

Winter 2020

Molecular Social Reality, the Cultural Helix & Sequencing Social DNA

Aidan Christopher Haughey
Portland State University

Follow this and additional works at: <https://pdxscholar.library.pdx.edu/honorsthesis>



Part of the [Theory, Knowledge and Science Commons](#)

Let us know how access to this document benefits you.

Recommended Citation

Haughey, Aidan Christopher, "Molecular Social Reality, the Cultural Helix & Sequencing Social DNA" (2020).
University Honors Theses. Paper 942.
<https://doi.org/10.15760/honors.965>

This Thesis is brought to you for free and open access. It has been accepted for inclusion in University Honors Theses by an authorized administrator of PDXScholar. Please contact us if we can make this document more accessible: pdxscholar@pdx.edu.

Molecular Social Reality, The Cultural Helix & Sequencing Social DNA

- Aidan Christopher Haughey

Social Reality occupies an opaque space in Sociology. Despite being a fundamental aspect of society, its oblique nature has made examination difficult (Searle, 1995). When a subject is so slippery--mired in subjective description, it does beg the question, is it worth analysis? The events of 2020 have provided a glimpse into how conflicting social realities can gravely upset large scale societies when confusion and "misinformation" set the stage for chaotic political theater and rampant social dysfunction in America, "demonstrations have erupted en masse around the country, and they are increasingly met with violence by state actors, non-state actors, and counter-demonstrators alike...Without significant mitigation efforts, these risks will continue to intensify" (Roudabeh Kishi, Sam Jones, 2020). While it is easy to think of these events (Proud Boy rallies that result in horrific violence, national anti mask sentiments that result in political chaos, elected officials endorsing white nationalism, etc.) purely as a problem of misinformation or inadequate education, of individuals not having the academic or technological training to discern truth from disinformation, it is more accurate to describe it as a highly complex set of competing social realities--of fundamentally different conceptions on what specific meaning is attributed to specific symbols (written, visual, or verbal symbols) which has led to this social upheaval. The sheer force of these competing social realities demands measurement of their impact on individuals and societies.

To address this need, my thesis provides new ways of conceptualizing and measuring social reality--encouraging the cataloguing and studying of social realities as a means of understanding their evolution across space and time.

Introduction

Social reality blankets our existence: it is the fabric that ties together symbol and performance into meaning--the substance of symbolic reality. It exists between objective reality (immutable facts of existence like gravity, space, and time) and an individual's ever morphing subjective reality (what an individual personally believes). Defining social reality is a difficulty unto itself, but to begin simply, "There are things that exist because we believe them to exist...money, property, governments, marriage," (Searle, 1995): when something is believed socially (by more than one person), it is real in that society (Thomas Theorem, 2020).

The difficult question is then, how do we study social reality? How can we clearly define social reality when its own definitions are mutable, subjective, and particular to specific social spaces? These are, of course, some of the fundamental questions plaguing social reality research, as Toshio Takeshita writes in *Exploring The Media's Role In Defining Reality*, "The question of how the media mediate between the external objective reality and our social reality (our belief of what the world is like) has been one of the most fundamental themes of mass communications research" (15-27, 1997).

As a means of gaining a clearer understanding of what a social reality is, I've developed a theory for measuring social realities; and a framework and model for measuring social reality presented in media--using a highly trained Machine Learning Algorithm (MLA, a neural network). I also provide a short experiment to demonstrate how this process works on a practical level. This segments my thesis into three main sections: an explanation of my theory of *Molecular Social Reality*, an explanation of my framework and model, *Social DNA* and *The Cultural Helix*, and a MLA experiment which demonstrates how my framework and model operate.

To be clear, I see my theory of *Molecular Social Reality* as separate from my framework and model which are applications of the theory, not part of the theory itself.

Definitions

There are several terms I will be using repeatedly throughout this thesis that deserve a preliminary explanation which will be expanded throughout this thesis.

Social reality is the agreed upon attribution of meaning to a symbolic form only possible when multiple individuals interact and agree on the meaning of a concept and its language (whether that language is verbal, visual, mathematical, etc.). For instance, naming a building that holds money a “bank”. Although a bank does not objectively exist (objectively a bank is just a collection of materials, people, and other things), when we agree on the attribution of “depositing currency” to the word “bank”, that is now socially real for us.

Molecular Social Reality is the main theory my thesis operates on: It is my proposal for how to measure and categorize social reality. The specifics of the theory are in the sections below, but to summarize, social reality can be measured as discrete units of time and space that hold specific organizations of basic social information, and can be categorized accordingly to show all instances of when and where that social reality exists, along with the specific social configurations that social reality propagates.

Social DNA is a framework I have constructed that utilizes my theory of molecular social reality to create a means of “sequencing” Social DNA from media. Social DNA refers to specific social data presented in media as baseline units of social information (in a sense the “genetics” of a social reality), that can be measured and sequenced just like biological DNA is sequenced to understand the underlying molecular components of a biological organism. Social DNA creates a measurable unit of social information based on the double helical model of DNA: it does not extend any further analogy between biology and sociology.

The Cultural Helix is the specific model I have created for measuring and visualizing units of social reality. It is directly based on the double helical structure of DNA, and is used in my *Social DNA* framework as means of visualizing sequenced data. DNA presents an excellent model for understanding discrete units of Social Reality that allows for analysis and measurement of social data presented in media.

Media is consistently analyzed throughout this paper for the social reality presented in a piece of media. By media, this paper refers to any configuration of content made for experiencing by individuals (or a collective of individuals) that uses any form of language as a means of expression (a video, a book, a song, a picture, etc.).

Space and Time are often used in this paper as reference points to locate a social reality. By space and time this paper is referencing space time--the classical mathematical model that the three dimensions of space are inextricably connected to a fourth dimension of time (Carroll, Sean M., 2019.). It is easiest to think of as an x and y graph, with x representing space (for instance, any point on Earth) and y representing time (for instance 5:00pm May 6th, 1996). Although this is an extremely simplistic view of spacetime, it is all that is required for this thesis--understanding that space and time are a way of marking where phenomenon occur. For example, a specific configuration of spacetime might be the center of the Hall street between 5th and 6th street in Portland, Oregon at 6pm Tuesday, March 13th, 2019. That specific moment in space and time is a reference. Any mention of space and time in this paper simply means, at a specific place at a specific time.

Molecular Social Reality

Molecular Social Reality is the theory at the heart of this thesis. It is created on the simple but important notion that social reality exists as discrete instances of space and time, in which a social reality is present and active, and that these instances are measurable based on social factors presented in said social reality.

Molecular social reality is key to understanding my arguments as it is a means of measuring social reality. Critical to it, are the notions of attributing specific social qualities (attributions of age, race, gender, and resource) to instances of space and time. This is vital to objectively understanding social reality as social reality is specific to time and space. For

instance, a social reality in an American midwestern town attributes specific qualities to social phenomenon at a specific time (a negative stance on gay marriage in the 1950's), however, social realities existing in the adjacent town might attribute different qualities to the same social phenomenon (an acceptance of gay marriage). The first town mentioned might even have that social reality change over fifty years (to a positive stance on gay marriage). The point is, social realities evolve and mutate specific to time and location: so considering social reality as discrete moments of space and time allows for when and where a social reality exists to be measured. By tabulating as many instances of the social reality in sequence and considering a social reality as the sum of those instances, a total description of that social reality's existence is possible-- which is the value proposition of *Molecular Social Reality* as a theory.

Social Reality In Media

As this thesis has discussed, social reality is the attribution of a meaning to a symbol by two or more individuals, and although molecular social reality is a theory about social reality existing in real time, an analysis of how a social reality spreads in real time is beyond the scope of this paper; except for in the case of media. A piece of media is experienced in real time by an individual, and contains specific configurations of social reality (Yan F., 2020), which makes it a perfect subject for social reality research--and specifically, molecular social reality, as the experience of the presented social reality (in a piece of media) maps exactly to space and time. For instance, watching a one hour film would map every second of the film experienced to the space and time of the individual watching the film--in this sense, experiencing media becomes indexical to the propagation of social realities.

There can be, in actuality, a complex web of social realities present in a piece of media which complicates analysis. For instance, if a character in a film is presented as Latinx (through their name, dialogue, etc.), that is the social reality presented for that character's race, despite

the fact that the character might be portrayed by someone of a different ethnicity--like Al Pacino, an Italian American, portraying a Columbian drug lord in the film *ScarFace*.

This is why the framework and model I created as an application of *Molecular Social Reality* only focus on the social reality presented in media--the intentional configuration of social data, not social reality incidentally present in media.

Social DNA & The Cultural Helix

To demonstrate an application of molecular social reality, I've developed a baseline unit of social reality modeled after the molecular makeup and physical structure of biological DNA. This framework scopes social reality from a whole into micro units of social reality, *Social DNA*, to analyze frame by frame presentations of social reality in media. It is built on the premise that a piece of media contains the social DNA of a social reality. The data gathered from each frame of a specific piece of media reflects the four key constituents of social reality: gender, race, age, and resource (resource being corporeal or incorporeal, like a physical or intellectual commodity) connected along the values being attributed in the frame (like patriotism, banality, happiness, etc.), and the emotional tone being presented. For instance, a single frame from the film *American Psycho* might show an adult white male using technology to commit violence against women, in a happy mood. Each sequenced frame presents these key elements of social reality (sometimes with neutral elements if race, age, gender, resource, value, or emotion cannot be identified), and by representing the entire sequence of a piece of media through data, a social reality can be seen quantitatively and visually.

The Cultural Helix is a model for understanding social reality presented in media as containing the "DNA" of the social reality it springs from. It is based on the literal structure of biological DNA (OpenStax, DNA Structure and Sequencing, 2014) and represents the most basic unit of social data a social reality can present in a frame of media. It replaces DNA's four

nucleotides with four social nucleotides: race, age, sex, and resource, which form the interconnecting “proteins” of the social DNA. As with DNA, these nucleotides are surrounded by a sugar phosphate ladder (one side sugar and one side phosphate). The “sugar” is the emotion presented in a piece of media, and the “phosphate” is the value being attributed. For instance, a frame of media might present a young white male soldier fighting in anger with military weapons as an act of valor: in this example you have each component of a strand of social DNA presented: race, age, gender, resource, emotion, and value. Figure 0.A below shows my model of the cultural helix, for representing social data visually.

The *Cultural Helix*

*a Two Dimensional Model of Social DNA
at the sociomolecular level*

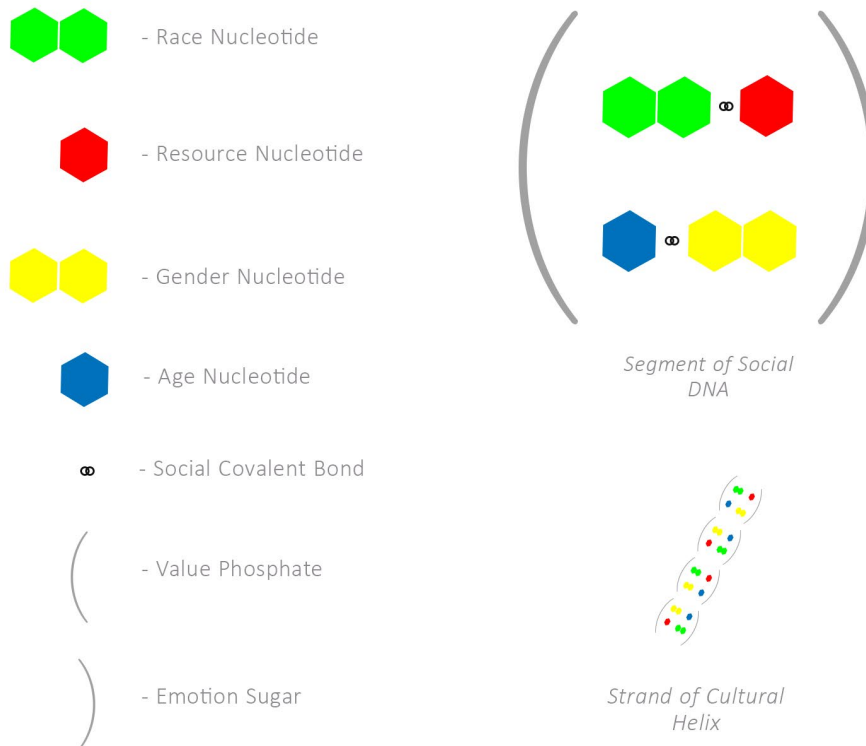


Figure 0.A: The Cultural Helix. A very clear use of the double helical model of DNA with the Guanine nucleotide being replaced with a “race” nucleotide, the Thymine nucleotide being replaced with a “resource” nucleotide, the Adenine nucleotide being replaced by the “gender” nucleotide, and the Cytosine nucleotide being replaced with the “age” nucleotide. Race and age were given the more complex of the four nucleotides as these variables can contain multiple

data points for a single presentation (someone can be multiracial or multi gender), whereas age and resource can only have a single data point.

The race variable is what races are portrayed by the subjects in a frame of a piece of media. The gender variable is what genders are portrayed by the subjects in a single frame. The age variable is the represented ages of the subjects in a frame. The resource variable would be the physical object, intellectual or religious force, or surroundings an individual or individuals are using, affected by, or next to in a single frame. The resource variable breaks down into categories: technological, financial, sexual, domestic, art, education, etc. as do each of the other variables. How to define and code these categories (for example age could be young or old, or a specific age range 12-16, 20 - 25, 45-60) should be debated by a cross disciplinary team of academics and laypeople, as the defining of these variables into an acceptable language for cataloging and studying will be difficult and require input from across the social spectrum to be accurate.

The emotional sugar variable is the emotion being presented in a piece of media. What emotion is being projected through configuration of the media: sadness, anger, confusion, joy? This can be difficult to ascertain as presented emotion is often manifested through a combination of music, dialogue, and visual language (someone frowning, yelling, crying, etc.), but it is a critical part of the social reality being presented. It may be difficult to measure, but when taken on a macro scale, it becomes easier. For instance, music patterns can be learned through repeated input of common music trends into a MLA. Western music, for example, uses major scales and tempos above 125 bpm to express elation and triumph; and use minor scales at or below 125 bpm to express despondency, loss, and introspection. If enough western films were input into a MLA, coded specifically to recognize those music characteristics (measurable as frequencies and bpm), a MLA could easily recognize that a frequency and bpm (and possibly lyrical) arrangement projects a specific emotion. This would be specific to time and space, as

different cultures and times utilize different instrumentation and conceptions of music, but after being fed enough media from a specific era and location, an MLA could recognize the emotion a piece of music represents.

The value phosphate is the most difficult component of Social DNA to categorize, as it represents the value being presented in a frame of Social DNA. While the value of some media is obvious (like fascists' propaganda is obviously nationalistic), training a MLA to recognize the value present in a less obvious piece of media will require analysis and input from media literacy, anthropologists, historians, and laypeople local to the time and space the media come from.

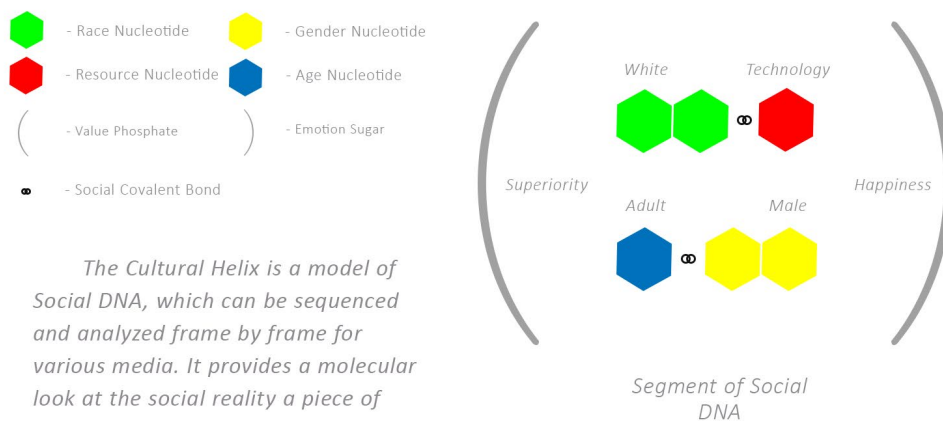
The last component is the covalent bond that allows these nucleotides to come together. In my model this is labeled as the "social covalent bond" which is simply the time and space the media presents in.

The classifications I have provided in my framework are not complete. This is because for the framework to be considered seriously, it invites continual input from qualified sources in classifying social realities in their various configurations. A single person is not capable of creating all the appropriate classification for this framework, it would require thorough interrogation from cross-discipline perspectives to become valid. Which is to say, although I created this framework, I am not qualified to complete the classifications of gender, age, race, resource, emotion, and value--as that requires input from a range of qualified perspectives.

As this model might not seem to present much information in its raw form, below, in figure 0.B, is a single frame of media (from the film *American Psycho*) that has been sequenced as an example to show how this model actually works.

The *Cultural Helix*

a Segment of Social DNA Explored



The Cultural Helix is a model of Social DNA, which can be sequenced and analyzed frame by frame for various media. It provides a molecular look at the social reality a piece of media presents.

Here is a frame from the film American Psycho dissected as a segment of social DNA. The age, race, resource, and gender are identified as: adult, white, technology, and male. They are bonded along the emotional sugar bridge of happiness, and the value phosphate of superiority.



Frame from American Psycho

Figure 0.B: *A single sequenced frame from the film American Psycho, showing how each part of the Cultural Helix corresponds to a specific piece of social data*

Experimental Application of Social DNA Sequencing

To demonstrate how the model and framework of Social DNA and The Cultural Helix work I have devised a short partial sequencing of the film *American Psycho*. This partial

sequencing will only be for about 3 minutes of the beginning of the film to reveal a small percentage of its social DNA. Because creating robust MLAs takes tremendous resources I will only train and code my MLA for the presented gender of each frame sequenced.

Extraction Method - Sequencing Through Machine Learning

In a full application of Social DNA, a frame of media would be analyzed by a highly trained neural network that could then output the data of each frame of a media. However, given my lack of time, money, and personnel, I decided to code an MLA to only recognize presented gender in a narrow scope, by training it on a small sequence of 297 coded frames (about 4 minutes of video) from the film *American Psycho*. Given that I do not have the resources to make a robust sequencing of Social DNA from a piece of media, showing that MLA can be trained to recognize one component of Social DNA demonstrates the value of how the framework of Social DNA and The Cultural Helix could be applied.

The coding system I made for my MLA was purposefully limited (it did not consider presented gender outside of male and female), but given that the only genders presented in the selected sequence were male and female, it was not a serious issue. I created four coding classes and applied them to the 297 sequential frames from *American Psycho*, a portion of which is visible in figure 1.A below.

The screenshot shows an Excel spreadsheet with the following data:

Image_ID	Class
1	Image_ID Class
2	frame0.jpg 2
3	frame1.jpg 2
4	frame2.jpg 3
5	frame3.jpg 3
6	frame4.jpg 3
7	frame5.jpg 3
8	frame6.jpg 3
9	frame7.jpg 3
10	frame8.jpg 3
11	frame9.jpg 3
12	frame10.jp 3
13	frame11.jp 3
14	frame12.jp 3
15	frame13.jp 3
16	frame14.jp 3
17	frame15.jp 3
18	frame16.jp 3
19	frame17.jp 3
20	frame18.jp 3
21	frame19.jp 3
22	frame20.jp 3
23	frame21.jp 3
24	frame22.jp 3
25	frame23.jp 3
26	frame24.jp 0
27	frame25.io 3

Figure 1.A: Section of the 297 frame sequence I coded with each frame name on the first column and corresponding coding on the second column. 0 = presented male, 1 = presented female, 2 = presented male and female, 3 = neither male or female presented

They are hosted on a CSV file and accessed through the programming language Python (a data science programming language I used to write this Machine Learning Algorithm). Each attribution of a code corresponds to 1 of the 297 frames: labeling them for learning by the MLA.

I based my algorithm on a MLA tutorial for recognizing coded images. I first built an algorithm according to the tutorial's specification, and then modified the coding, images, and algorithm to reflect my social DNA coding.

I wrote the script in python, a user-friendly programming language common in data science and image analysis. The script connects to a site called Tensorflow (an organization that provides free machine learning resources) to partition a small layer of a neural network to train and use.

This is all done by executing the script on my computer, which connects to Tensorflow and builds the model in real time, displays the results, then releases the borrowed neural network. Below, in figure 2.A, is the first section of the algorithm, which learns to predict the presented gender in an image based on the coded images I have trained it on.

```

1 from future import absolute_import
2 from keras.utils import np_utils
3 from keras.preprocessing import image # for preprocessing the images
4 from skimage.transform import resize # for resizing images import cv2 # for capturing videos
5 import math # for mathematical operations
6 import matplotlib.pyplot as plt # for plotting the images
7
8 import pandas as pd
9
10 import numpy as np # for mathematical operations
11
12 data = pd.read_csv('mappingAP - mapping.csv') # reading the csv file
13 data.head() # printing first five rows of the file
14
15
16
17
18 X = [] # creating an empty array
19 for img_name in data.Image_ID:
20     img = plt.imread('./frames/' + img_name)
21     X.append(img) # storing each image in array X
22 X = np.array(X) # converting list to array
23
24
25
26
27
28
29 y = data.Class
30 dummy_y = np_utils.to_categorical(y) # one hot encoding classes
31
32
33
34
35
36
37

```

Figure 2.A: Importing the required code libraries and images for learning

This first section imports the required code libraries and resources from various sites that provide open-source resources. My script cannot run without having these libraries first loaded into my code environment. From there it “reads” all the coded images I have provided in a csv file (297 coded frames on presented gender), then it stores them in a collection (an array), and then creates a class out of them for the MLA to understand the coding I have provided. The greyed out text in the script are known as “comments”, or code that will not actually execute because it’s hidden: these are useful for leaving notes for yourself or other developers that might use or modify your script in the future.

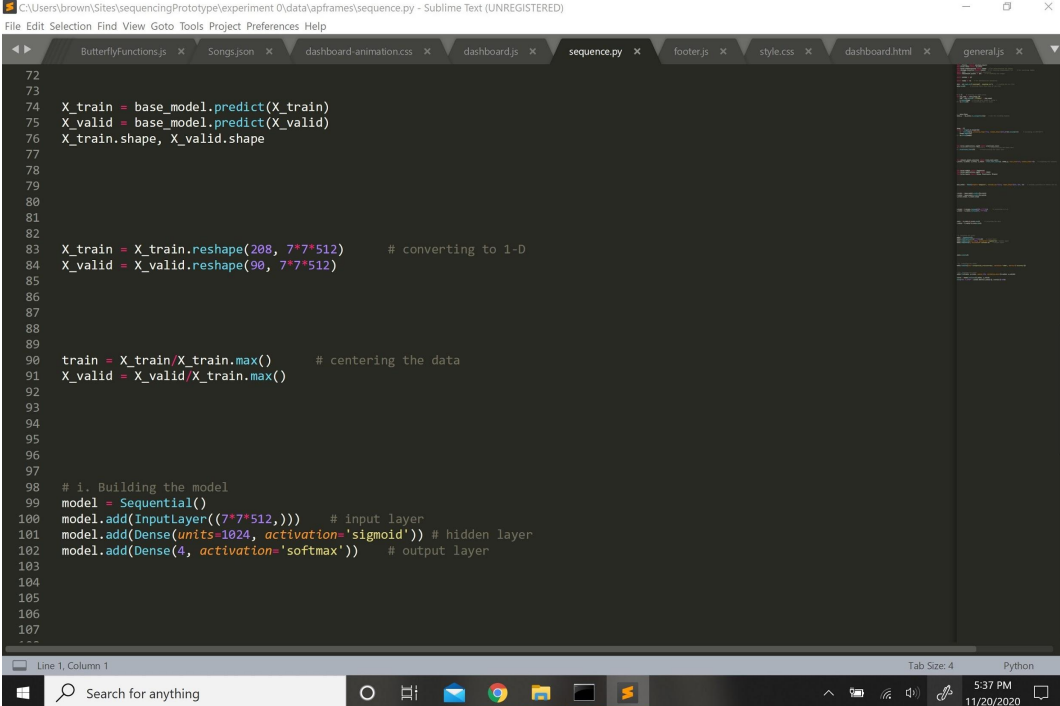

```

36
37 image = []
38 for i in range(0,X.shape[0]):
39     a = resize(X[i], preserve_range=True, output_shape=(224,224)).astype(int) # reshaping to 224*224*3
40     image.append(a)
41 X = np.array(image)
42
43
44
45
46
47 from keras.applications.vgg16 import preprocess_input
48 #X = preprocess_input(X, mode='tf') # preprocessing the input data
49 X = preprocess_input(X) # preprocessing the input data
50
51
52
53
54
55 from sklearn.model_selection import train_test_split
56 X_train, X_valid, y_train, y_valid = train_test_split(X, dummy_y, test_size=0.3, random_state=42) # preparing the validation set
57
58
59
60
61 from keras.models import Sequential
62 from keras.applications.vgg16 import VGG16
63 from keras.layers import Dense, InputLayer, Dropout
64
65
66
67
68
69 base_model = VGG16(weights='imagenet', include_top=False, input_shape=(224, 224, 3)) # include_top=False to remove the top layer
70
71
72

```

Figure 2.B: Reshaping image data and importing a pretrained model as a base for my model

This next section, seen above in figure 2.B, reshapes the stored images into dimensions the MLA can better understand, instead of the photos original dimensions. That data is then preprocessed using resources from tensorflow, that helps our model better understand them by adding information helpful for image analysis by a neural network. The pretrained model (VGG16) is then imported and used as a base for our model to better help it analyze aspects like shading and perspective (which can introduce unintended biases in the MLA). The images are then split into training and validation sets. The training set is the images the MLA will learn from, the validation set is the coded images the MLA will be tested on for accuracy in predicting presented gender based on my image coding.



```
72
73
74 X_train = base_model.predict(X_train)
75 X_valid = base_model.predict(X_valid)
76 X_train.shape, X_valid.shape
77
78
79
80
81
82
83 X_train = X_train.reshape(208, 7*7*512) # converting to 1-D
84 X_valid = X_valid.reshape(98, 7*7*512)
85
86
87
88
89
90 train = X_train/X_train.max() # centering the data
91 X_valid = X_valid/X_train.max()
92
93
94
95
96
97
98 # i. Building the model
99 model = Sequential()
100 model.add(InputLayer((7*7*512,))) # input layer
101 model.add(Dense(units=1024, activation='sigmoid')) # hidden layer
102 model.add(Dense(4, activation='softmax')) # output layer
103
104
105
106
107
--
```

Figure 2.C

The model is then configured for image analysis and built using one input layer (a layer that takes data), on hidden layer (the internal MLA logic that builds itself based on training), and one output layer (the answer given after the data from the input layer is analyzed by the logic of the hidden layer). This can all be seen in figure 2.C above, with corresponding comments in the script denoting what is happening in each line of code.

```

106
107
108
109 model.summary()
110
111
112
113
114 # ii. Compiling the model
115 model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
116
117
118
119 # iii. Training the model
120 model.fit(train, y_train, epochs=100, validation_data=(X_valid, y_valid))
121
122 scores = model.evaluate(X_valid, y_valid)
123 print("%s: %.2f%%" % (model.metrics_names[1], scores[1]*100))

```

Figure 2.D: The model summarizes itself, trains, and is then tested for accuracy

This final section of the image-prediction-building-and-training outputs aspects of the model, compiles and trains the models on the collected reshaped data, and then tests the accuracy of the model on 100 validation images (the images set aside from the training set). Each validation image is tested for accuracy by the model, and based on the scores out of 100 validation images, an accuracy of x percent is attained. Figure 2.D above shows this carried out in code.

It is a very basic script that creates a model based on the input of the 297 coded frames I provided. The model imports my coded images, uses a VGG16 pretrained model to help improve accuracy, reshapes the images (so the model can better understand them) and sorts them into two categories (testing and validation), partitions a layer of 1024 neurons from tensorflow to create a neural network, builds the active model, trains the model on the training set of images, and finally tests for accuracy against 100 validation images generated by tensorflow. The process is shown in the following figures as it would happen in real time--it is conducted through a command prompt.

```

Select Nodejs command prompt
2020-11-20 08:41:27.408694: W tensorflow/core/framework/cpu_allocator_impl.cc:81] Allocation of 411041792 exceeds 10% of
free system memory.
2020-11-20 08:41:33.164409: W tensorflow/core/framework/cpu_allocator_impl.cc:81] Allocation of 411041792 exceeds 10% of
free system memory.
2020-11-20 08:41:33.309666: W tensorflow/core/framework/cpu_allocator_impl.cc:81] Allocation of 411041792 exceeds 10% of
free system memory.
2020-11-20 08:41:39.154813: W tensorflow/core/framework/cpu_allocator_impl.cc:81] Allocation of 411041792 exceeds 10% of
free system memory.
Model: "sequential"

Layer (type)                 Output Shape                 Param #
-----
dense (Dense)                (None, 1024)                25691136
-----
dense_1 (Dense)              (None, 4)                   4100
-----
Total params: 25,695,236
Trainable params: 25,695,236
Non-trainable params: 0

Epoch 1/100
7/7 [=====] - 1s 141ms/step - loss: 0.9087 - accuracy: 0.6298 - val_loss: 0.4131 - val_accuracy
: 0.9222
Epoch 2/100
7/7 [=====] - 1s 101ms/step - loss: 0.2701 - accuracy: 0.9231 - val_loss: 0.1711 - val_accuracy
: 0.9667
Epoch 3/100
7/7 [=====] - 1s 102ms/step - loss: 0.1197 - accuracy: 0.9567 - val_loss: 0.2214 - val_accuracy
: 0.9333
Epoch 4/100

```

Figure 3.A: The above frame shows the neural network with 1024 active neurons

Figure 3.A shows an output of a “summary” of the model before it begins to test itself against the 100 validation images set aside from the 297 coded images. In the first section of lines you can see tensorflow using more than 10% of my laptops computing power to train and build my model (given that my model is very basic, this demonstrates the sheer computational power it takes to create and run even basic 4 parameter MLAs). The second section is a summary of my model, displaying how many active neurons it is currently using under the “Output Shape” column, along with how many parameters it is trained on (4), and other aspects of the model. The final section of figure 3.A shows the actual test of the model against each validation image.

```

Node.js command prompt - python sequence.py
C:\Users\brown\Sites\sequencingPrototype\experiment 0\data\apframes>python sequence.py
2020-11-20 08:41:15.497752: I tensorflow/stream_executor/platform/default/dso_loader.cc:48] Successfully opened dynamic
library cudart64_101.dll
2020-11-20 08:41:26.626454: W tensorflow/stream_executor/platform/default/dso_loader.cc:59] Could not load dynamic libra
ry 'nvcuda.dll'; dLError: nvcuda.dll not found
2020-11-20 08:41:26.626689: W tensorflow/stream_executor/cuda/cuda_driver.cc:312] failed call to cuInit: UNKNOWN ERROR (
303)
2020-11-20 08:41:26.634078: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:169] retrieving CUDA diagnostic inform
ation for host: DESKTOP-N7BFDUT
2020-11-20 08:41:26.636834: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:176] hostname: DESKTOP-N7BFDUT
2020-11-20 08:41:26.637879: I tensorflow/core/platform/cpu_feature_guard.cc:142] This TensorFlow binary is optimized wit
h oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations:
AVX2
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2020-11-20 08:41:26.645375: I tensorflow/compiler/xla/service/service.cc:168] XLA service 0x1f9f29a4d20 initialized for
platform Host (this does not guarantee that XLA will be used). Devices:
2020-11-20 08:41:26.645526: I tensorflow/compiler/xla/service/service.cc:176] StreamExecutor device (0): Host, Default
Version
2020-11-20 08:41:27.239135: W tensorflow/core/framework/cpu_allocator_impl.cc:81] Allocation of 411041792 exceeds 10% of
free system memory.
2020-11-20 08:41:27.408694: W tensorflow/core/framework/cpu_allocator_impl.cc:81] Allocation of 411041792 exceeds 10% of
free system memory.
2020-11-20 08:41:33.164409: W tensorflow/core/framework/cpu_allocator_impl.cc:81] Allocation of 411041792 exceeds 10% of
free system memory.
2020-11-20 08:41:33.309666: W tensorflow/core/framework/cpu_allocator_impl.cc:81] Allocation of 411041792 exceeds 10% of
free system memory.
2020-11-20 08:41:39.154813: W tensorflow/core/framework/cpu_allocator_impl.cc:81] Allocation of 411041792 exceeds 10% of
free system memory.

```

Figure 3.B: shows the neural network being partitioned from tensorflow and training.

```

Node.js command prompt
7/7 [=====] - 1s 96ms/step - loss: 4.3175e-04 - accuracy: 1.0000 - val_loss: 0.1018 - val_accu
acy: 0.9778
Epoch 93/100
7/7 [=====] - 1s 96ms/step - loss: 4.2431e-04 - accuracy: 1.0000 - val_loss: 0.1019 - val_accu
acy: 0.9778
Epoch 94/100
7/7 [=====] - 1s 96ms/step - loss: 4.1709e-04 - accuracy: 1.0000 - val_loss: 0.1020 - val_accu
acy: 0.9778
Epoch 95/100
7/7 [=====] - 1s 95ms/step - loss: 4.0988e-04 - accuracy: 1.0000 - val_loss: 0.1022 - val_accu
acy: 0.9778
Epoch 96/100
7/7 [=====] - 1s 95ms/step - loss: 4.0321e-04 - accuracy: 1.0000 - val_loss: 0.1022 - val_accu
acy: 0.9778
Epoch 97/100
7/7 [=====] - 1s 95ms/step - loss: 3.9654e-04 - accuracy: 1.0000 - val_loss: 0.1023 - val_accu
acy: 0.9778
Epoch 98/100
7/7 [=====] - 1s 96ms/step - loss: 3.9022e-04 - accuracy: 1.0000 - val_loss: 0.1024 - val_accu
acy: 0.9778
Epoch 99/100
7/7 [=====] - 1s 95ms/step - loss: 3.8367e-04 - accuracy: 1.0000 - val_loss: 0.1024 - val_accu
acy: 0.9778
Epoch 100/100
7/7 [=====] - 1s 96ms/step - loss: 3.7748e-04 - accuracy: 1.0000 - val_loss: 0.1023 - val_accu
acy: 0.9778
3/3 [=====] - 0s 17ms/step - loss: 0.1023 - accuracy: 0.9778
accuracy: 97.78%
C:\Users\brown\Sites\sequencingPrototype\experiment 0\data\apframes>python sequence.py

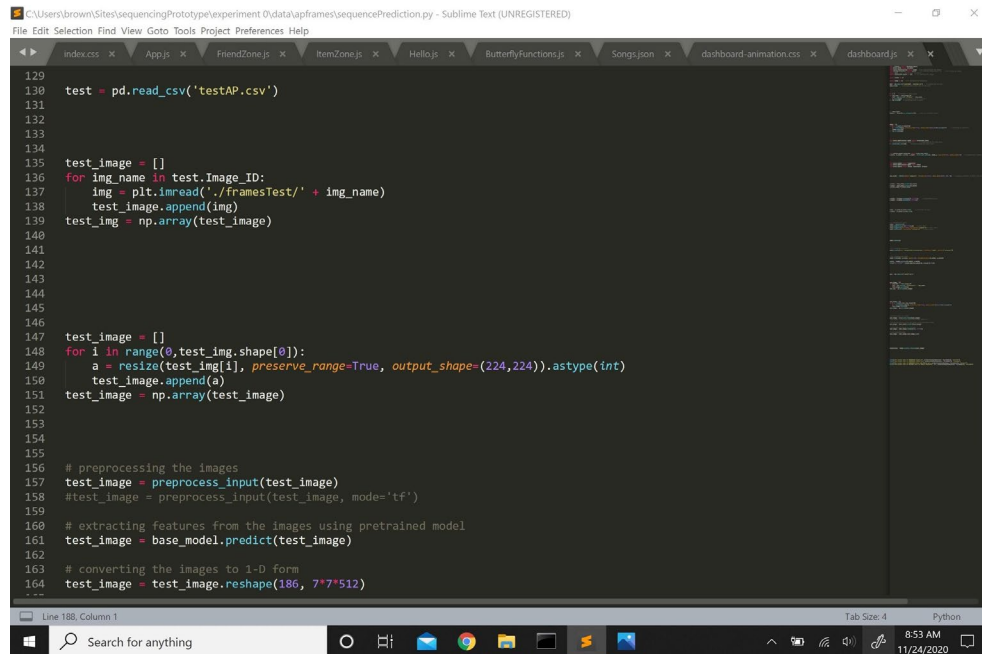
```

Figure 3.C: shows the accuracy of the model on each provided validation image (an epoch).

The final result of the analysis reveals a 97.78% accuracy for the model

After ascertaining the accuracy of my model, I tested it on a sequence of 185 frames from the beginning of the film, up to before the coded sequence begins. This was done by adding a second section to the MLA script that utilizes the model created, that predicts

presented gender, on a selection of “test” images I provided. The script is shown in the figures below.

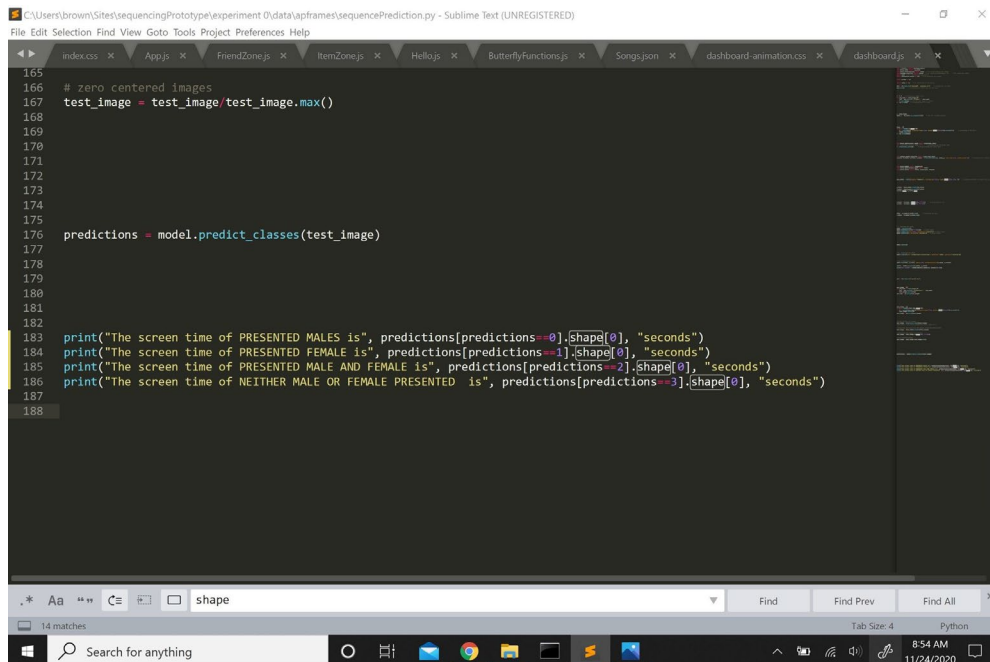


```

129
130 test = pd.read_csv('testAP.csv')
131
132
133
134
135 test_image = []
136 for img_name in test.Image_ID:
137     img = plt.imread('./framesTest/' + img_name)
138     test_image.append(img)
139 test_img = np.array(test_image)
140
141
142
143
144
145
146
147 test_image = []
148 for i in range(0, test_img.shape[0]):
149     a = resize(test_img[i], preserve_range=True, output_shape=(224,224)).astype(int)
150     test_image.append(a)
151 test_image = np.array(test_image)
152
153
154
155
156 # preprocessing the images
157 test_image = preprocess_input(test_image)
158 #test_image = preprocess_input(test_image, mode='tf')
159
160 # extracting features from the images using pretrained model
161 test_image = base_model.predict(test_image)
162
163 # converting the images to 1-D form
164 test_image = test_image.reshape(186, 7*7*512)
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188

```

Figure 4.A: The MLA gathers my test images into a collection and processes them for prediction--very similarly to the first section of the script



```

165
166 # zero centered images
167 test_image = test_image/test_image.max()
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188

```

```

165
166 # zero centered images
167 test_image = test_image/test_image.max()
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188

```

```

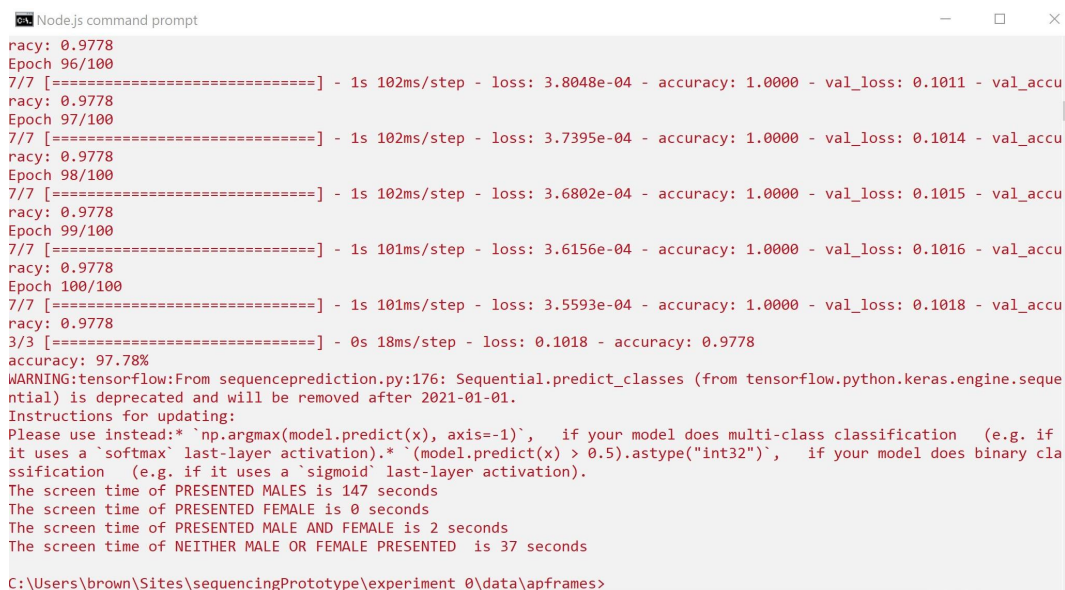
183 print("The screen time of PRESENTED MALES is", predictions[predictions==0].shape[0], "seconds")
184 print("The screen time of PRESENTED FEMALE is", predictions[predictions==1].shape[0], "seconds")
185 print("The screen time of PRESENTED MALE AND FEMALE is", predictions[predictions==2].shape[0], "seconds")
186 print("The screen time of NEITHER MALE OR FEMALE PRESENTED is", predictions[predictions==3].shape[0], "seconds")
187
188

```

Figure 4.B: The MLA then attempts to predict the gender presented in each test frame I provided, based on the learning it gained from my coded images

This portion of the script (shown in figures 4.A and 4.B) is much simpler as it does not need to train or create a model (since the model is already created in the first section of the script), and only has to reshape/preprocess the provided images and run an analysis on them using the already created and trained model.

After running analysis, the model was able to predict a surprisingly accurate percentage of presented gender on the test clips. The output is shown in figure 5.A below



```

Node.js command prompt
racy: 0.9778
Epoch 96/100
7/7 [=====] - 1s 102ms/step - loss: 3.8048e-04 - accuracy: 1.0000 - val_loss: 0.1011 - val_accu
racy: 0.9778
Epoch 97/100
7/7 [=====] - 1s 102ms/step - loss: 3.7395e-04 - accuracy: 1.0000 - val_loss: 0.1014 - val_accu
racy: 0.9778
Epoch 98/100
7/7 [=====] - 1s 102ms/step - loss: 3.6802e-04 - accuracy: 1.0000 - val_loss: 0.1015 - val_accu
racy: 0.9778
Epoch 99/100
7/7 [=====] - 1s 101ms/step - loss: 3.6156e-04 - accuracy: 1.0000 - val_loss: 0.1016 - val_accu
racy: 0.9778
Epoch 100/100
7/7 [=====] - 1s 101ms/step - loss: 3.5593e-04 - accuracy: 1.0000 - val_loss: 0.1018 - val_accu
racy: 0.9778
3/3 [=====] - 0s 18ms/step - loss: 0.1018 - accuracy: 0.9778
accuracy: 97.78%
WARNING:tensorflow:From sequenceprediction.py:176: Sequential.predict_classes (from tensorflow.python.keras.engine.seque
ntial) is deprecated and will be removed after 2021-01-01.
Instructions for updating:
Please use instead: * `np.argmax(model.predict(x), axis=-1)`, if your model does multi-class classification (e.g. if
it uses a `softmax` last-layer activation). * `(model.predict(x) > 0.5).astype("int32")`, if your model does binary cla
ssification (e.g. if it uses a `sigmoid` last-layer activation).
The screen time of PRESENTED MALES is 147 seconds
The screen time of PRESENTED FEMALE is 0 seconds
The screen time of PRESENTED MALE AND FEMALE is 2 seconds
The screen time of NEITHER MALE OR FEMALE PRESENTED is 37 seconds

C:\Users\brown\Sites\sequencingPrototype\experiment 0\data\apframes>

```

Figure 5.A: This output shows the predicted gender of each unknown frame in the test sequence I provided. The MLA was able to predict with above 90% accuracy the gender of each test frame, shown in the outputs “The screen time of X is Y seconds”

Results

My MLA revealed an accuracy of 97.78% in being able to recognize gender based on the Social DNA coding I provided. Such a high accuracy was honestly surprising and could be due to the simplicity of my coding scheme and the VGG16 pretrained model that my MLA model uses as a base. These pre-trained models are likely already biased to recognize presented white male or female genders as pretrained models tend to be biased in recognition towards white mainstream genders (S. Cardenas and S. F. Vallejo-Cardenas, 2019).

Although an accuracy this high is a little suspicious (since it was only trained on 297 coded images), it does clearly show that coding an MLA to recognize social data is entirely possible. If an MLA can be trained to recognize gender, it can likely be trained to recognize any visual pattern (which is not something new in the world of data analysis). With time, money, and a diverse staff a comprehensive social coding system could be devised and taught to a complex neural network that could then analyze large amounts of data from media, revealing quantifiable social data.

The model's prediction of presented gender on the test images is startlingly accurate, which does correspond to the 97.78% accuracy rating the model generates. For instance, there are no single frames of PRESENTED FEMALE gender in the test images I provided and that was predicted correctly by the MLA. There was also about 37 seconds of NEITHER MALE NOR FEMALE PRESENTED which the MLA correctly predicted. The only serious discrepancy was between PRESENTED MALES and PRESENTED MALES AND FEMALES, with the prediction biased towards recognizing PRESENTED MALE more often, instead of PRESENTED MALE AND FEMALE. This is likely because my coding did not have a robust set of PRESENTED MALE AND FEMALE frames for the MLA to learn from. I am confident that with better training a more accurate prediction could have been ascertained.

The data garnered from this experiment acts as a partial sequencing of a 3 minute sequence of *American Psycho*, and while it does not reveal race, age, or resource, it does show a small percentage of gender representation presented in the social reality of *American Psycho*.

The fact that I lacked the resources to code for the other variables of Social DNA in this sequence is an expected part of Social DNA sequencing; and the model of The Cultural Helix was actually built to show full or partial sequencing, as sequencing new or unknown media will likely reveal only partial sequences until an MLA can be coded to understand new configurations of social reality. Figure 6.A below shows what this partial sequencing looks like, utilizing the data from my experiment.

Partial Sequence

Frames



Sequence

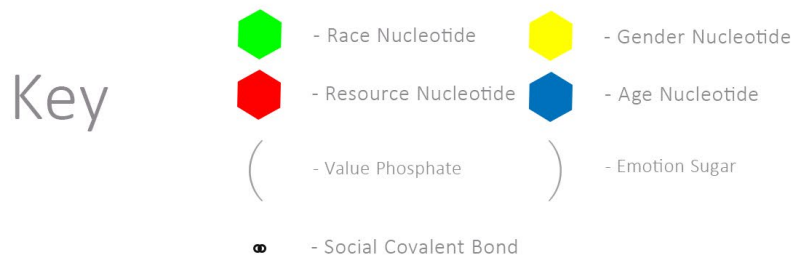
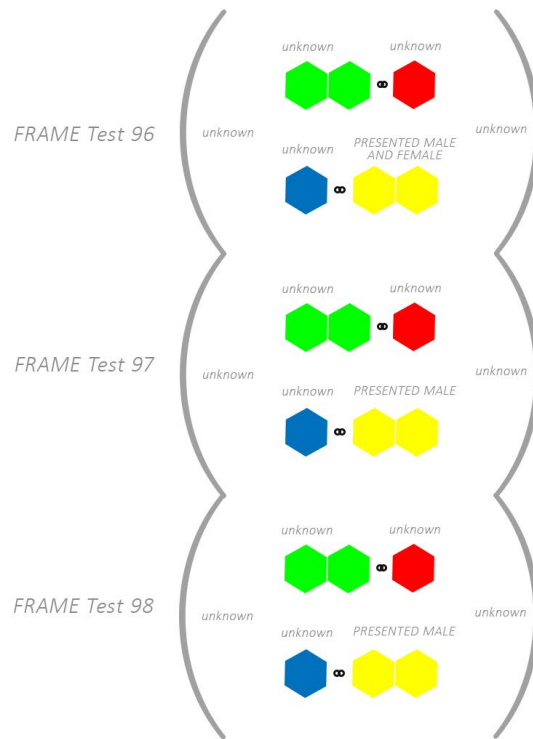


Figure 6.A: This partial sequencing carried out by my MLA reveals the gender presented in the social DNA of the sequenced frames. This figure only shows three of the 185 sequenced test frames, with unidentifiable social data being labeled as unknown

Limitations, Considerations, & Future Applications

Coding Social Reality in video is very complex and poses a number of coding considerations. Although pretrained models (like the VGG16 I used in my MLA) do help alleviate some of these problems I argue for a completely new model to be trained and built to remove institutional biases likely present in pretrained models. Putting aside considerations already raised in this paper on the difficulty of coding presented gender, age, race, resource, there are practical considerations like perspective, which pose difficulties in themselves. This was the case in coding certain frames like the one featured in figure 7.A below, which posed a difficulty in ascertaining gender or genders presented.



Figure 7.A: Frame 182 the Model was trained on

I coded this as PRESENTED MALE, but there are no distinct male characteristics in the frame, and furthermore, there is technically a PRESENTED FEMALE in the background of the shot. Should this be coded as PRESENTED MALE or PRESENTED MALE AND FEMALE? For sake of ease in this experiment I coded it as PRESENTED MALE, but it warrants serious debate, and brings to mind the incredible complexity in forming agreeable coding schemes for presented social realities on a practical level. When we know characters are presenting with a specific gender in a piece of media (because of plot or character details revealed) it's simple

enough to code for the presented gender, but what about background and extra characters whose presented gender is not explicit?

Another serious consideration is reflection. Will an MLA be confused by coding the frame pictured in figure 7.B as PRESENTED MALE when the subject's reflection is morphed with the art piece (picture of Les Miserable) presented?



Figure 7.B: This coded frame demonstrates the difficulty of coding for gender in reflections

I labeled this as PRESENTED MALE for sake of ease, but, it begs further investigation as to how reflections should be coded.

My last serious consideration is language within visual media. What a character says can have a direct impact on Value and Emotion being presented in a piece of media. As I only coded for presented gender in my experiment, I did not have to consider the difficulty of coding for language. However, given that captions can be provided to corresponding frames and neural networks can be trained to recognize and respond to speech (like GPT-3, a new neural network software that can recognize and react to real human speech with cogent and believable responses), measurement of language presented in a frame of Social Reality should be undertaken as well.

Putting these considerations aside for the moment and looking to the future, the whole point of this framework is to build a neural network that can recognize social data from specific

eras and locations. The MLA can then be fed media, sequence it, and record it. By feeding the MLA media from a specific era, you could then build a large database of social reality that maps how social realities evolve in specific regions of time and space.

For instance, if an MLA was trained appropriately and then fed enough media from fascist regimes in WW2 it could recognize fascism as unique configurations of Social DNA (consistently portraying specific gender and race in disparaging forms, nationalism as a resource, etc.), you could then feed it current forms of media from contemporary authoritarian regimes and see how the fascist strand of Social DNA from the 1930s compares to fascist media from today. An appropriately trained MLA could tell you how the sequences of social DNA matched or changed on specific configurations, how the social reality of fascism has evolved over time--and it would be specific to location and era. With a comprehensive enough MLA you could trace the evolution of authoritarian social realities over the world and compare them to emerging authoritarian regimes.

However, this application of Social DNA again brings us to the problem of coding. How is a fascist social reality coded and recognized? In the case of fascism in WW2, so much incredible analysis has already undergone on the subject (for generations) it would be much easier to recognize and code media from that era, however, emerging media often springs from the font of modernity, and the same analysis that worked on previous social reality will likely only sequence portions of emerging social realities--which often apply new meanings to old terms, making analysis difficult. This means MLAs that track social reality would have to be constantly overseen and updated by a dedicated team of cross disciplinary professionals, to sequence emerging social reality accurately.

Conclusion

There are undoubtedly problems within my theory, framework, and experiment. I do not see this work as perfect, or even DNA as necessarily the best model for measuring infinitesimal units of Social Reality. However, I am in full confidence of the efficacy of this line of thought. Social reality does literally spread through instances of time and space, whether that's verbal instances between individuals, interaction with media, or any other form of interaction, social realities exist through propagation. To not seek a clear understanding of what each social reality is, and how interaction between it and individuals occurs would be to ignore the power of Social Reality, which the year of 2020 has shown us to be folly.

It is easy to characterize the mass confusion, chaos, and civil upset of 2020 as misinformation. That the large swaths of Qanon, Trump, and militia supporters are simply unintelligent, misinformed, and unable to decipher arguments academically; however, the problem is more fundamental. The complex demographics that form 21st century nations (demographics feeding off the unending buffet of internet social realities) have incredibly complex and contentious social realities that make civil engagement disturbingly difficult, as demographics have literal differing conceptions of what something means and is.

This complexity will multiply as technology and media continue to center human life. Careful observation of this is warranted as global societies become further segmented and confused by the clash of social realities. A map of how these social realities evolve over time and space will be critical for understanding why demographics do or don't work well together.

Bibliography

Carroll, Sean M. *Spacetime and geometry*. Cambridge University Press, 2019.

Roudabeh Kishi, Sam Jones. *Demonstrations & Political Violence In America: New Data For Summer 2020* . ACLED, 3rd Sept. 2020, <https://acleddata.com/2020/09/03/demonstrations-political-violence-in-america-new-data-for-summer-2020/>. Accessed 25th Nov 2020.

Takehita, Toshio. "Exploring the Media's Roles in Defining Reality: From Issue-Agenda Setting to Attribute-Agenda Setting." Chapter in *Communication and Democracy: Exploring the Intellectual Frontiers in Agenda* edited by Maxwell E. McCombs, Donald Lewis.

OpenStax, DNA Structure and Sequencing. OpenStax CNX. Feb 24, 2014
<http://cnx.org/contents/53bb4f0d-1c4a-4451-a97a-e24e2b9e7776@7>.

Yan F. (2020). "Media Construction of Social Reality." Chapter in *Image, Reality and Media Construction*. Springer, Singapore.

Thomas Theorem. *Oxford Reference*. Retrieved 8 Jun. 2020, from
<https://www.oxfordreference.com/view/10.1093/oi/authority.20110803104247382>.

S. Cardenas and S. F. Vallejo-Cardenas, "Continuing the Conversation on How Structural Racial and Ethnic Inequalities Affect AI Biases," 2019 IEEE International Symposium on Technology and Society (ISTAS), Medford, MA, USA, 2019, pp. 1-7, doi: 10.1109/ISTAS48451.2019.8937853.

Searle, J. R., & Willis, S. (1995). *The construction of social reality*. Simon and Schuster.