deep Learning" (2020). *Masters Theses & Specialist Projects*. Paper 3462.

<https://digitalcommons.wku.edu/theses/3462>

This Thesis is brought to you for free and open access by TopSCHOLAR®. It has been accepted for inclusion in Masters Theses & Specialist Projects by an authorized administrator of TopSCHOLAR®. For more information, please contact [topscholar@wku.edu](mailto:topscholar@wku.edu).

VIDEO GAME GENRE CLASSIFICATION BASED ON DEEP LEARNING

A Thesis  
Presented to  
The Faculty of the Department of Mathematics  
Western Kentucky University  
Bowling Green, Kentucky

In Partial Fulfillment  
Of the Requirements for the Degree  
Master of Science

By  
Yuhang Jiang

December 2020

VIDEO GAME GENRE CLASSIFICATION BASED ON DEEP LEARNING

Date Recommended 11/16/2020  
Lukun Zheng Digitally signed by Lukun Zheng  
Date: 2020.11.16 17:22:45 -06'00'

---

Dr. Lukun Zheng, Director of Thesis  
Zhonghang Xia Digitally signed by Zhonghang Xia  
Date: 2020.11.17 08:56:36 -06'00'

---

Dr. Zhonghang Xia  
Nguyen, Thanh Digitally signed by Nguyen, Thanh  
DN: cn=Nguyen, Thanh, ou=Western Kentucky University, ou=Department of  
Mathematics, email=lan.nguyen@wku.edu, c=US  
Date: 2020.11.17 08:45:17 -06'00'

---

Dr. Lan Nguyen  
Melanie A. Autin Digitally signed by Melanie A. Autin  
Date: 2020.11.16 19:02:50 -06'00'

---

Dr. Melanie Autin



Associate Provost for Research and Graduate Education

## ACKNOWLEDGMENTS

First and foremost, I would like to praise and thank God, who has granted countless blessings, knowledge, and opportunities to me, so that I am finally able to accomplish the thesis.

I would like to thank my advisor, Dr. Lukun Zheng, for his encouragement, patience and passion. I feel motivated and encouraged every time I attend his meetings. He also guided me on the research and thesis, providing me suggestions that improved this work.

I also would like to thank my thesis committee, Dr. Zhonghang Xia, Dr. Lan Nguyen and Dr. Melanie Autin for their insightful comments, advice, and support.

I am grateful to my parents, my grandma and my uncle Dr. Rui Zhang who encouraged me when I faced challenges.

## CONTENTS

<b>1. Introduction</b>	<b>1</b>
<b>2. Related work</b>	<b>5</b>
<b>3. Methodology</b>	<b>7</b>
3.1. Deep neural network . . . . .	7
3.2. Image-based models . . . . .	10
3.2.1. MobileNet . . . . .	10
3.2.2. ResNet . . . . .	11
3.2.3. Inception . . . . .	12
3.3. Text-based architectures . . . . .	13
3.3.1. Recurrent neural network . . . . .	13
3.3.2. Universal Sentence Encoder . . . . .	15
3.4. Multi-modal fusion . . . . .	16
<b>4. Experimental study</b>	<b>18</b>
4.1. Dataset . . . . .	18
4.1.1. Preprocessing . . . . .	20
4.2. Experiments . . . . .	22
4.2.1. Results . . . . .	23
<b>5. Analysis</b>	<b>25</b>
<b>6. Conclusion and future work</b>	<b>30</b>
<b>References</b>	<b>30</b>

## LIST OF FIGURES

1.0.1 Some sample video game covers, the genres are a) <i>Sport</i> , b) <i>Shooter</i> , c) <i>Adventure</i> and d) <i>Racing</i> . . . . .	2
1.0.2 Some sample video game covers, the genres are a) <i>Puzzle</i> , b) <i>Adventure</i> , c) <i>Fighting</i> and d) <i>Puzzle</i> . . . . .	3
3.1.1 A feed-forward neural network with one hidden layer . . . . .	8
3.1.2 Relationship between the network, layers, loss function, and optimizer[15]	9
3.2.1 The Structure of Depth-wise Separable Convolution . . . . .	11
3.2.2 Example of a residual block . . . . .	12
3.2.3 Inception module with dimension reductions[20] . . . . .	13
3.3.1 The repeating module in an LSTM[24] . . . . .	14
3.3.2 Structure of an LSTM cell[24] . . . . .	15
3.4.1 Multi-modal fusion architecture . . . . .	17
4.1.1 The original genre distribution . . . . .	19
4.1.2 Modified genre distribution . . . . .	21
5.0.1 Confusion matrix of multi-modal fusion: vertical axis represents the observed genres; horizontal axis represents the predicted genres. . . . .	27
5.0.2 T-SNE plot of video game genre classification . . . . .	29

## LIST OF TABLES

4.1.1	The 21 video game genres . . . . .	18
4.1.2	Samples from the original dataset . . . . .	20
4.1.3	Grouping of the related genres . . . . .	21
4.1.4	Modified samples . . . . .	22
4.2.1	The Top 1 and Top 3 accuracies for the image-based models on the testing set . . . . .	24
4.2.2	The Top 1 and Top 3 accuracies for the text- based models on the testing set . . . . .	24
4.2.3	Multi-modal fusion accuracy with ResNet50 and Universal Sentence Encoder . . . . .	24
5.0.1	Multi-modal fusion accuracy by game genre on the testing set . . . . .	26
5.0.2	Misclassified examples . . . . .	28

# VIDEO GAME GENRE CLASSIFICATION BASED ON DEEP LEARNING

Yuhang Jiang

December 2020

33 Pages

Directed by: Lukun Zheng, Zhonghang Xia, Lan Nguyen and Melanie Autin

Department of Mathematics

Western Kentucky University

Video games have played a more and more important role in our life. While the genre classification is a deeply explored research subject by leveraging the strength of deep learning, the automatic video game genre classification has drawn little attention in academia. In this study, we compiled a large dataset of 50,000 video games, consisting of the video game covers, game descriptions and the genre information. We explored three approaches for genre classification using deep learning techniques. First, we developed five image-based models utilizing pre-trained computer vision models such as MobileNet, ResNet50 and Inception, based on the game covers. Second, we developed two text-based models, using Long-short Term Memory(LSTM) model and the Universal Sentence Encoder model, based on the game descriptions. For the third approach, we constructed a multi-modal fusion model, which concatenates extracted features from one image-based model and one text-based model. We analysed our results and revealed some challenges that exist in the task of genre classification for video games. Some future works are also proposed.



# CHAPTER 1

## INTRODUCTION

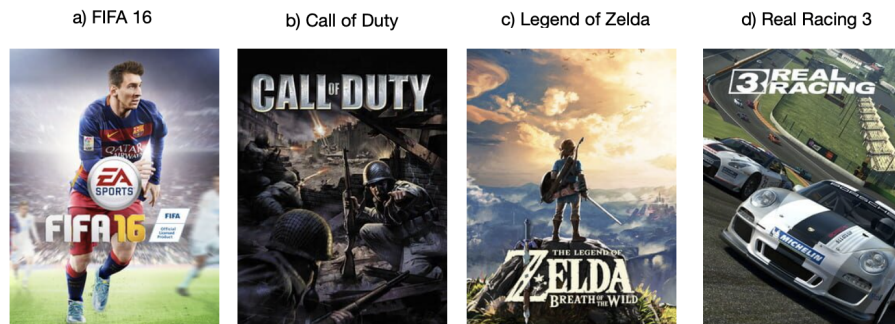
Genres are stylistic categories of literature, music, or other forms of art or entertainment characterized by their forms, contents, subject matters, and the like. Genres are often used to identify, retrieve, and organize works of interest. Genre classification is the process of grouping works together based on certain stylistic similarities. It has been used in a wide range of areas such as music, paintings, film, books, etc.[23, 7, 13] These important tasks were traditionally done manually, which has many limitations on cost, time and other resources. With the growing capacity of computational powers of modern computers, many different automatic genre classification techniques have been developed and used in domains such as movies, books, paintings, etc. In this study, we focus on the task of genre classification of video games based on the cover and textual description using deep learning techniques. To the best of our knowledge, this is the first attempt on automatic genre classification of video games using deep learning techniques.

The rise of video games was only about 50 years ago[19], and it has experienced huge development ever since then. In recent decades, there have been more than 2.5 billion video games from all over the world and the number is still growing.

Every video game can be classified into at least one genre. In general, a video game is released in a box with carefully designed cover art with some text printed on it. Based on the covers and the descriptive texts, we propose three different approaches for genre classification of video games: image-based approach using the game covers, text-based approach using the textual description of the games, and a multi-modal fusion as an approach using both the game covers and the textual description of the games.

The covers and texts aim to convey the major information about the game to the

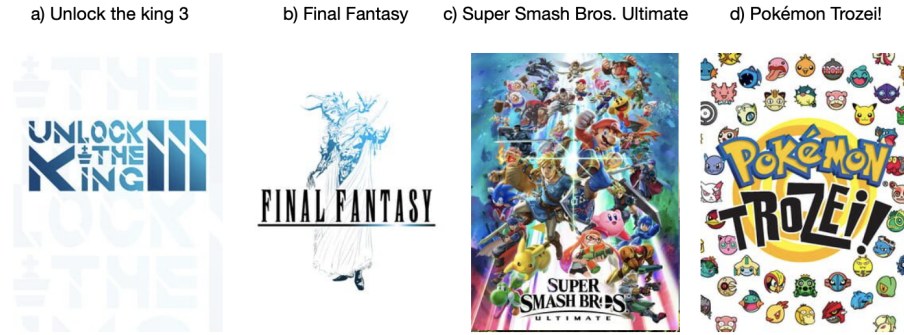
potential consumers. The descriptive text of a video game usually gives an introduction about the gameplay, worldview, characters, etc. Those are important information to determine the genre. Video game covers are special images with some representative patterns and combinations on them, which are designed to make people feel fun, joy and interest to play them. A video game cover often has features related to its genre so that it is attractive to people.



**Figure 1.0.1:** Some sample video game covers, the genres are a) *Sport*, b) *Shooter*, c) *Adventure* and d) *Racing*

Figure 1.0.1 shows a sample of 4 different video games. These cover images present important information about the game content. In Figure 1 a), we can see the soccer player on the cover, which indicates that this game belongs to the genre *Sport*. In Figure 1 b), we can infer that this is a *Shooter* game according to the soldier and weapon on the cover. In Figure 1 c), there's a man standing there and holding a sword. It is closely related to its genre *Adventure*. In Figure 1 d), the game genre is *Racing*, which is represented by the race car on its cover.

There exist several challenges in the study of video game genre classification. First, some game covers are far away from the genres. As in Figure 1.0.2, a) and b) are monotonous, with only a few patterns and text on it, while c) and d) have too many characters and patterns, which make it hard to predict the genre by a computer.



**Figure 1.0.2:** Some sample video game covers, the genres are a) *Puzzle*, b) *Adventure*, c) *Fighting* and d) *Puzzle*

Secondly, in most cases, a video game belongs to more than one genre, which makes it troublesome, even for a human not having played the game, to categorize it into a single category. The third difficulty is that the game genres are not precisely defined. There is no clear distinction between some genres such as *Action* and *Fighting*. Moreover, there are few works on genre classification of video games, and it is hard to find such an already available dataset. To overcome these difficulties, we propose three different approaches using deep learning, which have reached high performances across a wide variety of problems. The applications of deep learning techniques on many visual recognition and categorization tasks can achieve a satisfactory level of performance. The contributions of this study include:

1. We created a large dataset of 50,000 video games that includes games cover images, description text, title text, and genre information.
2. We explored some Convolutional neural networks (i.e., MobileNet, Inception and ResNet) for image-based classification using the video game covers. Moreover, we explored Long-short Term Memory and Universal Sentence Encoder for text-based classification using the descriptions on video games.
3. We merged the best image-based model and the best text-based model together

to construct a multi-modal fusion architecture by concatenating the extracted features from these two different models.

4. A thorough evaluation and analysis of the results is given.

This study can aid the future research based on video games by making the dataset available to the public. It also helps the automatic processing for categorizing the video games, which can be applied in a video game store, making it more efficient for the staff to do such work. This study could also be extended to fields like book covers, movie posters and music album classification.

We discuss some related works in Chapter 2. Chapter 3 presents some deep learning models and methodologies that we will use. In Chapter 4, we will present the experimental study. In Chapter 5, we discuss and analyse the results. Finally, in Chapter 6, we conclude the paper and discuss some future works.

## CHAPTER 2

### RELATED WORK

Genre classification is a deeply explored research subject by leveraging the strength of deep learning. It has been applied in a variety of domains like movies, books and music. However, video game genre classification has drawn little attention in academia.

Some studies classify movie genres based on a variety of attributes. Early attempts at movie poster classification used Naïve-Bayes, C4.5 Decision Tree, and k-Nearest Neighbors as base classifiers with a RAKEL ensemble method[12]. Then, Sung et al. performed the multi-label movie poster classification based on a single movie poster[23]. They used modified ResNet-50, VGG-16 and DenseNet-169. Some other researchers have explored classifying a movie based on trailers[22] and the plot synopsis[2].

Book cover classification is also a popular domain of genre classification. Iwana et al. used pretrained architecture LeNet[17] and AlexNet[1] to find the genre by its cover[13]. Kjartansson et al. explored a variety of CNNs (i.e. SqueezeNet, VGG16 and ResNet50) based on the same dataset[14]. They also applied text CNN on the titles of the books in addition to the cover images. Moreover, Kundu has done the book cover classification using both the image of the cover and text extracted from the cover. They have used pre-trained transfer learning methods on image classification[16] and also applied LSTM and Universal Sentence Encoder for text classification.

Some other researchers have applied genre classification on music. Petrovsk et al. has done music genre classification based on the music albums by obtaining a feature vector representation of the images for exploiting the feature space obtained from its

cover art[18]. Huang et al. detected the musical genre using a RBF kernel support vector machine, k-nearest neighbors, a basic feed-forward network, and an advanced CNN based on the input of raw amplitude data and transformed mel-spectrograms of that raw amplitude data[11].

In the domain of video games, Clarke explored the current affordances and limitations of video game genre with an emphasis on classification theory[5]. Clearwater discussed current state of video game genre theory in more detail[6]. In our study, we propose to focus on genre classification of the video games based on attributes of the video game cover and its descriptive text.

Our first two approaches are to use deep learning models to classify the game genre based on game covers and descriptive texts, respectively, and compare the results. Then we focus on a multi-modal fusion that can concatenate the features extracted from both the images and text. The multi-modal fusion architecture has been used to classify the book genre using the book covers and text extracted from the book[16].

## CHAPTER 3

### METHODOLOGY

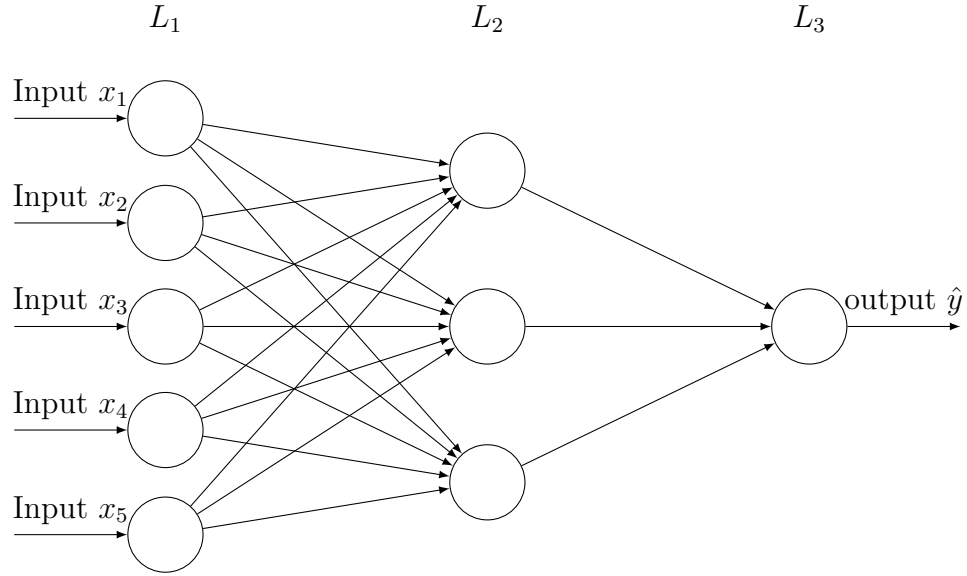
In this chapter, we first briefly elaborate the fundamental background of deep neural network. We also present some image-based and text-based deep learning models for video game genre classification. Finally, we talk about our multi-modal fusion architecture.

#### 3.1 Deep neural network

A deep neural network is the architecture which consists of layers which are connected together. Each layer consists of several neurons, and each neuron has an activation function and computes the activation  $a$  by

$$a = f(W^T X + b),$$

where  $W = \{w_1, w_2, \dots, w_n\}$  is a weight vector,  $X = \{x_1, x_2, \dots, x_n\}$  is the input for this neuron,  $b$  is the bias, and  $f$  is an activation function. The outputs of neurons will be collected and sent to the neuron in the next layer as the input. By repeating this from layer to layer, the network will finally output the prediction. This process is called forward-propagation[16].



**Figure 3.1.1:** A feed-forward neural network with one hidden layer

In Figure 3.1.1, the first layer (input layer, denoted as  $L_1$ ) receives the input  $\{x_1, x_2, x_3, x_4, x_5\}$ . The second layer (denoted as  $L_2$ ) is the hidden layer, and the third layer (output layer, denoted as  $L_3$ ) outputs the prediction  $\hat{y}$ . Suppose the activation of  $i^{th}$  unit in layer  $l$  is denoted as  $a_i^{(l)}$ ; the activations of the hidden layer is computed as:

$$a_1^{(2)} = f(w_{11}^{(1)}x_1 + w_{12}^{(1)}x_2 + w_{13}^{(1)}x_3 + w_{14}^{(1)}x_4 + w_{15}^{(1)}x_5 + b_1^{(2)})$$

$$a_2^{(2)} = f(w_{21}^{(1)}x_1 + w_{22}^{(1)}x_2 + w_{23}^{(1)}x_3 + w_{24}^{(1)}x_4 + w_{25}^{(1)}x_5 + b_2^{(2)})$$

$$a_3^{(2)} = f(w_{31}^{(1)}x_1 + w_{32}^{(1)}x_2 + w_{33}^{(1)}x_3 + w_{34}^{(1)}x_4 + w_{35}^{(1)}x_5 + b_3^{(2)})$$

where  $w_{ij}^{(l)}$  denotes the weight parameter corresponding to the connection between the  $j^{th}$  unit in layer  $l$  and the  $i^{th}$  unit in layer  $l + 1$ , and  $b_i^{(l)}$  denotes the bias of the



$i^{th}$  unit in layer  $l$ . The prediction  $\hat{y}$  is computed by

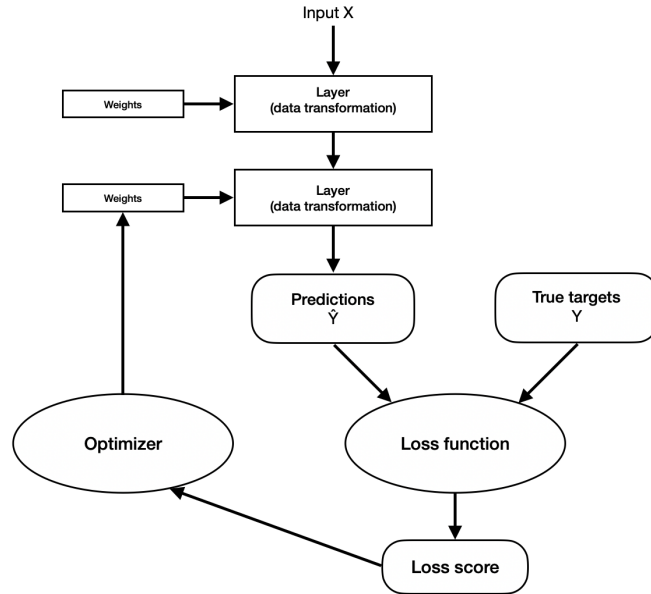
$$\hat{y} = \sigma(w_{11}^2 a_1^2 + w_{12}^2 a_2^2 + w_{13}^2 a_3^2),$$

where  $\sigma$  is the activation function for the output layer.

To train the parameters of a neural network, it first needs to observe and measure how far the output is from our target. The loss function

$$loss = L(\hat{Y}, Y)$$

compares the predictions  $\hat{Y}$  and the targets  $Y$  and yields a loss value measuring how well the predictions match the targets[15]. The loss value will be sent to the optimizer for updating the weights and, therefore, minimizing the loss value. Figure 3.1.2 shows the relationship between the network, layers, loss function, and optimizer.



**Figure 3.1.2:** Relationship between the network, layers, loss function, and optimizer[15]

## 3.2 Image-based models

Convolutional neural networks (CNNs) are some special networks that can be used for computer vision tasks. CNNs are feed-forward neural networks which have hidden layers called convolutional layers.

In the convolutional layers, units are spacially arranged to form matrices called filters. When the convolutional layer receives the input (e.g., images), it will slide the filter over the input itself to detect some patterns displayed on the image. Suppose the filter is a  $n \times n$  matrix  $\mathbf{F} = f_{i,j}$ , and it will be sliding over each  $n \times n$  pixel matrix  $\mathbf{P} = p_{i,j}$  of the image[16]. The convolution of the filter and the pixel matrix is computed by

$$F \cdot P = \sum_{i=1}^n \sum_{j=1}^n f_{i,j} p_{i,j}.$$

All those convolutions will be collected and put in a special position to form a feature map. A CNN might have multiple filters and therefore produces multiple feature maps. Each feature map looks for different features of an image. These feature maps will be sent to a pooling layer for reducing the amount of parameters and computation in the network.

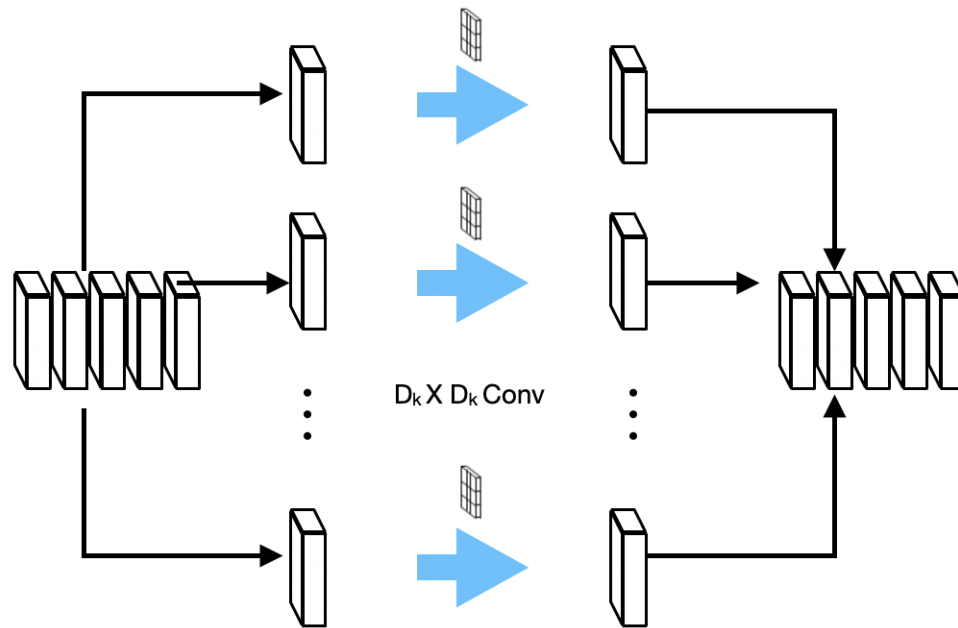
In the rest of this section, we will introduce some widely used transfer learning CNN architectures for image classification tasks, including MobileNet, ResNet and Inception.

### 3.2.1 MobileNet

MobileNets are CNNs for mobile and embedded vision applications based on a streamlined architecture, which use Depth-wise Separable Convolutions to build lightweight deep neural networks. The authors introduce two simple global hyper-

parameters that efficiently trade off between latency and accuracy. Depth-wise Separable Convolution is used to reduce the model size and complexity. These kinds of architectures are very useful for mobile and embedded vision applications[10]. Figure 3.2.1 shows the structure of Depth-wise Separable Convolution.

The first version of MobileNet is MobileNetV1, while MobileNetV2 was created using the concepts of MobileNetV1 by significantly decreasing the number of operations and memory needed while retaining the same accuracy. MobileNetV2 is specifically tailored for mobile and resource constrained environments[21].



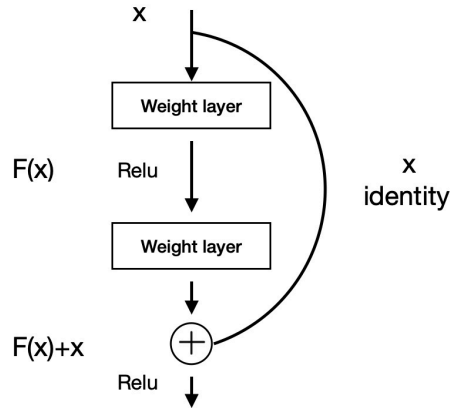
**Figure 3.2.1:** The Structure of Depth-wise Separable Convolution

### 3.2.2 ResNet

ResNets, short for Residual Networks, are CNNs that made it possible to train ultra deep neural networks. Those networks can contain hundreds to thousands of layers and still perform with a high accuracy.[8] ResNets are not only used for computer

vision, they are also being used in non computer vision tasks with better accuracy.

To fix the vanishing gradient problem that exists in deep neural networks, which makes it hard to train, ResNet has introduced a so-called “identity shortcut connection” that skips one or more layers, as shown in Figure 3.2.2.

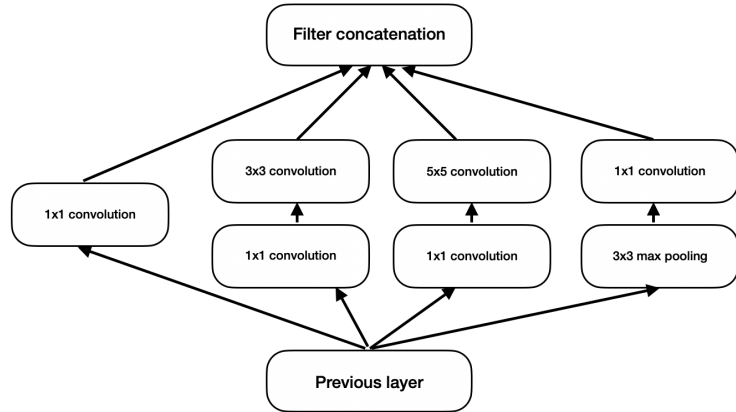


**Figure 3.2.2:** Example of a residual block

### 3.2.3 Inception

Inception has reduced the number of parameters for allowing more efficient computation. The inception layer in the architecture is the core concept of a sparsely connected architecture.

In the inception layer, it works by adding a convolution on an input with not one, but three different sizes of filters ( $1 \times 1$ ,  $3 \times 3$ ,  $5 \times 5$ )[20]. A parallel  $3 \times 3$  max pooling layer has also been added. In addition, an extra  $1 \times 1$  Convolutional layer has been applied before the  $3 \times 3$  and  $5 \times 5$  convolutional layers and after the max pooling layer mainly used for reducing dimensionality. This is shown in Figure 3.2.3.



**Figure 3.2.3:** Inception module with dimension reductions[20]

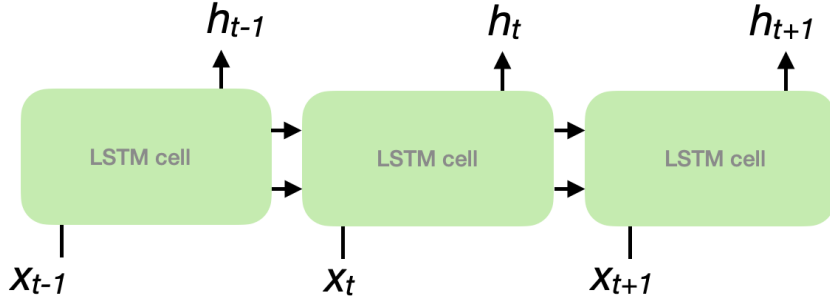
### 3.3 Text-based architectures

#### 3.3.1 Recurrent neural network

In general, the input and output of a neural network are supposed to be independent. Sometimes there are some type of data such as time-series data that each input and output are not independent. For example, each day's stock price is predicted by analysing the previous stock prices. In cases like that, we can use recurrent neural networks to deal with these problems. Recurrent Neural Networks (RNN) remembers all the information in each time step such as previous input and output[16]. In other words, RNN can learn to use the past information.

Long short term memory

Long short-term memory (LSTM) is a widely used neural network architecture of RNN. It was designed to mitigate vanishing and exploding gradient problems which exist in some Deep Neural Networks[9].



**Figure 3.3.1:** The repeating module in an LSTM[24]

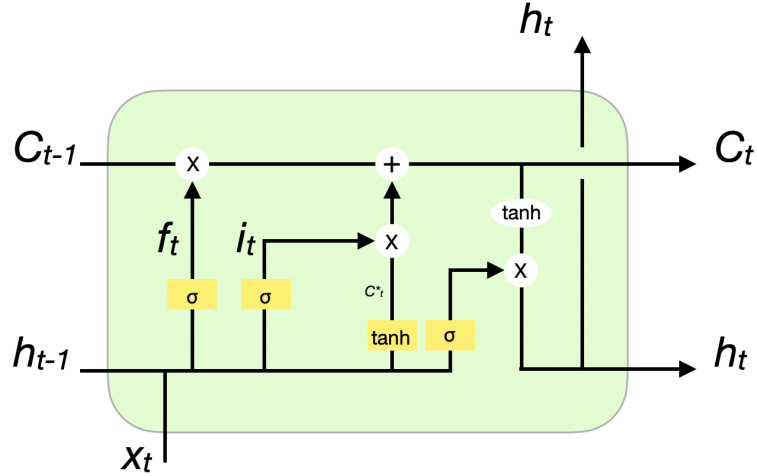
LSTM is an architecture that at each of its time steps, some parameters have been set and act as gates to decide what information to remember and what to forget. In each LSTM cell, there are three binary gates of the same shape, including an input gate, a forget gate and an output gate. The input gate controls whether the memory cell is updated. The forget gate controls if the memory cell is set to 0, while the output gate controls whether the information of the current cell state is visible. That is, the LSTM has the ability to remove or add information to the cell state, carefully regulated by those gates mentioned above[24]. Figure 3.3.1 shows the structure of an LSTM cell.

In Figure 3.3.2,  $x_t$  is the input of this unit and  $h_{t-1}$  is the output from the last unit. These two values will be concatenated for computing the output  $f_t$  from the forget gate layer by

$$f_t = \sigma(W_f \cdot [x_t, h_{t-1}] + b_f),$$

where  $W_f$  is the weight matrix, and  $b_f$  is the bias. Similar to  $f_t$ , the output of the input gate layer  $i_t$  is computed by

$$i_t = \sigma(W_i \cdot [x_t, h_{t-1}] + b_i).$$



**Figure 3.3.2:** Structure of an LSTM cell[24]

Next, a  $\tanh$  layer creates a vector of new candidate values,  $C_t^*$ , which will be added to the old cell state  $C_{t-1}$  from the previous time step to update to a new cell state  $C_t$  by

$$C_t = f_t \cdot C_{t-1} + i_t \cdot C_t^*.$$

The output gate layer determines what information to output for the next unit and it will also be sent to the next unit by

$$o_t = \sigma(W_o \cdot [x_t, h_{t-1}] + b_o);$$

$$h_t = o_t \cdot \tanh(C_t).$$

### 3.3.2 Universal Sentence Encoder

The Universal Sentence Encoder is another module that can be used for our text classification. The Universal Sentence Encoder encodes text into high dimensional vectors that can be used for some natural language tasks[3]. It has two variations;

i.e., one trained with Transformer encoder and the other trained with Deep Averaging Network (DAN)[4]. The one trained with Transformer encoder has higher accuracy, but it is more computationally intensive. The other one has a lower accuracy but it is less computationally intensive.

To get a higher accuracy, we consider using the Universal Sentence Encoder trained with Transformer encoder. The module we use is The Universal Sentence Encoder Lite module, a lightweight version of Universal Sentence Encoder. It is trained on a variety of data sources and a variety of tasks with the aim of dynamically accommodating a wide variety of natural language understanding tasks[4]. Since the module is trained and optimized for greater-than-word length text, it is an appropriate one for our dataset, in which the textual data are short paragraphs.

### 3.4 Multi-modal fusion

To perform the genre classification based on both images and texts, we propose to apply the multi-modal fusion architecture. The multi-modal fusion architecture combines two different types of aforementioned models, one image-based model and one text-based model.

We first analyse the performance of each model. We choose the best performing image-based model when applied to our image classification. Then, we choose the better performing model of either LSTM and Universal Sentence Encoder based on our text classification.

We add an extra dense layer after the two models, then we simply concatenate the image and text features extracted by them. We add another dense layer on the bottom, which outputs the predictions. Overall, we feed the data of video game images and the descriptive texts to the architecture and then the multi-model architecture



can predict the video game by learning both text and image features simultaneously, as shown in Figure 3.4.1.

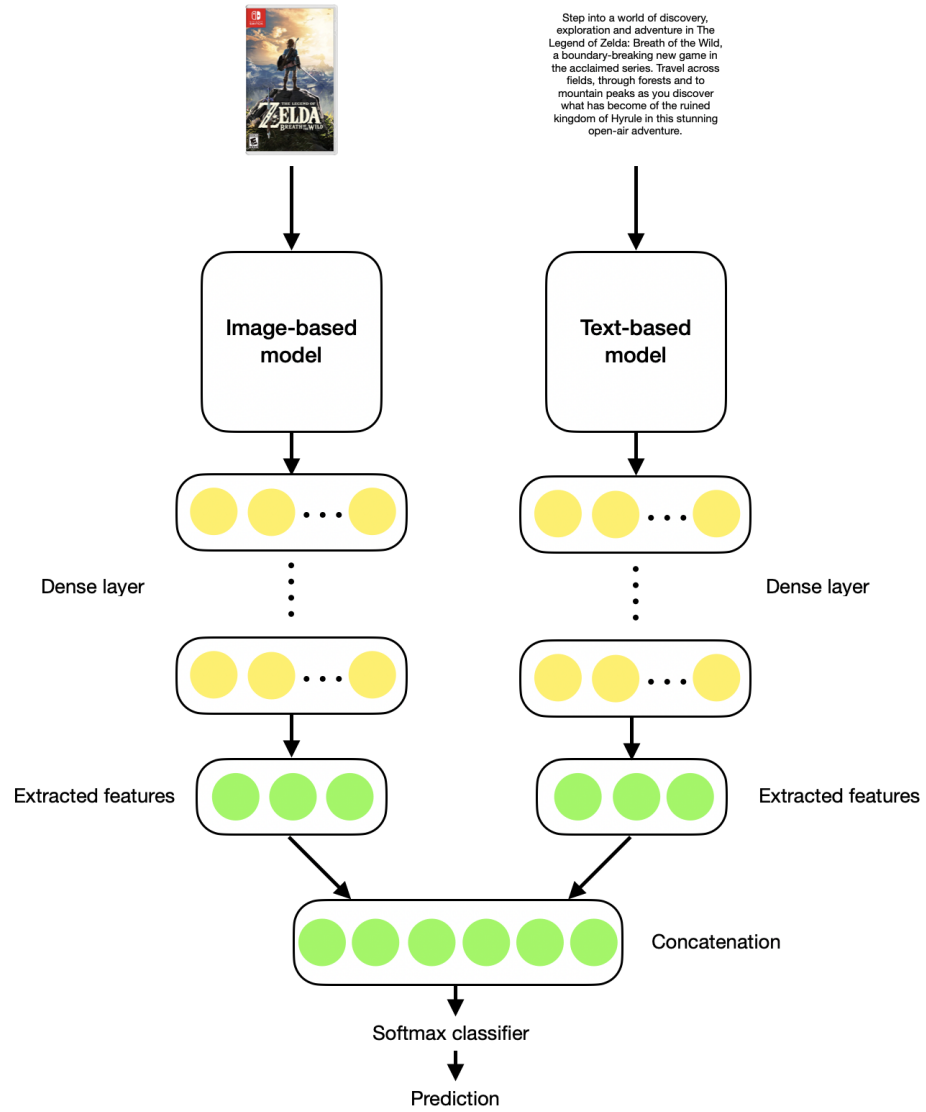


Figure 3.4.1: Multi-modal fusion architecture

## CHAPTER 4

### EXPERIMENTAL STUDY

#### 4.1 Dataset

We compiled a dataset of 50,000 video games in 21 different game genres including information about title text, the game cover image, descriptive text, and the associated genres, collected from IGDB.com<sup>1</sup>, a video game database intended for both game consumers and video game professionals alike. Table 4.1.1 displays the 21 video game genres of our dataset.

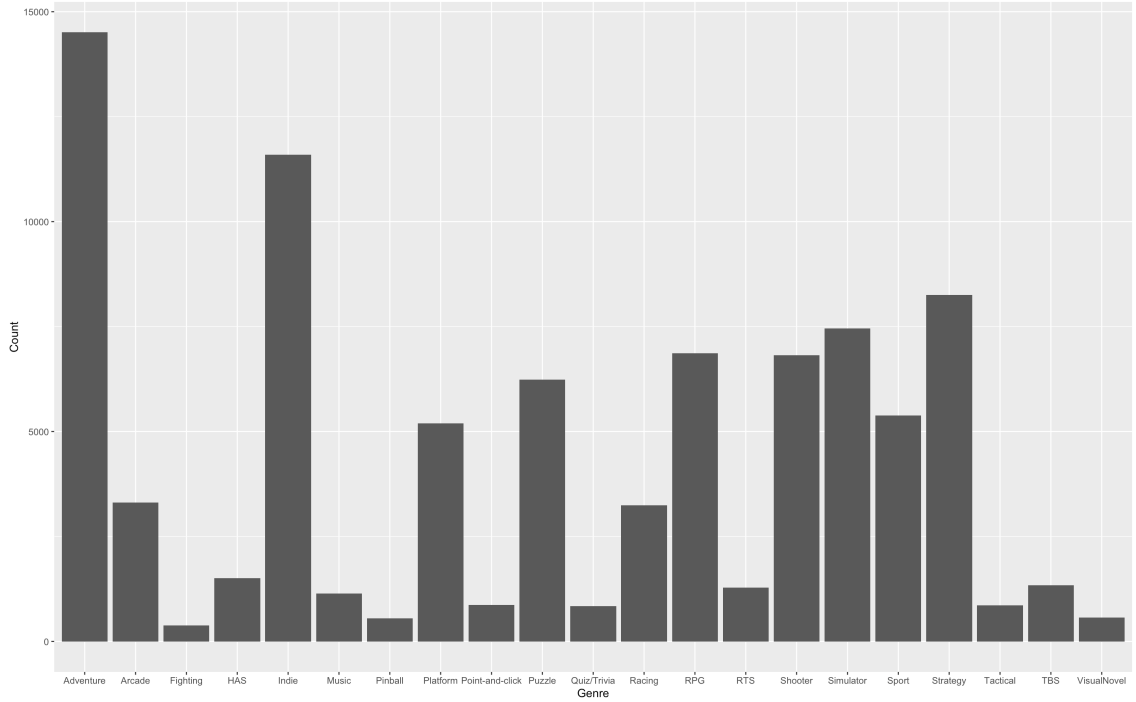
Hack and slash/Beat'em up	Adventure	Arcade
Fighting	Indie	Music
Pinball	Platform	Point and click
Puzzle	Quiz/Trivia	Racing
Real Time Strategy(RTS)	Role-playing(RPG)	Shooter
Simulator	Sport	Strategy
Tactical	Turn-basedstrategy(TBS)	VisualNovel

**Table 4.1.1:** The 21 video game genres

In our original dataset, each video game has three attributes: an image (game cover), a descriptive text, and its genre(s). Some of the video games have multiple genres. The most common genre is *Adventure*, with 14512 video games having this genre, whereas the least common genre is *Fighting*, with only 379 video games. Figure 4.1.1 depicts the original genre distribution.

---

<sup>1</sup><https://www.igdb.com/>



**Figure 4.1.1:** The original genre distribution

Some examples from the original dataset are given in Table 4.1.2.



Title	Cover	Description	Genre(s)
Battlefield: Bad Company 2		Battlefield: Bad Company 2 brings the award-winning Battlefield gameplay to the forefront of PC gaming with best-in-class vehicular combat and unexpected "Battlefield moments". New vehicles like the ATV and a transport helicopter allow for all-new multiplayer tactics on the Battlefield. With the Frostbite-enabled Destruction 2.0 system, you can take down entire buildings and create your own fire points by blasting holes through cover. You can also compete in four-player teams in two squad-only game modes, fighting together to unlock exclusive awards and achievements. Battles are set across expansive maps, each with a different tactical focus. The game also sees the return of the B Company squad in a more mature single-player campaign.	Adventure, Shooter, Simulator, Strategy
Red Dead Redemption		A modern-day Western epic, Red Dead Redemption takes John Marston, a relic from the fast-closing time of the gunslinger, through an open-world filled with wildlife, mini games and shootouts. Marston sets out to hunt down his old gang mates for the government, who have taken away his family, and meets many characters emblematic of the Wild West, heroism and the new civilization along his journey.	Adventure, Role-playing, Shooter
The Legend of Zelda: Breath of the Wild		Step into a world of discovery, exploration and adventure in The Legend of Zelda: Breath of the Wild, a boundary-breaking new game in the acclaimed series. Travel across fields, through forests and to mountain peaks as you discover what has become of the ruined kingdom of Hyrule in this stunning open-air adventure.	Adventure, Role-playing

Table 4.1.2: Samples from the original dataset

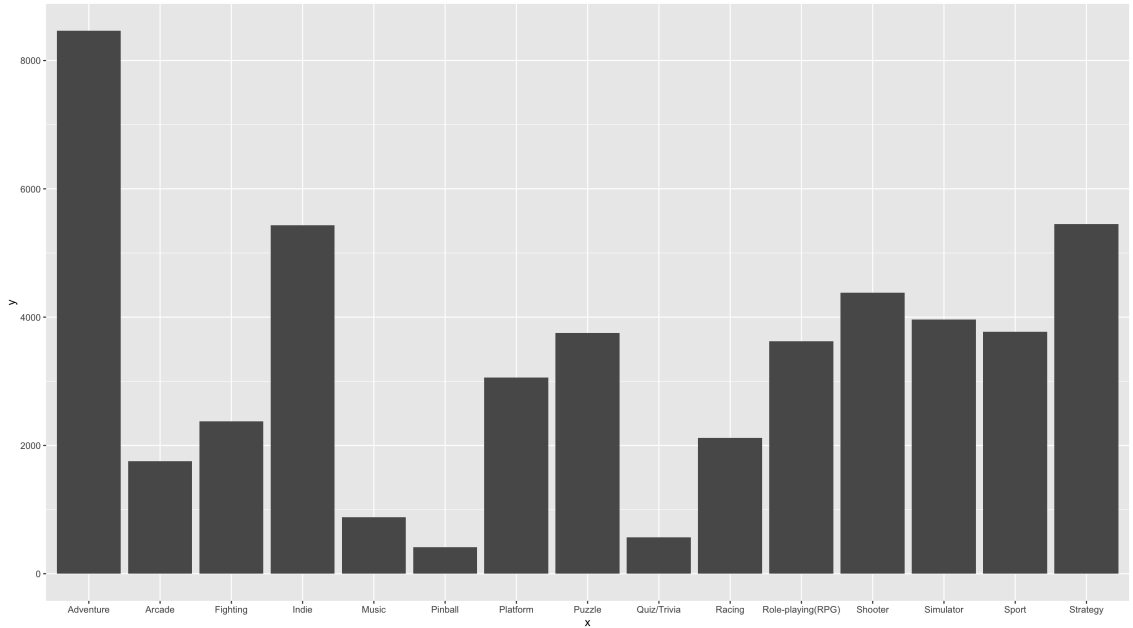
### 4.1.1 Preprocessing

To perform genre classification, we have made some modifications to the original dataset. First, we assume a video game has a single genre, thus, we selected one genre for each video game. There is no information indicating which genre is the “main genre,” therefore we selected it randomly. The genres in the original dataset were labeled without considering the internal relationship among some of the genres. For instance, *Real Time Strategy (RTS)* is a sub-genre of the genre *Strategy*. *Hack and slash/Beat'em up* is a sub-genre of the genre *Fighting*. To avoid the problems caused by these misleading labeling issues, we grouped some genres into one genre, see Table 4.1.3 for details.

Original genres	Grouped as
Strategy, Tactical, Real Time Strategy(RTS), Turn-based strategy(TBS)	Strategy
VisualNovel, Point and click, Adventure	Adventure
Hack and slash/Beat'em up, Fighting	Fighting

**Table 4.1.3:** Grouping of the related genres

In our modified dataset, the number of genres is reduced to 15, where *Adventure* has the highest frequency of 8462, and *Pinball* has the lowest frequency of 416. Figure 4.1.2 depicts the modified genre distribution.






**Figure 4.1.2:** Modified genre distribution

Second, the video game covers are all JPEG files, with RGB color space. The dimensions are different, where they are  $264 \times 352$  in most cases. We resized all the images to  $224 \times 224$ . We picked those dimensions for two reasons: (1) overcoming the memory limitation and (2) there are pre-trained weights for these dimensions.

Third, for text preprocessing, we use the a nltk toolkit for tokenizing the textual data for the input of LSTM. That is, we split the text into lists of words and replace

the words with tokens by assigning a unique integer to all the words. We only tokenize the words with a frequency of at least 10 to eliminate the less important words that might interfere with the accuracy. For the Universal Sentence Encoder, we feed original texts.

Table 4.1.4 displays the modified samples from Table 4.1.2.

Title	Cover	Description	Genre	Genre id
Battlefield: Bad Company 2		1. Original text 2. Encoded text	Adventure	2
Red Dead Redemption		1. Original text 2. Encoded text	Shooter	14
The Legend of Zelda: Breath of the Wild		1. Original text 2. Encoded text	Role-playing	13

**Table 4.1.4:** Modified samples

## 4.2 Experiments

We have three approaches to perform the video game genre classification. First, we use MobileNetV1, MobileNetV2, ResNet50, InceptionV1 and InceptionV2 as image-based models based on video game covers. We have made some modifications to all the image-based models by adding a dense layer of 512 hidden units with ReLU activation and a dense layer of 15 units with softmax activation. Second, we use the LSTM(256) and Universal Sentence Encoder as text-based models based on the descriptive texts. For LSTM, we use the softmax activation for the output layer. Third, we choose the two best models from the image-based and text-based models

for the multi-modal fusion. We collect the features extracted by the these models, then we concatenate these extracted features and feed them into a dense layer with a softmax activation. We use Adam to optimize the sparse-categorical-cross-entropy loss for this model. Then, we analyse the accuracy of each of the aforementioned models. We conduct these experiments on Google Colab<sup>2</sup> using Python and the package Keras with TensorFlow backend.

The output of our network is a 15 dimensional vector  $P = \{p_1, p_2, p_3, \dots, p_{15}\}$ , where  $p_i$  represents the probability of prediction corresponding to genre  $i$  with  $0 \leq p_i \leq 1$  and  $\sum_{i=1}^{15} p_i = 1$ . We evaluate the Top K accuracy, which is the percentage of samples for which the observed genre is one of the Top K genres based on the prediction probabilities from our model. For instance, the the Top 1 accuracy is the percentage of samples for which the observed genre matches the genre with the largest prediction probability. The Top 3 accuracy is the percentage of samples for which the observed genre is among the Top 3 genres based on the prediction probabilities.

In this study, we analyse the Top 1 and Top 3 accuracy. We use 70% of the dataset for training, 10% for validation, and 20% for testing.

### 4.2.1 Results

Table 4.2.1 gives the results of our experiments for the image-based models on the testing set.

---

<sup>2</sup><http://colab.research.google.com>

Models	Top 1 accuracy	Top 3 accuracy
MobileNetV1	29.2%	58.7%
MobileNetV2	28.7%	57.1%
<b>ResNet50</b>	<b>31.4%</b>	<b>61.7%</b>
InceptionV1	27.4%	54.4%
InceptionV2	28.3%	56.5%

**Table 4.2.1:** The Top 1 and Top 3 accuracies for the image-based models on the testing set

ResNet has the best performance based on the Top 1 and Top 3 accuracies, while the MobileNetV2 has the lowest Top 1 and Top 3 accuracies. These results appear to be very close to each other. Table 4.2.2 gives the results of LSTM and Universal Sentence Encoder applied on the texts only.

Models	Top 1 accuracy	Top 3 accuracy
LSTM	44.3%	72.1%
<b>Universal Sentence Encoder</b>	<b>47.7%</b>	<b>76.3%</b>

**Table 4.2.2:** The Top 1 and Top 3 accuracies for the text- based models on the testing set

The Universal Sentence Encoder has the better performance. Thus, we chose ResNet50 and the Universal Sentence Encoder for the multi-modal fusion. Table 4.2.3 shows the Top 1 and Top 3 accuracies for the multi-modal fusion model on the testing set.

Top 1 accuracy	Top 3 accuracy
49.9%	79.9%

**Table 4.2.3:** Multi-modal fusion accuracy with ResNet50 and Universal Sentence Encoder



## CHAPTER 5

### ANALYSIS

In this chapter, we analyse the results obtained by using the three approaches. We also discuss the problems that exist in our dataset or study.

We obtained the best Top 1 accuracy of 31.4% and Top 3 accuracy of 67.7% using ResNet50 based on the images, and we also obtained the best Top 1 accuracy of 47.7% and Top 3 accuracy of 76.3% using Universal Sentence Encoder based on the texts.

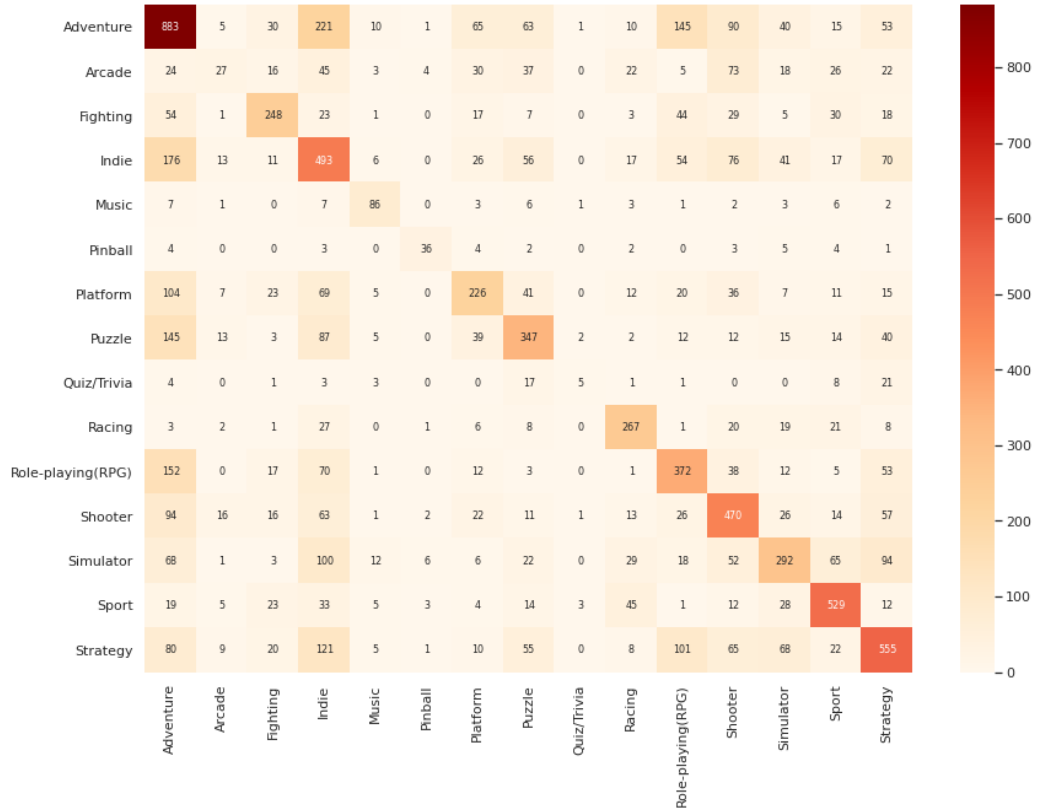
For the multi-modal fusion approach, we obtained Top 1 accuracy of 49.9% and Top 3 accuracy of 79.9%, which is as expected since it incorporates information from both the game cover images and the description text. However, the results do not show a significant greater performance than the text-based approach. It only gains 2% and 3% of Top 1 and Top 3 accuracies, respectively. We present our analysis using this approach.

Table 5.0.1 gives the Top 1 accuracy of multi-modal fusion measured by each genre.

Genre	Top 1 accuracy
Adventure	54.11%
Arcade	7.67%
Fighting	51.67%
Indie	46.69%
Music	67.19%
Pinball	56.25%
Platform	39.24%
Puzzle	47.15%
Quiz/Trivia	7.81%
Racing	69.53%
Role playing (RPG)	50.54%
Shooter	56.50%
Simulator	38.02%
Sport	71.88%
Strategy	49.56%

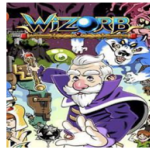
**Table 5.0.1:** Multi-modal fusion accuracy by game genre on the testing set

The best accuracy occurs in *Sport* with 71.88%. The accuracy scores of *Arcade* and *Quiz/Trivia* are all less than 10%; it is especially hard for our models to deal with video games from those genres. Figure 5.0.1 shows the confusion matrix for the multi-modal fusion.



**Figure 5.0.1:** Confusion matrix of multi-modal fusion: vertical axis represents the observed genres; horizontal axis represents the predicted genres.

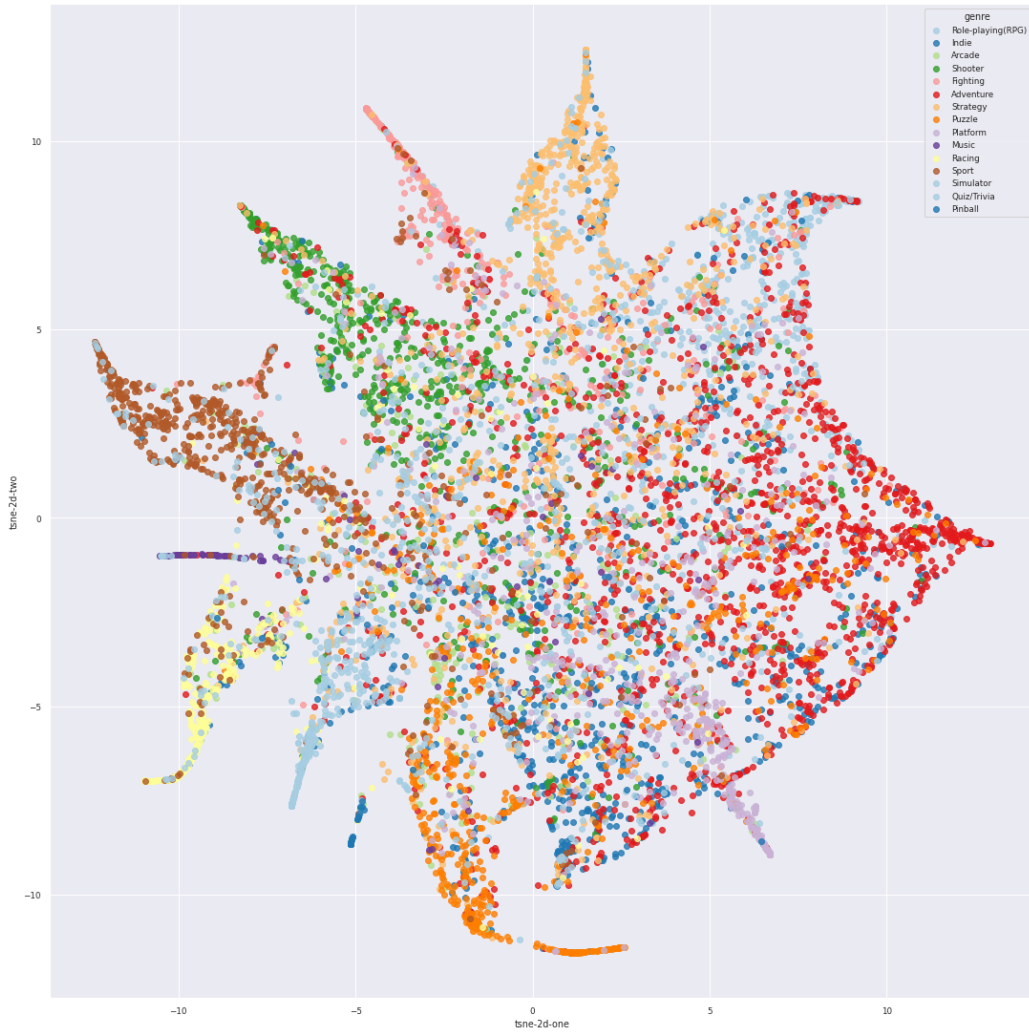
From Figure 5.0.1 we can see that 73 of the *Arcade* games were predicted as *Shooter* and another 45 were predicted as *Indie*. Only 27 *Arcade* games were correctly predicted. Similarly, 104 *Platform* games were predicted as *Adventure* while 226 were correctly predicted. A considerably large proportion of video games with these genres were wrongly predicted as another genre. We analysed the dataset, and one major problem we found that exists in our dataset is that some genres are not well divided. In the original dataset, 31.3% of *Arcade* games are also categorized to *Shooter*, which indicates the two genres are somehow overlapped. Similarly, 34.2% of *Platform* games are also categorized to *Adventure*, which might lead to misclassification. Table 5.0.2 has shown some misclassified examples.

Title	Cover	Original Genre	Predicted genre
SIDEARMS		Arcade	Shooter
Wizorb		Arcade	Indie
Assassin's Creed		Platform	Adventure
Grand Theft Auto		Shooter	Arcade

**Table 5.0.2:** Misclassified examples

The genres *Music*, *Sport* and *Racing* have reached relatively high accuracy at over 70%. These genres are clearly defined and therefore easier for models to deal with. We found that the descriptive texts of these genres have categorical information. For example, *Sport* games often include words like *soccer* and *squad*. *Music* games often include words like *rhythm* and *music*.

The T-SNE plot is given in Figure 5.0.2, from which we can see that *Music*, *Sport* and *Racing* are mapped separately from other genres. Genres like *Adventure*, *Platform*, and *Indie* tend to mix together. The overlap part of these dots reveals the complicated relationships of these genres.



**Figure 5.0.2:** T-SNE plot of video game genre classification

## CHAPTER 6

### CONCLUSION AND FUTURE WORK

In this paper, we developed three deep learning approaches for the task of video game genre classification: one based on the video game cover image, one based on the description text and one based on both the cover image and description text. To the best of our knowledge, this is the first attempt on automatic genre classification for video games using a deep learning approach. We compiled a large dataset of video games consisting of cover images, description text, title text, and the genre information. This dataset will be made available to the public in the future.

In terms of our results, the image-based approach has achieved a Top 1 accuracy of 31.4% and Top 3 accuracy of 67.7%. The text-based approach has achieved a Top 1 accuracy of 47.7% and Top 3 accuracy of 76.3%. The multi-modal fusion approach has achieved the greatest Top 1 and Top 3 accuracies of 49.9% and 79.9%.

There are also some future works that we can do. On one hand, video games usually have multiple genres. Thus, a multi-label genre classification of video games can also be performed based on this dataset. On the other hand, genre classification in other domains like music and movie has been based on the soundtracks and videos, thus, we will conduct a study on the effect of adding these modalities to our study on the task of video game genre classification.

## BIBLIOGRAPHY

- [1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012
- [2] Battu, Varshit, et al. "Predicting the Genre and Rating of a Movie Based on its Synopsis." *PACLIC*. 2018.
- [3] Bengio, Yoshua, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. "A neural probabilistic language model." *Journal of machine learning research* 3, no. Feb (2003): 1137-1155.
- [4] Cer, Daniel, et al. "Universal sentence encoder." *arXiv preprint arXiv:1803.11175* (2018).
- [5] Clarke, Rachel Ivy, Jin Ha Lee, and Neils Clark. "Why video game genres fail: A classificatory analysis." *Games and Culture* 12.5 (2017): 445-465.
- [6] Clearwater, David. "What defines video game genre? Thinking about genre study after the great divide." *Loading...* 5.8 (2011)
- [7] de Eguino, Miguel Flores Ruiz. "Deep Music Genre." (2016).
- [8] Dwivedi, Priya. "Understanding and Coding a ResNet in Keras." (2019).
- [9] Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." *Neural computation* 9, no. 8 (1997): 1735-1780.
- [10] Howard, Andrew G., Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. "Mobilenets: Efficient convolutional neural networks for mobile vision applications." *arXiv preprint arXiv:1704.04861* (2017).

- [11] Huang, Derek A., Arianna A. Serafini, and Eli J. Pugh. "Music Genre Classification."
- [12] Ivasic-Kos, Marina, Miran Pobar, and Ivo Ipsic. "Automatic movie posters classification into genres." International Conference on ICT Innovations. Springer, Cham, 2014.
- [13] Iwana, Brian Kenji, et al. "Judging a book by its cover." arXiv preprint arXiv:1610.09204 (2016).
- [14] Kjartansson, Sigtryggur, and Alexander Ashavsky. "Can you Judge a Book by its Cover?." (2017).
- [15] Ketkar, Nikhil, and Eder Santana. Deep Learning with Python. Vol. 1. Berkeley, CA: Apress, 2017.
- [16] Kundu, Chandra Shakhar. "Book Genre Classification By Its Cover Using A Multi-view Learning Approach." (2020).
- [17] LeCun, Yann, et al. "Gradient-based learning applied to document recognition." Proceedings of the IEEE 86.11(1998): 2278-2324.
- [18] Petrovski, Petar, and Anna Lisa Gentile. "Can you judge a music album by its cover?." (KNOW@ LOD/CoDeS)@ ESWC. 2016.
- [19] Picard, Martin. "The foundation of geemu: A brief history of early Japanese video games." Game Studies 13.2 (2013).
- [20] Raj, Bharath. "A simple Guide to the versions of the Inception network." Retrieved from Towards Data Science website: <https://towardsdatascience.com/a-simpleguide-to-the-versions-of-the-inception-network-7fc52b863202> (2018).



- [21] Sandler, Mark, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. "Mobilenetv2: Inverted residuals and linear bottlenecks." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4510-4520. 2018.
- [22] Simões, Gabriel S., et al. "Movie genre classification with convolutional neural networks." 2016 International Joint Conference on Neural Networks (IJCNN). IEEE, 2016.
- [23] Sung, Samuel, and Rahul Chokshi. "Classification of Movie Posters to Movie Genres."
- [24] Yan, S. "Understanding LSTM networks." Online). Accessed on August 11 (2015).