Drawing: A New Way To Search

Nguyet Minh Phu, Connie Xiao, Jervis Muindi
{minhphu, coxiao, jmuindi}@stanford.edu

OVERVIEW

Motivation: Using words can be limited when communicating across cultures and literacy levels. Images are a shared medium of communication that can beneficially bridge those divides.

We want to develop an efficient system that recognizes labels of hand-drawn images based on Google's QuickDraw dataset. We implemented a variety of models and found that an altered CNN was best for this task.

DATA

Google's QuickDraw is the world's largest doodling dataset, consisting of hand-drawn images from over 15 million people all over the world.

Examples:

Label/Class: banana hockey stick squirrel watermelon bathtub

Data features: We used the npy bitmap version of the data. Each drawing consists of raw pixel inputs with values from 0 to 255. We are take advantage of the fact that each image has only two colors, black and white to binarize the pixels.

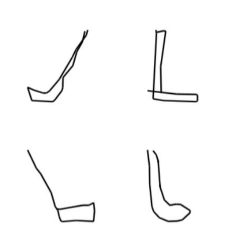
MODEL / RESULTS

LOGISTIC REGRESSION

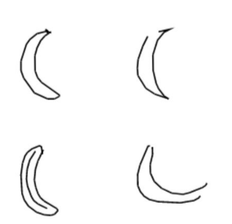
- For baseline, we used Logistic Regression, a simple and fast to train model using numpy bitmap of raw image pixels.

	Accuracy (%)		Training Time (s)	
Classes	10	50	10	50
Baseline	64.64	43.89	122	1089

Table 1. Results for linear regression



Hockey Stick Doodles



Banana Doodles

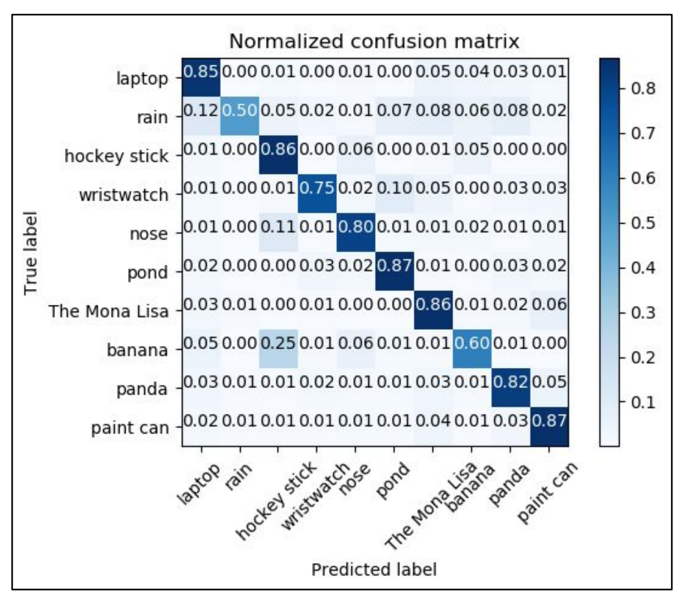


Figure 1. The confusion matrix for linear regression

- Linear Regression performs relatively well
- Banana is often confused with hockey stick which shows that there is a need for a more sophisticated model to make up for drawing quality

SUPPORT VECTOR MACHINE

- In Support Vector Machine with Kernel, some kernels may be more suited for the task of doodle classification, thus we implemented a SVM with four different kernels (Linear, RBF, Polynomial, Sigmoid) to identify the best one for this task empirically.

Model on 10 classes	Accuracy (%)	Training Time (s)
Linear Kernel	22.22	1831
RBF Kernel	61.01	2842
Polynomial Kernel	50.89	6673
Sigmoid Kernel	11.72	6971

Parameters: polynomial degree of 5, RBF coefficient of 1 and RBF gamma of 1, sigmoid coefficient of 1

Table 2. Results for SVMs

- Surprisingly SVMs performed worse than linear regression overall
- Suspect it is due to lack of parameter tuning: For the sigmoid kernel, if the chosen parameters are not well tuned, the algorithm can perform worse than random [1]
- Simultaneously, we found that our CNN was performing with an acceptable accuracy so we decided to focus on CNNs

CONVOLUTIONAL NEURAL NET

- A doodle is a simple image, thus some components of a CNN may be removed. We implemented a CNN, then simplified it by progressively removing layers and dense units to analyze the impact on accuracy and runtime.

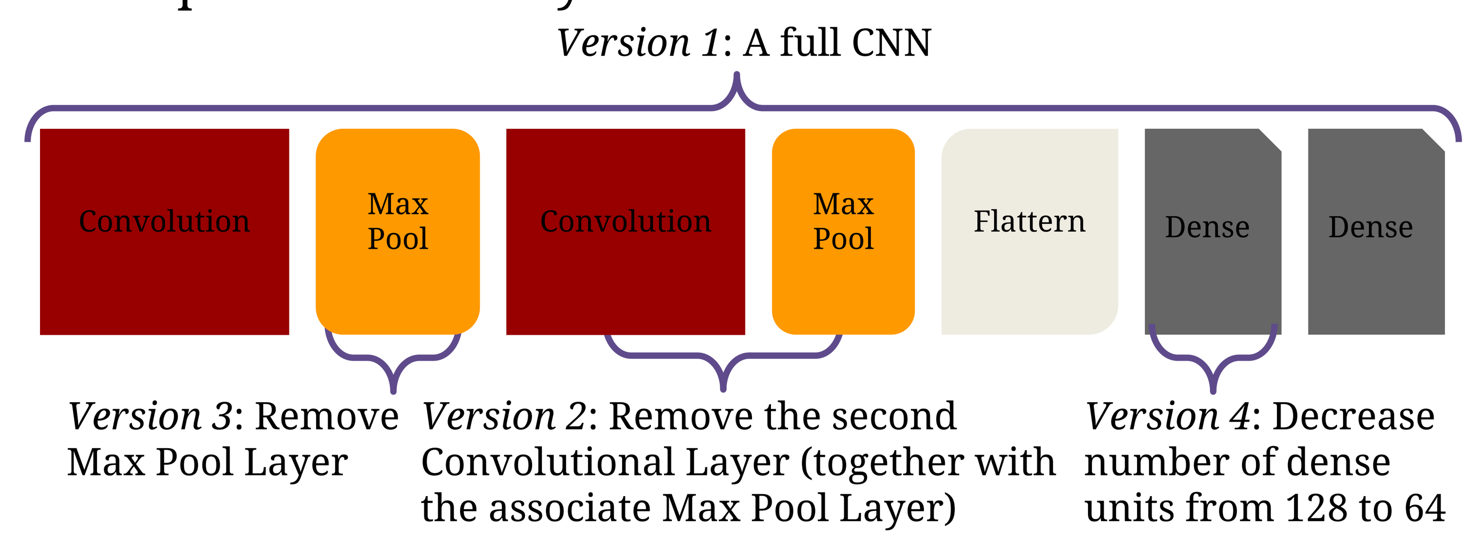


Figure 2. A sketch of how we progressively simplified CNN for doodle classification

	Accuracy (%)		Training Time (s)		Binarized
Number of classes	10	50	10	50	
v1: full CNN	86.59	82.12	3047	14949	No
	83.86	77.28	5391	13856	Yes
v2: remove 2nd convLayer	85.5	76.1	2450	11524	No
	82.16	70.27	2332	16617	Yes
v3: v2 + remove 1st maxPool	86.55	77.17	560	2455	Yes
v4: v2 with a dense layer of 64 units	85.6	70.29	628	3043	Yes

Table 3. Results for CNNs

TRANSFER LEARNING

- Training a deep-learning model from scratch is time-intensive. Transfer learning is one way to leverage pre-trained models for our task. We explored whether using pre-trained winning models from the ImageNet competition could help save time and improve accuracy for our task of doodle classification.

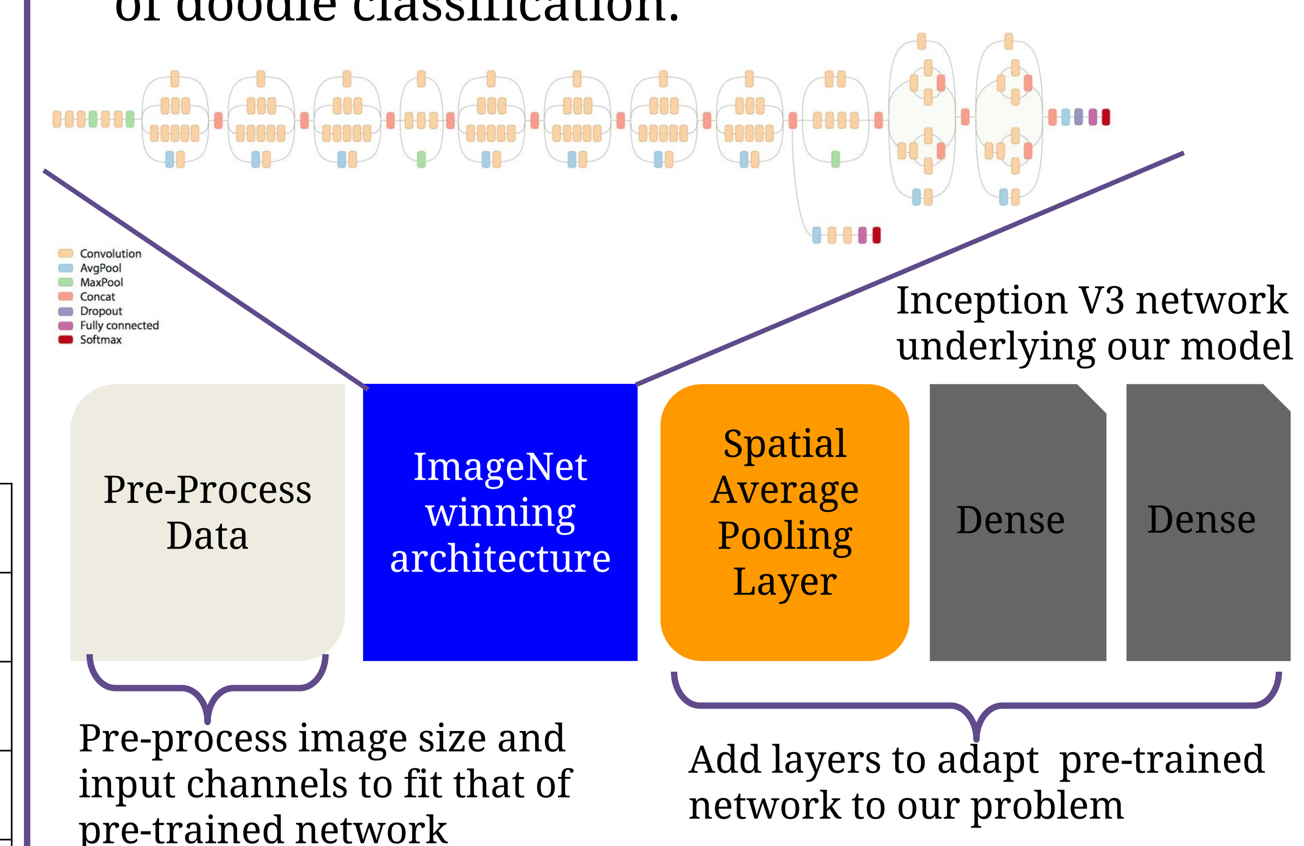


Figure 3. A sketch of how we carried out transfer learning

Models on 3 classes	Accuracy (%)	Training Time (s)
Inception v3	48.77	(stopped at 20 epochs)
MobileNet	75.4	10290
VGG	N/A	(Ran out of memory)
ResNet50	62.72	15232

Table 4. Results for transfer learning

DISCUSSION

Training Time Vs Accuracy tradeoff

- Judging purely on the basis of training time / training resources, Logistic regression was the fastest with a training time of about half an hour.
- However, assuming one has more resources available to train more sophisticated models, we find that CNN does the best for this task with a training time of O(hours) ~ 4+ hours.

Transfer Learning

- Transfer learning model was also expensive to train/tune due to the deep nature of the source models that we were only able to run it on 3 classes.
- Getting Transfer learning to work well likely requires more extensive model tuning.
- In our case, we find transfer learning not optimal if the goal is to optimize training time and accuracy.

FUTURE WORK

- Develop our most promising approach: More extensive experiments to determine the effect of each layer in CNN
- Explore a different approach of transfer learning: using transfer learning as a fixed feature extractor for logistic regression
- Develop other efficiency metrics: data efficiency - working on efficiency in conjunction with smaller datasets

REFERENCES

[1] R. Amami, D. B. Ayed, and N. Ellouze. Practical selection of svm supervised parameters with different feature representations for vowel recognition. arXivpreprint arXiv:1507.06020, 2015.
[2] Simon Kornblith, Jonathan Shlens, and Quoc V. Le. Do better imagenet models transfer better? CoRR, abs/1805.08974, 2018. URL http://arxiv.org/abs/1805.08974