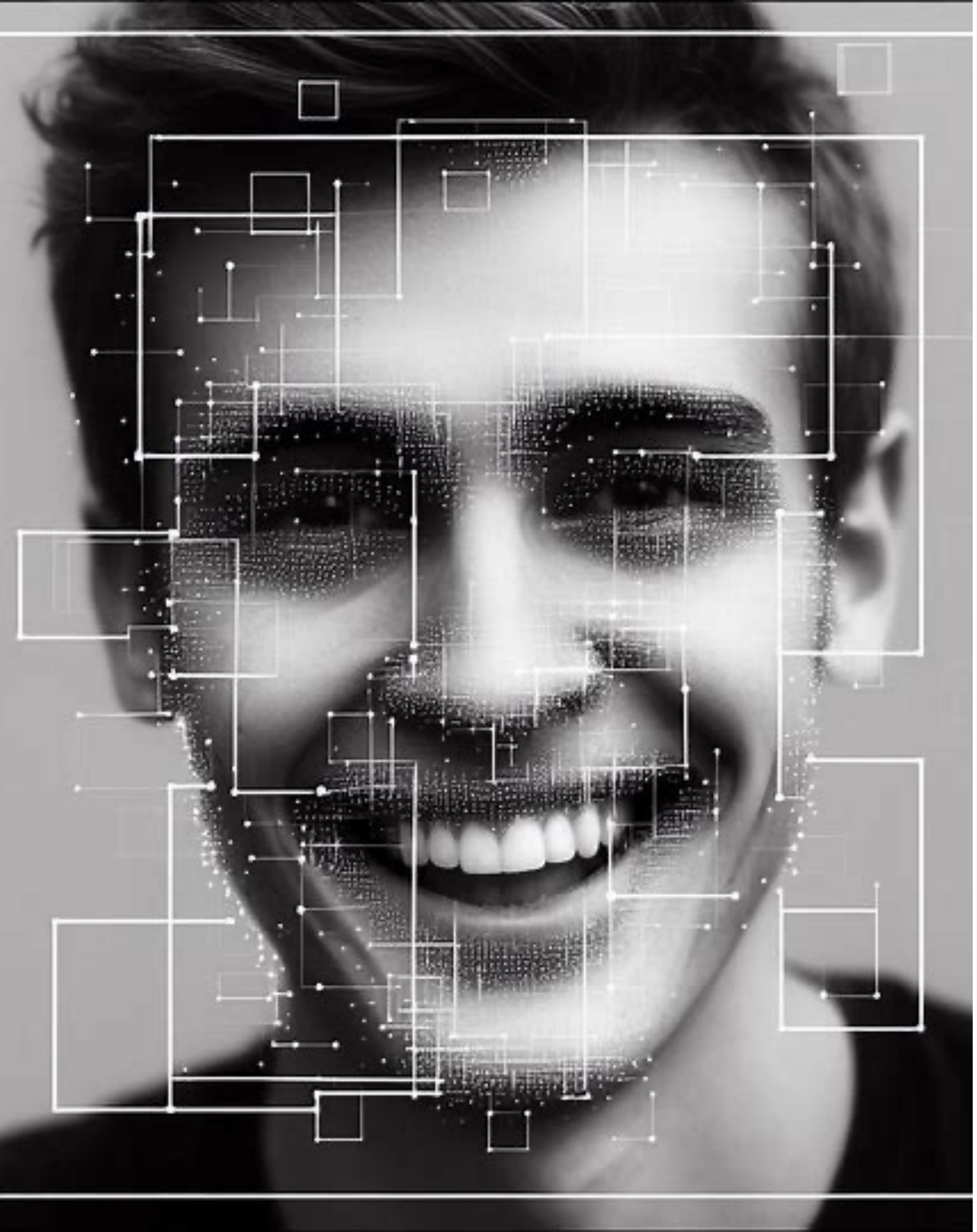
MoviEmotion

CAPSTONE

MIT PROFESSIONAL EDUCATION

cameron357@gmail.com

July 19, 2024



AGENDA

MoviEmotion

Problem to Solve

Problem Solution

Technical Solution

ROI & What’s Next

MoviEmotion

PROBLEM TO SOLVE

Happiness is important, today more than ever.
Entertainment creates **happiness**.

Movie script writers, both human and AI, need to know:

Which moments hold highest resonance?
Which patterns of resonant **moments** create the best viewer engagement?

The right sequence of resonant moments creates the secret recipe to blockbusters and sleeper successes.

cameron357@gmail.com

Ryan Reynolds, what EXACT moments would those be?

PROBLEM TO SOLVE – AS DESCRIBED BY RYAN REYNOLDS RE: UPCOMING DEADPOOL 2024 RELEASE

variety.com

VARIETY

Subscribe

"Well, I'm not saying that other people should do this, but my nine-year-old watched the movie with me and my mom, who's in her late 70s, and it was just was one of the best moments of this whole experience for me," he said. "Both of them were laughing their guts out, were feeling the emotion where I most desperately hoped people would be."

"Deadpool and Wolverine" is set for release on July 26 from Disney.

Read More About: Deadpool, Deadpool and Wolverine, Ryan Reynolds

cameron357@gmail.com

EXECUTIVE SUMMARY

MoviEmotion

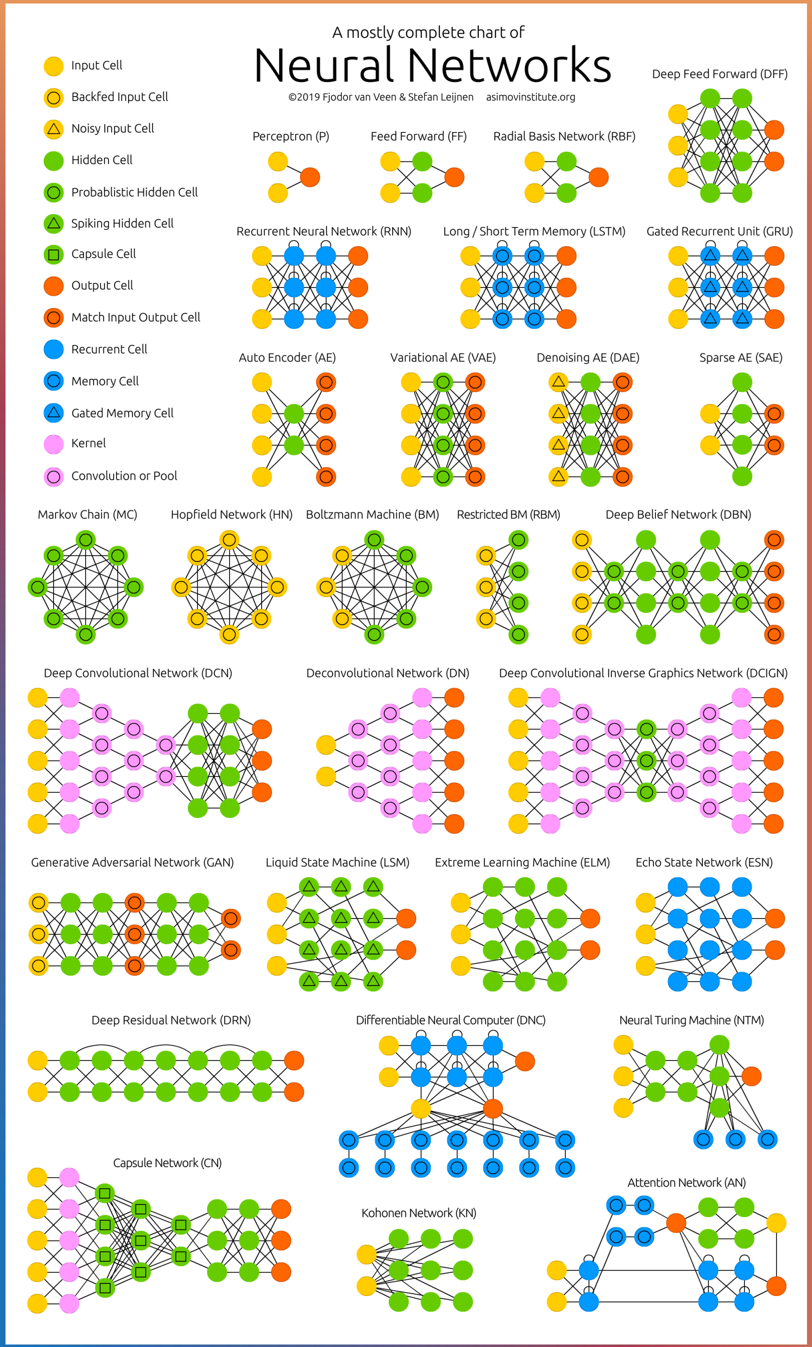
PROBLEM SOLUTION

cameron357@gmail.com

- Map a **movie script**, second by second, to the focus group's **timeline of feels**.
- Define the pattern of resonant moments that enable a studio to easily out-Hallmark a Hallmark channel holiday movie.
- **Done**! An emotion-detection neural network model trained on **> 15,000** images to an accuracy of 94%.

accuracy on
new test images.

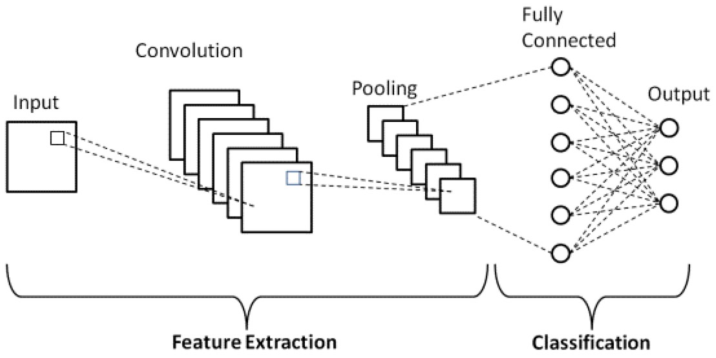
- 36 experiments, aka "training iterations"
- 40-70 epochs per iteration
- Each model training iteration took 45 min - 3 days
- ROI calcs based on Nvidia Tesla A100 GPU and Google Colab Pro+



https://www.asimovinstitute.org/neural-network-zoo/

TECHNICAL SOLUTION

LET'S DIVE IN



A **Convolutional Neural Network** (CNN) is the best neural network architecture for an image recognition problem. 'Best' means:

Highest **accuracy**. Lowest **computational costs**

High accuracy via:

☑ Spatial locality ☑ Translational invariance

Low computational cost via:

☑ Weight sharing between filters

RECOMMENDATION: CNN MODEL # 36

MoviEmotion's Emotion Recognition Architecture

cameron357@gmail.com

accuracy

2 x 2 --> 3 x 3 filter size on first two blocks increased accuracy +3%

Increasing filter size to 3x3 on later blocks decreased accuracy.

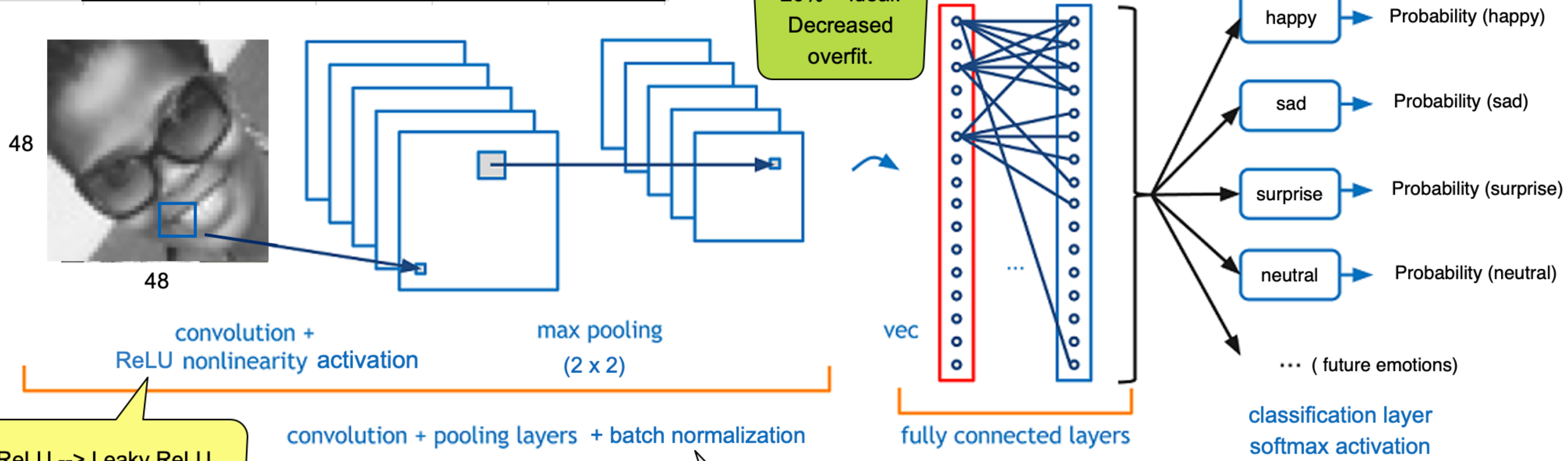
Adding third dense neuron layer increased accuracy on new test data 93 --> 94%

Convolutional Block	1	2	3	4	5
Filters	128	256	1024	1024	256
Filter (kernal) size	3 x 3	3 x 3	2 x 2	2 x 2	2 x 2

Dense Layer	1	2	3
Neurons	256	512	1024
Dropout	20%	20%	20%

Tried 30% --> Too much dropout. Decreased accuracy -12%.

20% = ideal. Decreased overfit.



ReLU --> Leaky ReLU decreased accuracy 88% to 78%.

No observable change

Legend: Good change | Bad change

36 EXPERIMENTS ON MY BIONIC CNN

Good Change
Bad Change

MoviEmotion: Mapping the Timeline of Feels vs a Movie's Script

cameron357@gmail.com

#	Hyperparameter	From	To	Result on TestAccuracy	Comment
1	Rectified Linear Unit (ReLU) Activation type on all layers `model.add(Activation('relu'))`	Leaky ReLU	ReLU	+ 10% .78 --> .88	**Activation** adds non-linearity to the neuron values. Without activation, a neural network would just act like a linear regression model. **ReLU activation** makes y = 0 if x < 0. This means negative values don't get propagated. Leaky ReLU has a tiny negative y slope when x is negative.
2	ReLU on Convolutional blocks Leaky ReLU on Dense Layers	ReLU Dense only	Leaky ReLU Dense only	- 4% .88 --> .84	Some research shows that Leaky ReLU can be better on dense layers. But not for this model!
4	Batch Normalization location BN makes inputs have mean=0, stdev=1	After ReLU	Before ReLU	- 10% .88 --> .78	Watched Ian Goodfellow's YouTube. Decided to experiment with Batch Normalization location.
5	ReduceLROnPlateau `min_delta = 0.0001`	.0001	.00001	- 3% .88 --> .85	Learning Rate on Plateau is a compile section callbacks list parameter. This slooowed the Epochs.
8	Change Optimizer to Stochastic Gradient Descent (SGD)	Adam	SGD	- 13% .88 --> .75	A neural network's optimizer adjusts weights and learning rates at start of back propagation.
9	Increase Filter size in ALL Convolutional blocks	2 x 2	3 x 3	- 4% .88 --> .84	Each dataset image is only 48 x 48. 3 x 3 is too big a filter size for all 5 conv blocks.
10	Increase Filter size in Convolutional blocks 1 - 3	2 x 2	3 x 3	+ 3% .85 --> .88	Keep final 2 convolutional blocks with small 2 x 2 filters are needed for fine detail detection.

36 EXPERIMENTS ON MY BIONIC CNN

Good Change

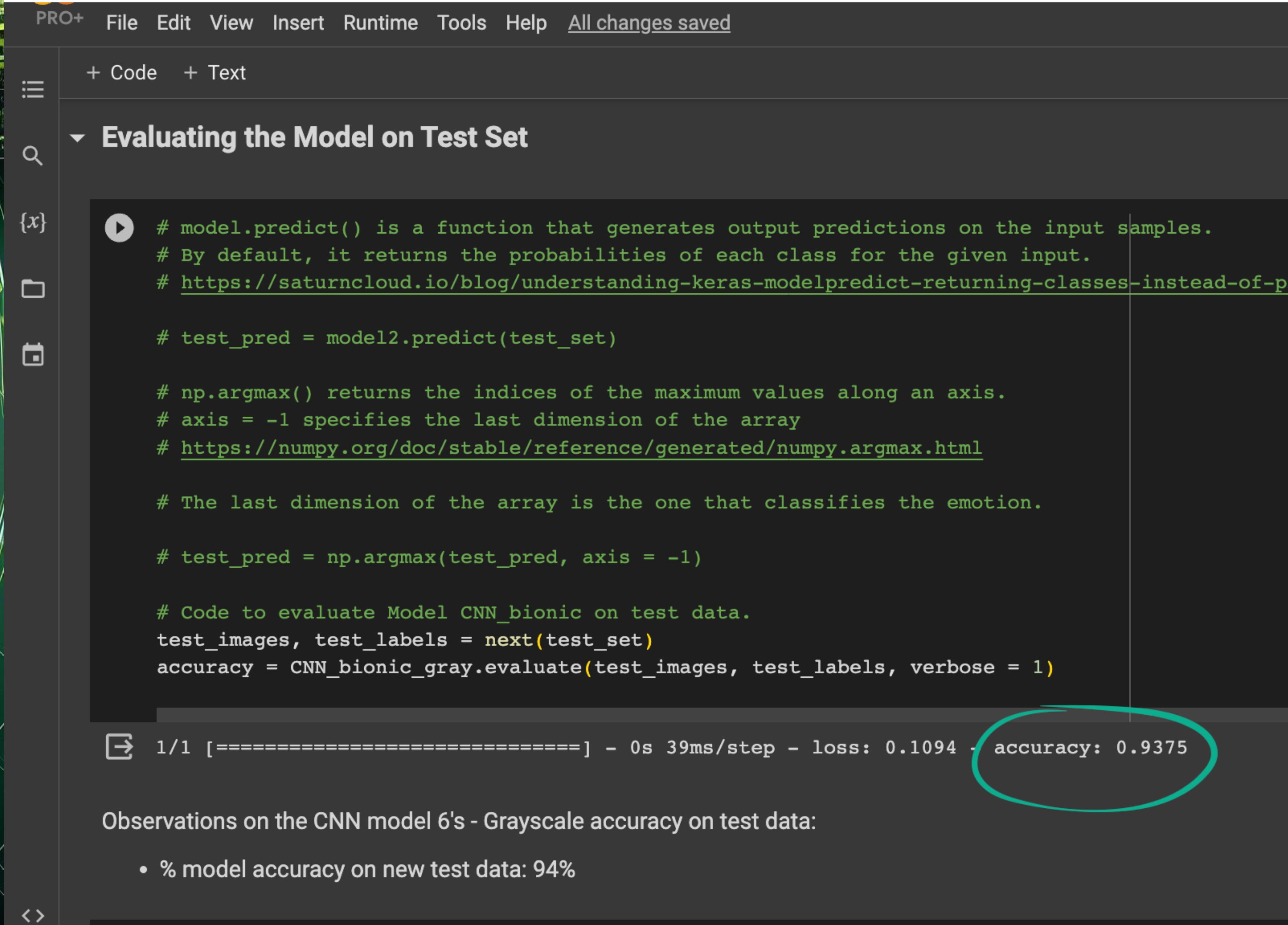
Bad Change

MoviEmotion: Mapping the Timeline of Feels vs a Movie's Script (2)

cameron357@gmail.com

#	Hyperparameter	From	To	Result on TestAccuracy	Comment
12	Slow the Optimizer Learning Rate `optimizers.Adam(learning_rate=.0001)`	.001	.0001	+ 4% .84 to .88	Optimizer learning rate determines how fast the model moves towards a loss minima. I also increased epochs from 40 to 60.
13	Cut batch size in half `model.fit(batch_size = 64)`	128	64	- 7% .88 --> .81	This smaller batch size: 1) Provided a worse estimate of the gradient 2) Gave fewer 'clues' to where global loss minima is located.
21	Increase Validation Split `model.fit(validation_split=0.3)`	20%	30%	+ 3% .88 --> .91	Increasing validation split to 30% helped! **Finally**: Test accuracy over 90% !
23	Increase Validation Split even more `model.fit(validation_split=0.4)`	30%	40%	- 10% .91 --> .81	40% validation split: too much. Goldilocks zone: 30%. Not 20. Not 40.
25	Lots of Data Loader Augmentation ex: increase range of brightness parameter	(0.,2.)	(0.7.,2.)	- 19% .91 --> .72	Widening the image brightness level range applied by data loader did NOT work.
27	L2 Regularization `model.add(kernel_regularizer=(0.01))`	none	0.01	- 35% .91 --> .56	Adding L2 regularization decreased accuracy. -35% (!). Minimal change to fit.
31	Increased neural network dense layer widths	128, 256	256, 512	+ 2% .91 --> .93	Good changes: More neurons per layer. More convolutional filters per block.
*36	Add a third fully connected dense layer `model.add(Dense(1024))`	2	3	+ 1% .93 --> .94	Adding a third final layer increased test accuracy to 94%. Rah!

BEST TEST ACCURACY
> 36 EXPERIMENTS



TRANSFER LEARNING

"IT SEEMED LIKE A GOOD IDEA"

cameron357@gmail.com

Transfer Learning holds the promise of re-using what a pre-trained model already knows.
Better accuracy. Less work.

However, 36 experiments later:
TL model accuracy < my custom model

Why did Transfer Learning **not** work?

- 💣 Color Mode mismatch `rgb vs grayscale`
- 💣 Domain mismatch `object images vs face images`
- 💣 Image dimension mismatch `224x224 vs 48x48`

```
Ex: Oxford University's Visual Geometry Group (VGG-16) model
imported at   transfer_layer = vgg.get_layer('block3_pool')
test data accuracy: just 73%
```

DATA QUALITY

CLEAN DATA
HAPPY SCIENCE

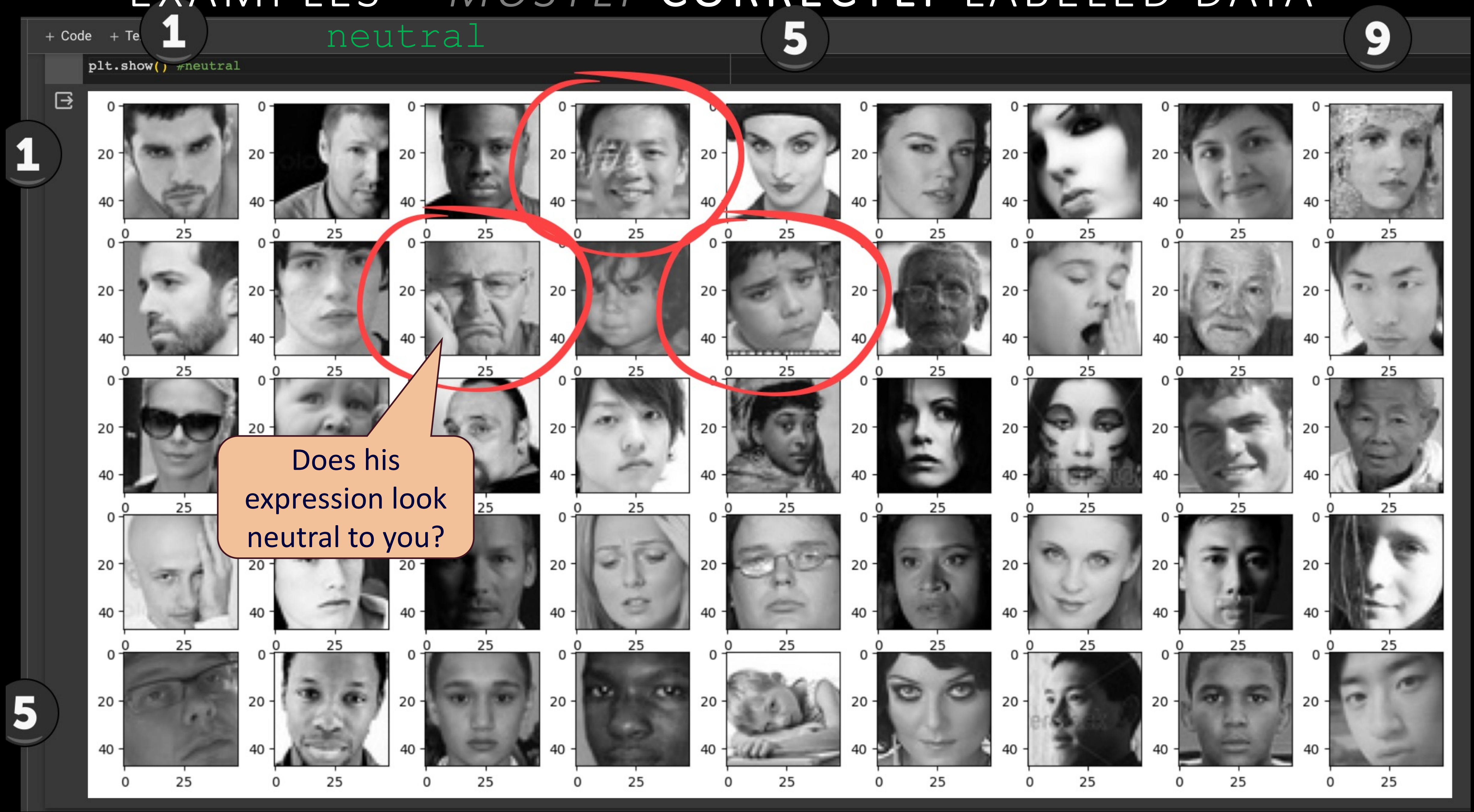
⏭ Next: The next data set must be fully cleaned before model training for the next epic.

Exploratory Data Analysis on > 15,000 images showed:

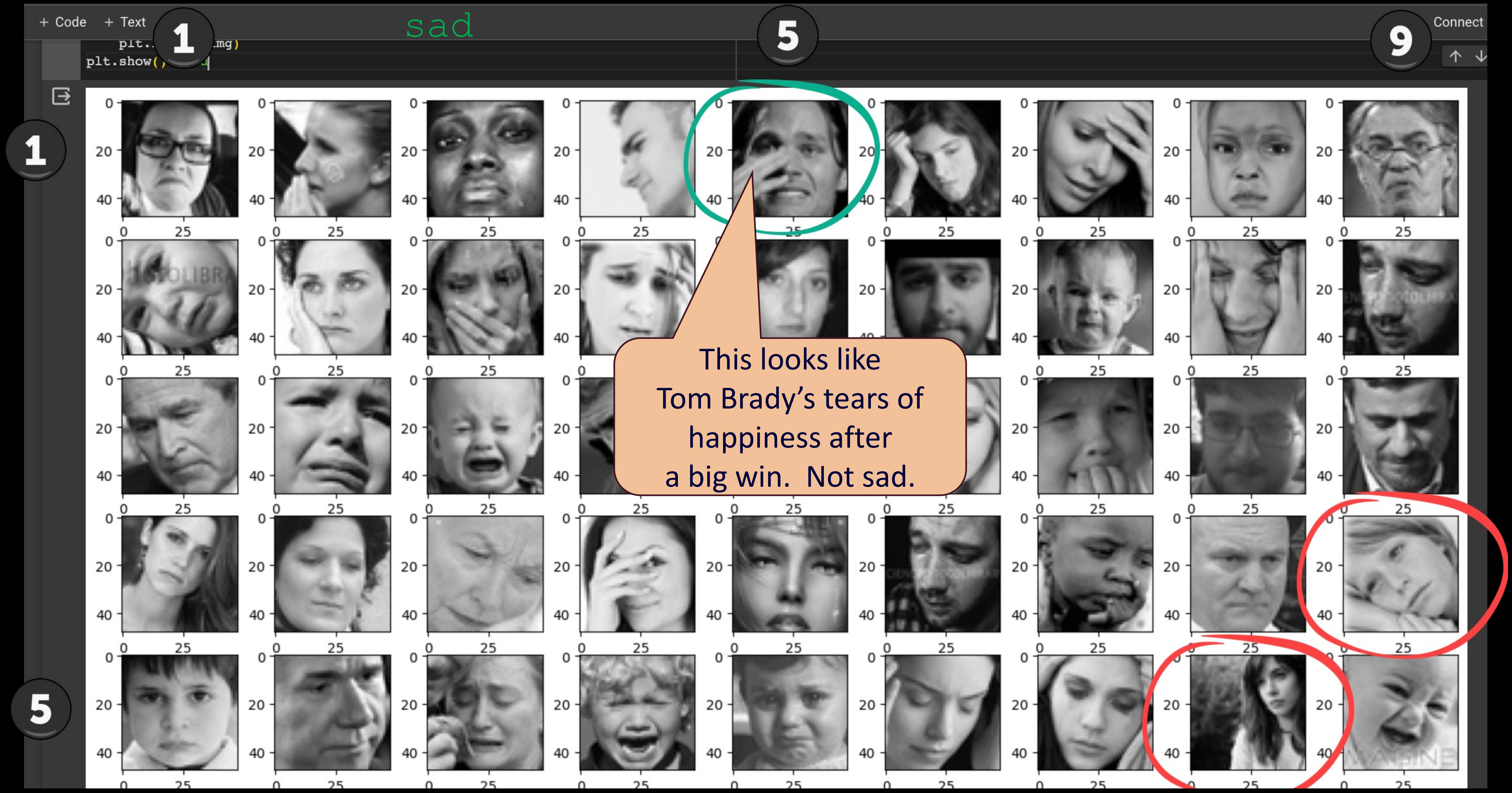
- Misclassified images
- Corrupted data - Glyphs or solid black / white images

💣 Result: Unhelpful patterns trained my model.

EXAMPLES – *MOSTLY* CORRECTLY LABELED DATA



EXAMPLES – *MOSTLY* CORRECTLY LABELED DATA

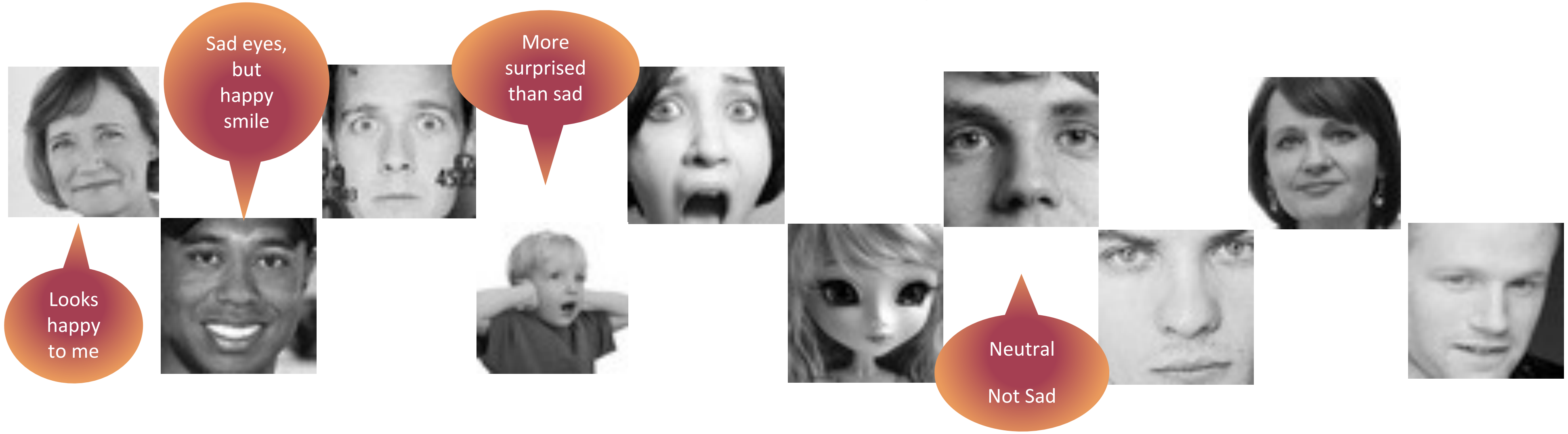


DATA SHENANIGANS

INCORRECTLY LABELED DATA

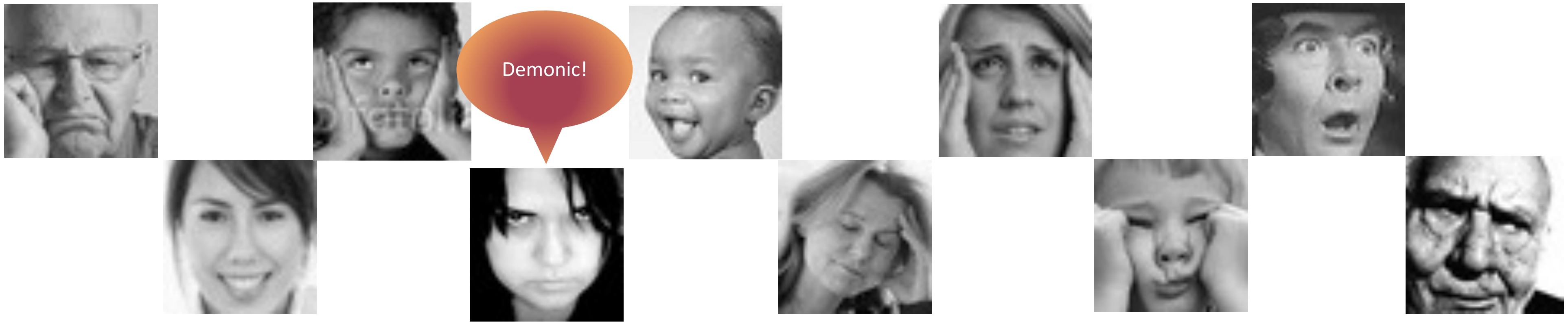
How would **you** label these images?

Labeled
‘sad’



If humans debate which emotion is being felt, how do we expect the machine to learn from it?

Labeled
‘neutral’



DATA SHENANIGANS

INCORRECTLY LABELED DATA (2)

How would **you** label these images?

Labeled 'happy'





Raised eyebrows --> identical to many surprised images.

Labeled 'surprise'

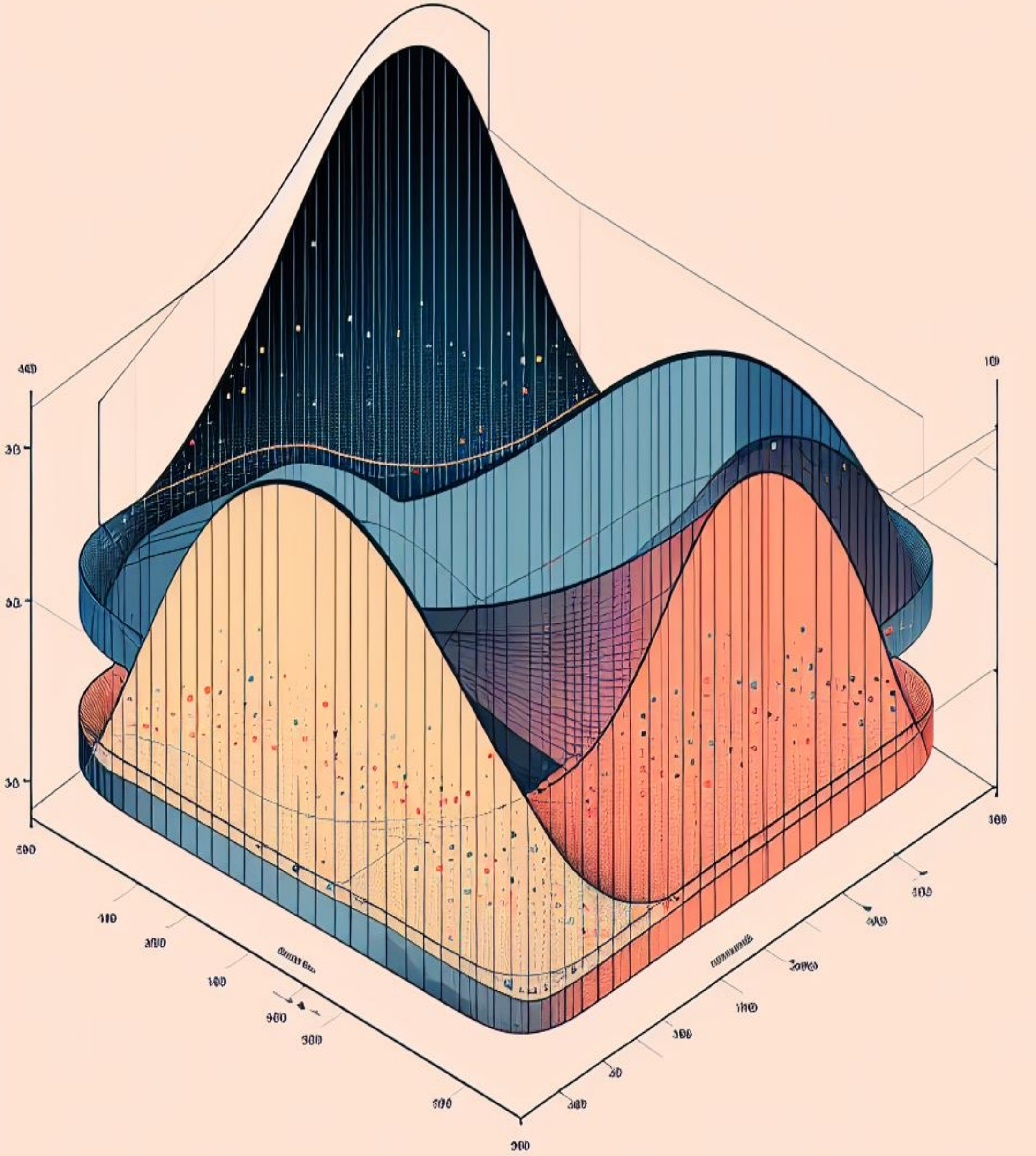




Looks identical to many happy images

Non-Facial images



MACRO OBSERVATIONS

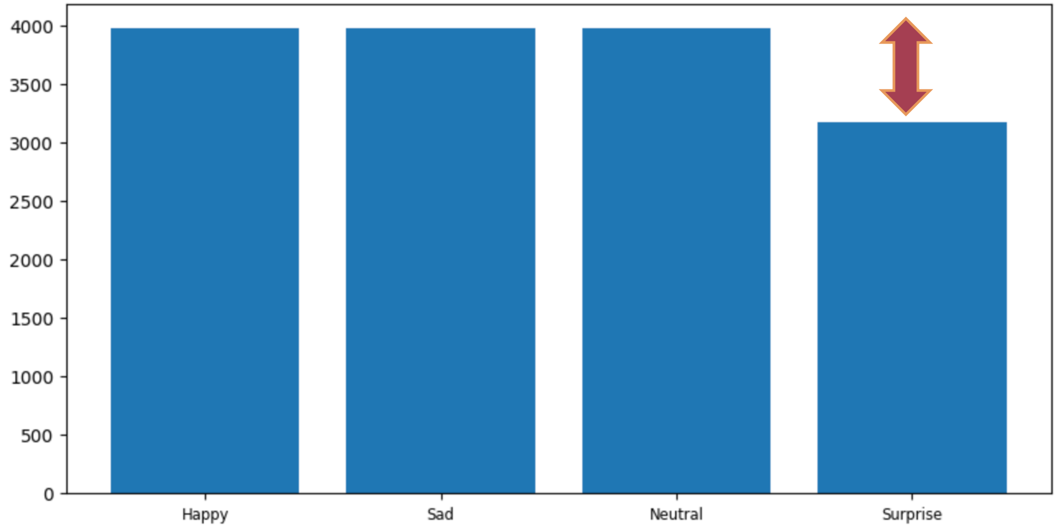
“EVIDENCE SPEAKS LOUDER THAN INTUITION” - STATSIG

Exploratory Data Analysis (EDA) also showed:

👁 Fewer surprised images

21% surprised vs 26% of total for each other emotions

Ideal: even split of classification category data



👁 Not enough **test** images.

Test dataset was just 0.6% of total

	Train	Validate	Test
Ideal %	70	20	10

I can address both problems in next sprint.

RETURN ON INVESTMENT (ROI)

CONSTRAINT: POSITIVE ROI WITHIN 6 QUARTERS

$$ROI = \frac{\text{Net investment gain}}{\text{Cost of investment}} \times 100$$

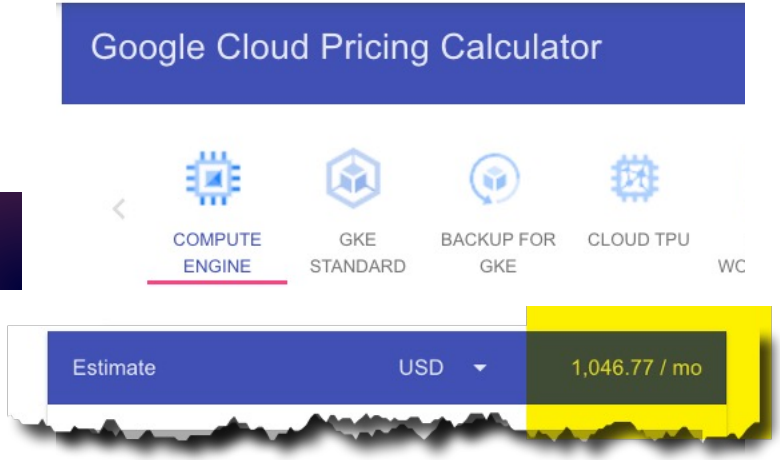
I will collaborate with your Financial Analyst to define tangible and intangible types of :

- Benefits
- Costs

for the ROI spreadsheet. Specific focus will be on accuracy of first 6 quarters' cash flow.

ROI – COSTS

CALCULATED - COMPUTE



Without a fast GPU, model training epochs becomes sloggy slow:

Training the model once (60 epochs):

1 Nvidia Tesla A100 GPU

vs

1 standard Google Cloud CPU **2 DAYS**

ROI – COSTS – SOFTWARE & COMPUTE

COMPUTE COSTS WILL BE MANAGEABLE

94% accurate Bionic Model #36
creation costs to date**:**

$110: $100 Google Colab Premium + $10
($10 additional compute credit purchased after I exceeded the premium subscription GPU time limit)

--> Assume $5,000 / month first year model costs

- Inference costs from ongoing use will be small vs initial training costs.
- Even 'huge' movie focus groups present a **tiny** workload compared other types of business problems. Yay!

Other tangible variable costs to estimate:
Gathering Data, Data Prep, Data Wrangling, Data Analysis, Data Transmission, Model Training, Model Testing, Model Deployment, Model Monitoring

Our friendly Financial Analyst will add to ROI the tangible fixed costs (Salaries, Benefits, Slack, Zoom,..)

ROI - COSTS

BEYOND SOFTWARE AND COMPUTE

Other **Tangible Costs** (continued):

- Device and installation cost to retrofit focus group screening theater with embedded seat back cameras.
- Legal fees to ensure full disclosure is well done to screening attendees.
- Integration of attendee terms and conditions acceptance into studio’s focus group app.

Intangible Cost:

- Opportunity cost of prioritizing this project vs other projects we could do

ROI - BENEFITS

NEED A GOOD 'CLOSER' ON THE STUDIO CONTRACTS

Go to Market (GTM) will target a complete solution service to the top 7 movie studios:

Walt Disney Pictures ($83 billion), Paramount (30), Lionsgate (36), MGM (13), Universal (12), Sony (7)

Tangible Benefits

Cash flows from revenue must 10x the final burdened tangible costs over a contract period of "$ ____ per season of movies"

Intangible Benefits

- Experience with implementing a verticalized AI application involving both hw and sw
- Brand awareness & goodwill via media attention

WHAT'S NEXT: PRIVACY

LET'S NOT BE CREEPY

- Complete, transparent, easy to skim disclosure.
- No personally identifiable information added to the image
- Image deletion after a defined period

RISKS

BIGGEST UNKNOWNS

- Focus group theater seat camera installation costs. Need studios to cover.
- Could we better position this solution? Have we figured out the studios' biggest pain point we can solve with it?

WHAT’S NEXT

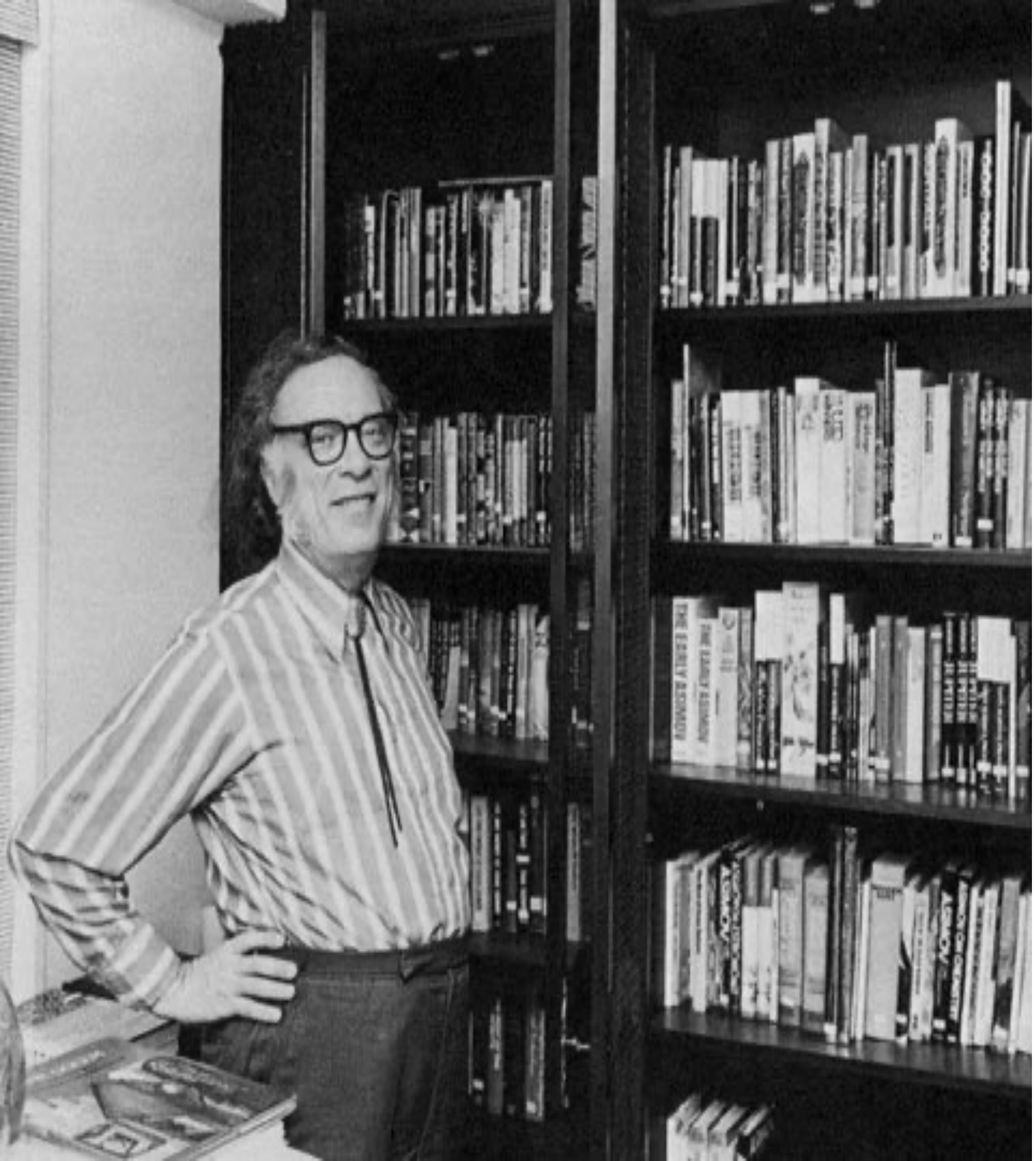
WHAT’S THE BIGGEST PAIN TO BE SOLVED?

- Listen hard to movie studio first impression : What is the **biggest insight** this can reveal for them? Audience sentiment flow and pattern? Or something else?
  Could **Trailer reaction patterns** be predictive? **Usefully** predictive? How?
- Consider defining a per-person **“neutral” baseline** from which to measure emotion. One person’s “mildly surprised” is another person’s “shocked”.
- Add **strength** of emotions
- Add more emotion **categories**.
  “tears of joy”, “happily surprised”

CREDITS

PEOPLE WHO INSPIRED

MIT Professors	Munther Dahleh, Stefanie Jegelka, Devavrat Shah, John Tsitsiklis, Caroline Uhler
NYU Professor	Trush Majmudar
Learning Mentors, Instructors, SMEs	Cristiano Aguiar, Benjamin Artuso, Jacob Curtis, Matthew Graziano, Scott Howard, Georg Huettenegger, Chris Marciniak, Sayed Mohammad Parvasi, Fred Premji, Sanat, Abhinanda Sarkar, Bhaskarjit Sarmah, Greg Schoeninger, D. Sculley, Akshey Singhal, Bradford Tuckfield, Davood Wadi
Colleagues	Josh Bode, Nathanial Brewer, William Cole, Nadiya Dragneva, Ameche Egbe, Rama Garg, Mike Gerges, Yvan Giroud, Stephen Lenzen, Aniruddha Mandrill, Zara Mubeen, Jennifer Northrup, Evan Pierce, Rama Rao, Joseba Ruiz, Tarachand Sahoo, Harp Singh, William Stanislaus, Jeff Stearns, Eliana Wassermann, Shyam Yadati
Friends	Matt Kenney, Mike Lake



CREDITS

CREATORS OF LEARNING

- **3 Blue 1 Brown**
  (excellent YouTube)
- Arxiv
- Bard
- Bing Image Creator
  (Most images created by Bing)
- Dall-e Image Creator, OpenAI
- Data Base Camp (German)
- Data Bricks
- **Geeks For Geeks**
  (best Python quick reference)
- **GitHub**
- IIETA
- Jcchouinard
- **Kaggle**
- Learn Data Sci
- **Machine Learning Mastery**
- MDPI
- **Medium**
- **Neptune.ai**
- Neural Network Zoo
- **OXEN.AI**
- **Phind** (Killer helpful tool. Thank you Bhaskarjit Sarmah for that tip!)
- Real Python
- Saturn Cloud AI
- Springer
- Stack Exchange
- Stack Overflow
- **StatQuest** (excellent YouTube)
- TensorFlow
- **Towards Data Science**

Photo: Isaac Asimov

APPENDIX

Thank you!