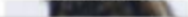


```
In [11]: data.show_batch(rows=3, figsize=(7,6))
```

miniature\_pinscher



leonberger



japanese\_chin



pug



havanese



```
In [13]: learn = cnn_learner(data, models.resnet34, metrics=error_rate)
```

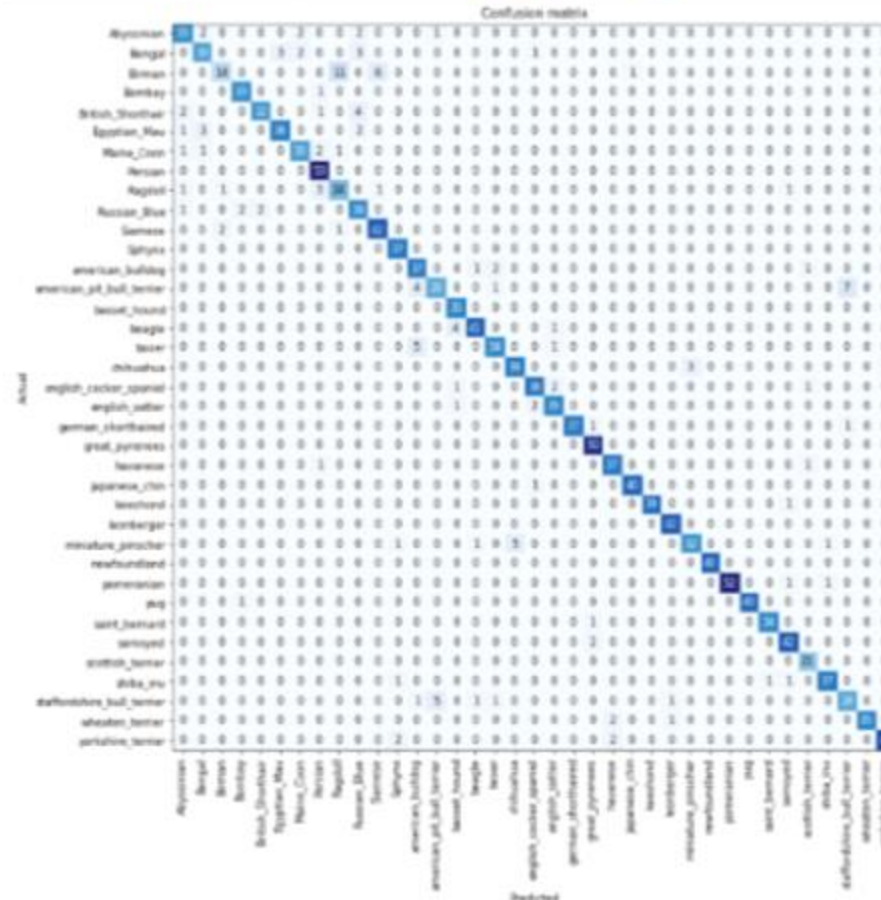
```
In [14]: learn.model
```

```
Out[14]: Sequential(
  (0): Sequential
    (0): Conv2d(3
    (3, 3), bias=Fals
    (1): BatchNorm
    _running_stats=Tr
    (2): ReLU(inplace=
    (3): MaxPool2D
    ceil_mode=False)
    (4): Sequential
      (0): BasicBlock
      (conv1): Conv2d(
      adding=(1, 1), bi
      (bn1): BatchNorm
      track_running sta
      (relu): ReLU(inplace=
      (conv2): Conv2d(
      adding=(1, 1), bi
      (bn2): BatchNorm
```

```
In [15]: learn.fit_one_cycle
```

epoch	train_loss	valid_loss
0	0.811533	0.1
1	0.551540	0.1
2	0.422648	0.1
3	0.328013	0.1

```
In [20]: interp.plot_confusion_matrix(figsize=(12,12), dpi=60)
```

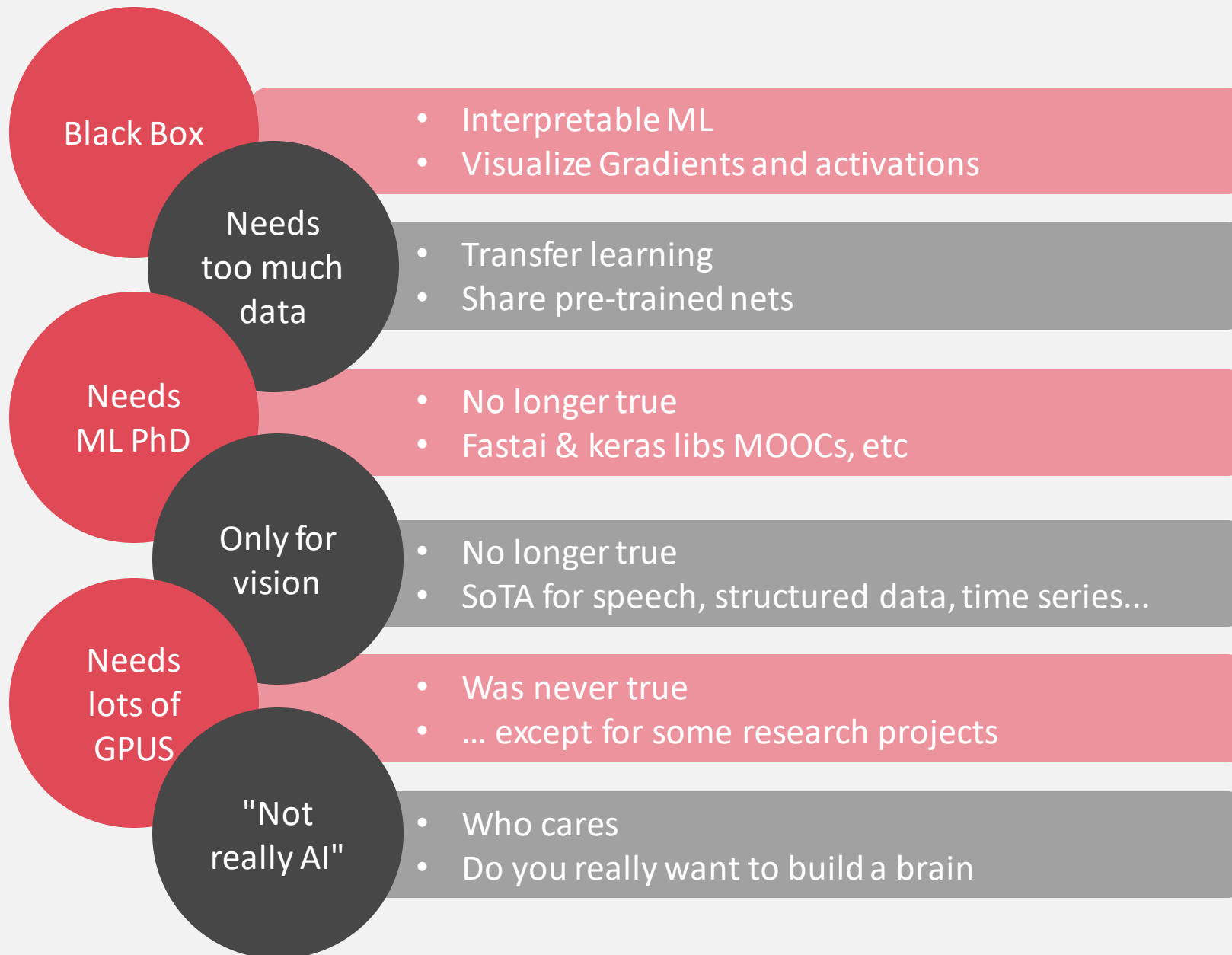


# fast.ai

Making neural nets  
uncool again



Josiah Laivins  
February 23, 2020



# Making Deep Learning Accessible

## **Software**

To make these available to use quickly, reliably, and with minimal code

## **Research**

Ways to make state of the art deep learning techniques more accessible

## **Community**

So that we can all help each other

## **Research**

So that as many people as possible use these

# What is fastai?



Popular ML library

One of the most popular programming languages

Different from classical programming. "**friendlier**".

[1] <https://pytorch.org/>

[2] <https://python.org/>

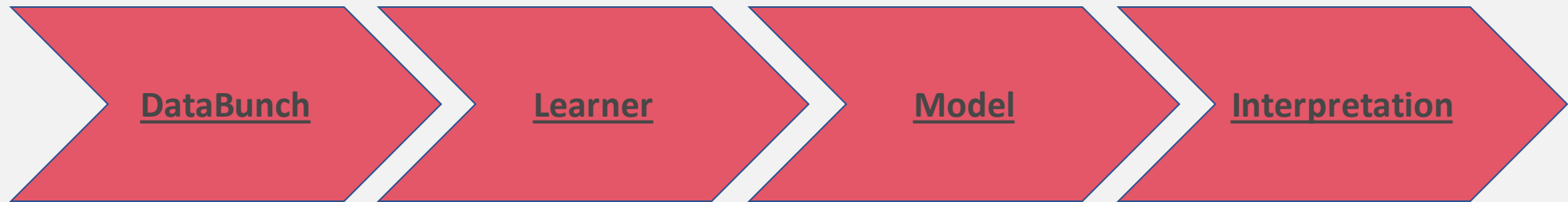
[3] <https://jupyter.org/>

## **DataBunch**

- Images, text, audio
- Data transformation
- Data analysis

## **Model**

- Neural net being trained
- Atomic and portable



## **Learner**

- Contains everything
- Says how and when things should happen

## **Interpretation**

- Post training quick analysis of model performance, troubleshooting

# Who is fastai for?

## **Newbie's**

Tied to a course for anybody to learn from

No math required to get into ML

Jupyter notebooks are far friendlier than most IDEs

## **Experienced Data Scientists**

Existing API code for pipelining

Easy comparison between other models

Attached course goes into detailed topics

Established codebase to build on top of

# fastai vision models

## Computer Vision models zoo

The fastai library includes several pretrained models from [torchvision](#), namely:

- resnet18, resnet34, resnet50, resnet101, resnet152
- squeezenet1\_0, squeezenet1\_1
- densenet121, densenet169, densenet201, densenet161
- vgg16\_bn, vgg19\_bn
- alexnet

On top of the models offered by torchvision, fastai has implementations for the following models:

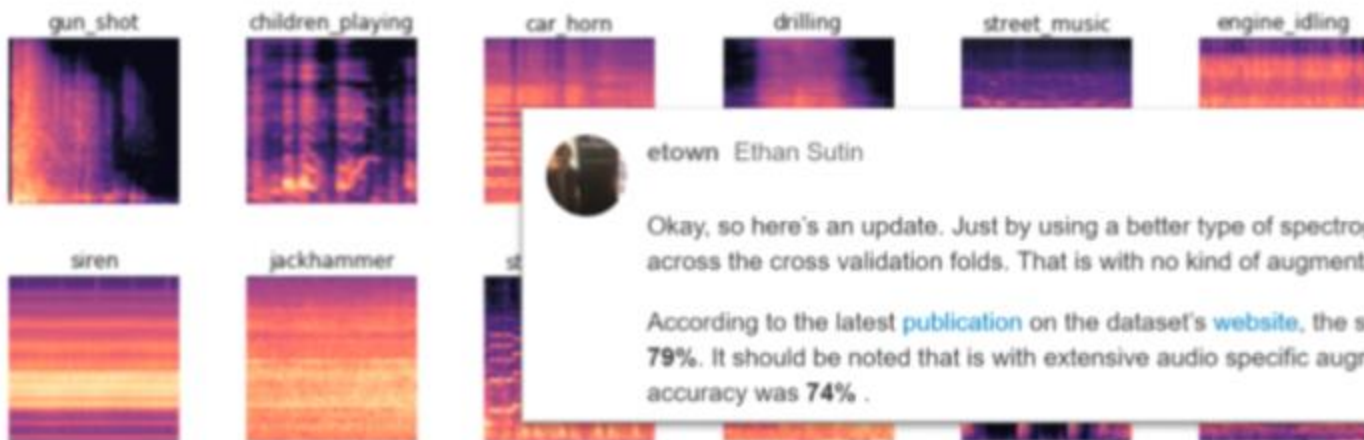
- Darknet architecture, which is the base of [Yolo v3](#)
- Unet architecture based on a pretrained model. The original unet is described [here](#), the model implementation is detailed in [models.unet](#)
- Wide resnets architectures, as introduced in [this article](#)

# Deep Convolutional Neural Networks and Data Augmentation for Environmental Sound Classification

Justin Salamon and Juan Pablo Bello

```
data = ImageDataBunch.from_folder(data_directory/'1', ds_tfms=[], size=224)
data.normalize(imagenet_stats)
```

```
data.show_batch(rows=6, figsize=(12,12))
```



etown Ethan Sutin

1 1 etown 1h

Okay, so here's an update. Just by using a better type of spectrogram, I was able to achieve **80.5%** accuracy across the cross validation folds. That is with no kind of augmentation at all.

According to the latest [publication](#) on the dataset's [website](#), the state-of-the-art mean accuracy achieved was **79%**. It should be noted that is with extensive audio specific augmentation, and without augmentation their top accuracy was **74%**.

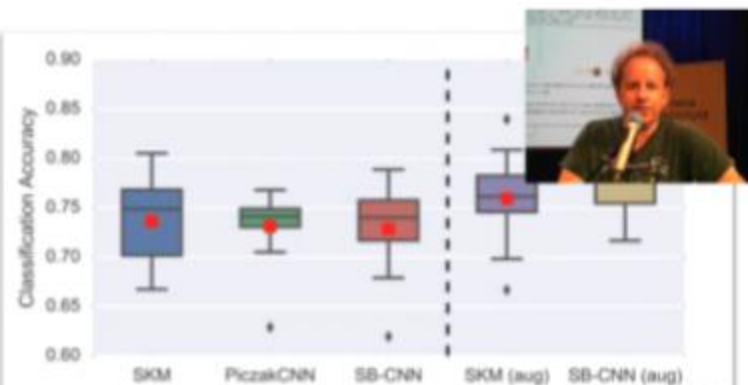


Fig. 1. Left of the dashed line: classification accuracy without augmentation – dictionary learning (SKM [7]), Piczak's CNN (PiczakCNN [11]) and the proposed model (SB-CNN). Right of the dashed line: classification accuracy for SKM and SB-CNN with augmentation.

[1]: <https://github.com/hiromis/notes/blob/master/Lesson2.md>

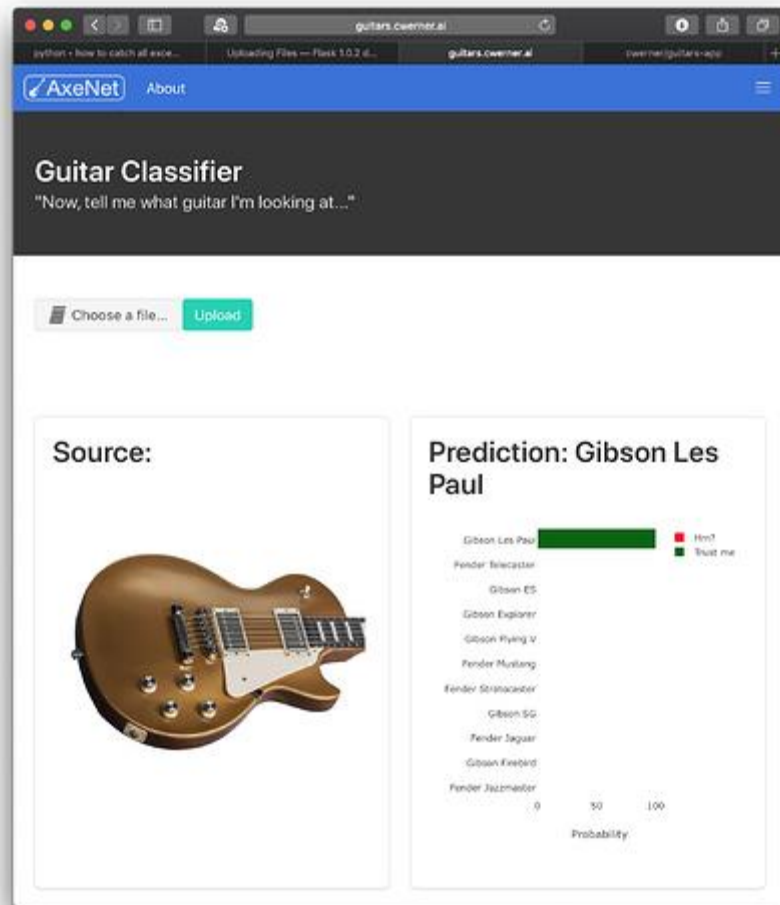
[2]: <https://forums.fast.ai/t/share-your-work-here/27676/215>



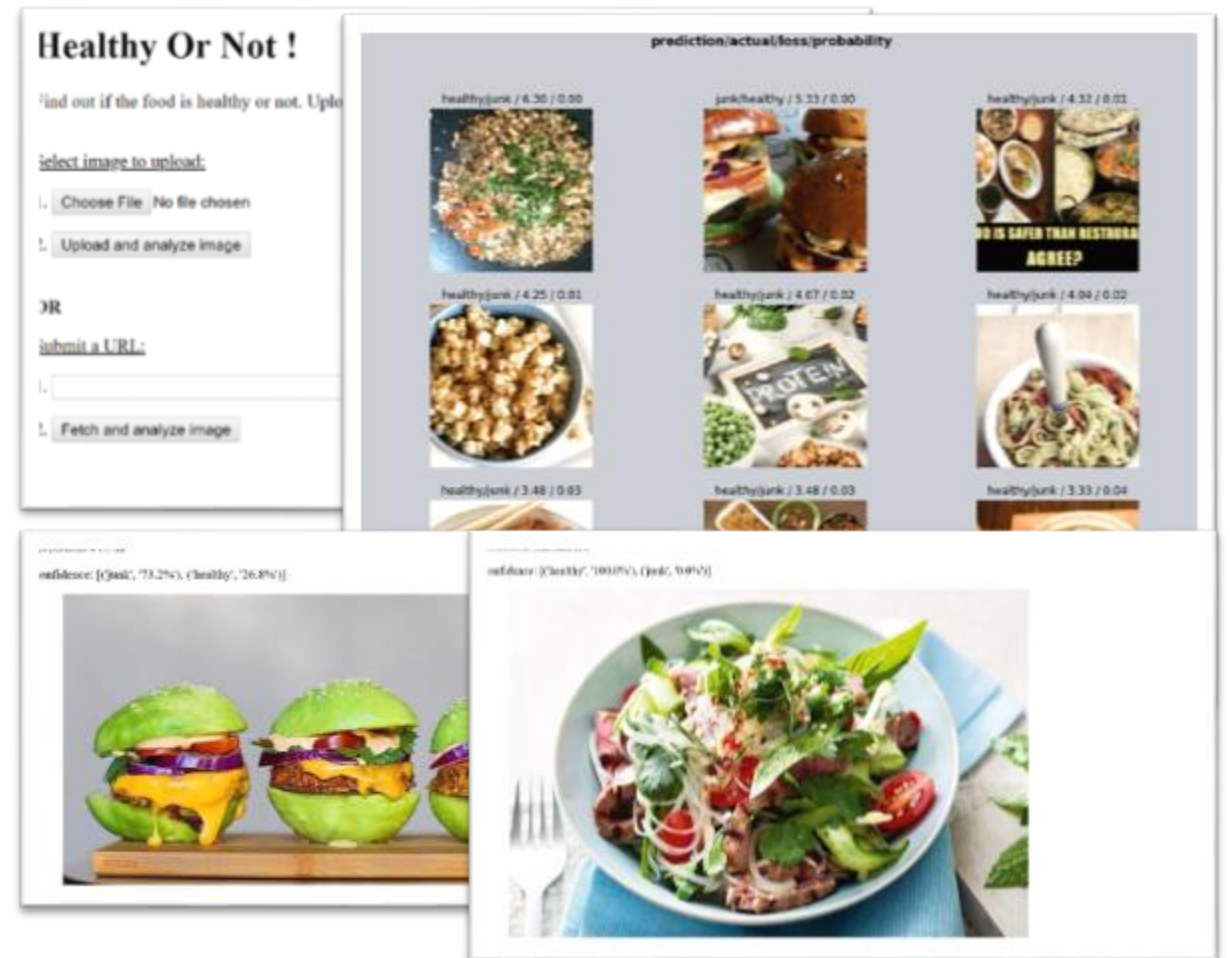
## A man with curly hair, wearing a dark green t-shirt, is speaking into a microphone. He is standing behind a podium. In the background, there is a screen displaying a grid of Devanagari characters and a sign that says "DATA SCIENCE".



## Guitar Classifier



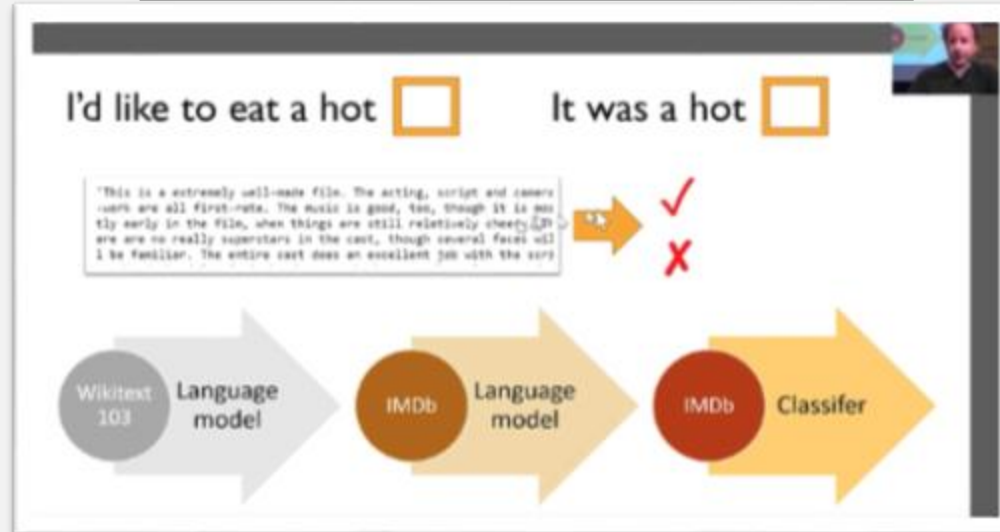
## Junk or Health Food Classifier



- [1]: <https://github.com/hiromis/notes/blob/master/Lesson3.md>  
[2]: <https://forums.fast.ai/t/share-your-work-here/27676/399>  
[2]: <https://forums.fast.ai/t/share-your-work-here/27676/340>

fast.ai not just classification

## Movie Review Sentiment Classification



## Legal Document Classification

### Legal Text Classifier with Universal Language Model Fine-tuning

LIM How Khang (LL.B. First Class Hon) (NUS), Advocate & Solicitor, Singapore

#### Overview

Recent work has shown that pre-training a neural language model on large public text datasets improves the accuracy of a neural text classifier while requiring fewer labelled training samples.

We use the Universal Language Model Fine-tuning method introduced by Howard and Ruder (2018)\* to train a legal text classifier to perform 19-way legal topic classification on a dataset of 3,588 legal judgments issued by the Singapore Court of Appeal and High Court.

\*Universal Language Model Fine-tuning for Text Classification  
https://arxiv.org/pdf/1801.06465v1.pdf  
https://github.com/harvardnlp/ud-pytorch

#### Dataset

The dataset was created by parsing the PDF Singapore Supreme Court judgments between 2000 and 2018 found on Singapore Law Watch (SLW), a free daily legal news website.

URL: <https://www.singaporelawwatch.sg/Subjects>

Size: 3,588 judgments (with only one fixed topic)

Topics: Civil Procedure, Criminal Law, Contract Law, Family Law, Companies, Tort, Arbitration, Legal Profession, Damages, Land, Insolvency Law, Building and Construction, Conflict of Laws, Administrative Law, Revenue Law, Trade Marks, Admiralty Law, Employment Law, Trusts.

#### Universal Language Model Fine-tuning by Howard and Ruder (2018)

Wiktext-103 Language Model

Specific Language Model

Specific Text Classifier

- Pre-trained language model on Wiktext-103
- Discriminative fine-tuning
- Started to regular learning rates
- Target task classifier fine-tuning with additional linear blocks and gradual unfreezing

#### Results

Accuracy: 82.56% accuracy on 19-way classification task  
Time taken to fine-tune LM (GTX 1080 Ti): approx. 3h 40m  
Time taken to train classifier (GTX 1080 Ti): approx. 1h

The model achieved strong results using the default settings in the fastai library. The incorrect test set predictions were reasonable errors due to an overlap in subject matter (e.g. Companies vs Insolvency Law, Revenue Law vs Land) and the generality of topics like Civil Procedure and Contract.

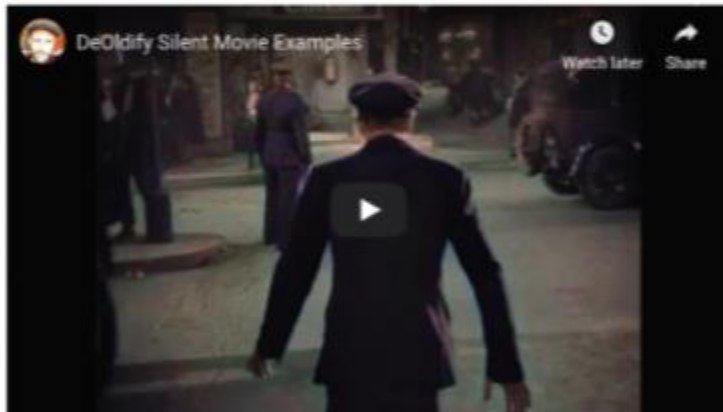
It would be useful to explore different levels of fine-tuning and whether multilabel classifier can achieve better results.

# GANs

## DeOldify

### Decrappification, DeOldification, and Super Resolution

In this article we will introduce the idea of "decrappification", a deep learning method implemented in [fastai](#) on [PyTorch](#) that can do some pretty amazing things, like... colorize classic black and white movies—even ones from back in the days of silent movies, like this:



The same approach can make your old family photos look like they were taken on a modern camera, and even improve the clarity of microscopy images taken with state of the art equipment at the [Salk Institute](#), resulting in 300% more accurate cellular analysis.

## Colorizer





# Collaborative Filtering

## Movie Recommendation System

### "How's that movie?"- Neural collaborative filtering with FastAI

Build a state-of-the-art movie recommendation system with just 10 lines of code



#### Looking at some predictions

While it's great to see the loss go down, let's look at some actual predictions of the model.

```
(users, items), ratings = next(iter(data.valid_dl))
preds = learn.model(users, items)
print('Real\\tPred\\tDifference')
for p in list(zip(ratings, preds))[:16]:
    print('{}\\t{:1f}\\t{:1f}'.format(p[0],p[1],p[1]-p[0]))
```

Real	Pred	Difference
5.0	4.2	-0.8
4.0	3.2	-0.8
3.0	3.4	0.4
3.0	2.4	-0.6
3.0	2.9	-0.1
2.0	2.4	0.4
4.0	4.2	0.2
4.0	4.6	0.6
4.0	3.6	-0.4
1.0	2.3	1.3
5.0	4.3	-0.7
2.0	2.9	0.9
4.0	4.0	0.0
4.0	3.7	-0.3

## Book Recommendation System

```
1 learn = collab_learner(data, n_factors=40, y_range=(1, 5), wd=1e-1)
```

Output:

```
Top idx:
array(['5000', '3315', '3313', '3312', '3311', '3309', '3308',
       '3307', '3306', '3304'], dtype='<U21')
```

Top names:

```
array(['Passion Unleashed (Demonica #3)', 'My Story', 'The Gargoyle',
       'Pretty Baby', ...,
       'Top Secret Twenty-One (Stephanie Plum, #21)', 'The Warrior
       Heir (The Heir Chronicles, #1)', 'Stone Soup',
       'The Sixth Man (Sean King & Michelle Maxwell, #5)'],
      dtype='<U144')
```

Most negative bias:

```
[(tensor(-0.1021), 'The Almost Moon', 2.49),
 (tensor(-0.0341), 'Skinny Bitch', 2.9),
 (tensor(-0.0325), 'Bergdorf Blondes', 3.0),
 (tensor(-0.0316), 'The Particular Sadness of Lemon Cake', 2.93),
 (tensor(-0.0148), 'The Weird Sisters', 3.08)]
...
```

[1]: <https://course.fast.ai/videos/?lesson=4>

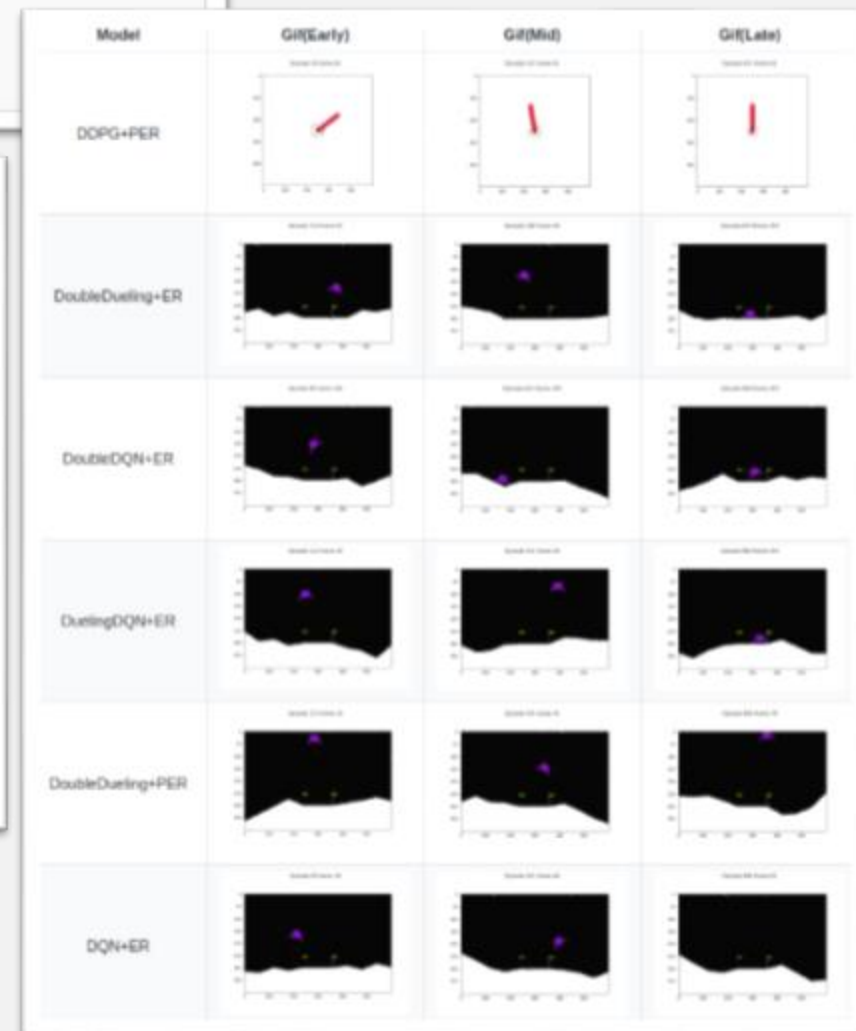
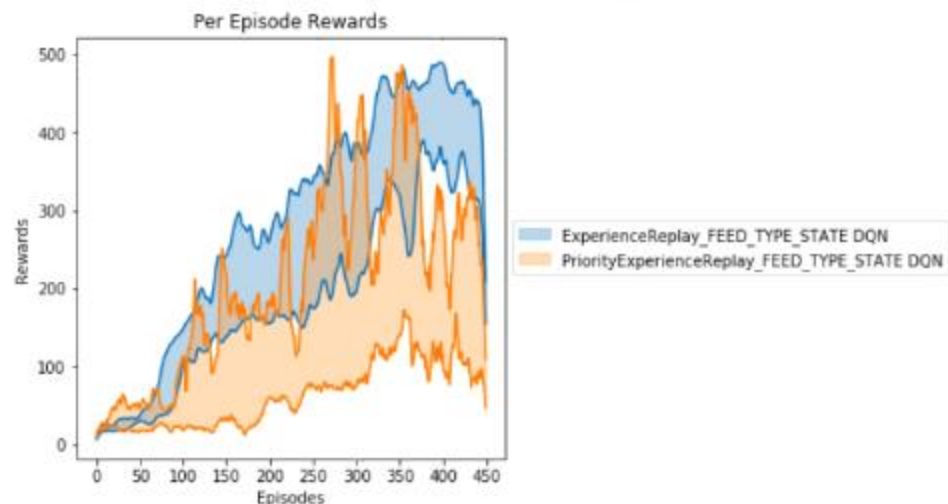
[2]: <https://towardsdatascience.com/collaborative-filtering-with-fastai-3dbdd4ef4f00>

[3]: <https://jovian.ml/aakashns/5bc23520933b4cc187cfe18e5dd7e2ed>

# Reinforcement Learning (experimental)

```
In [ ]: memory = ExperienceReplay(memory_size=1000000, reduce_ram=True)
        explore = GreedyEpsilon(epsilon_start=1, epsilon_end=0.1, decay=0.001)
        model = create_dqn_model(data=data, base_arch=DQNModule, lr=0.001, layers=[32,32], opt=optim.RM
        Sprop)
        learn = dqn_learner(data, model, memory=memory, exploration_method=explore,
        callback_fns=[RewardMetric, EpsilonMetric])
        learn.fit(3)
```

```
In [16]: per_group_interp = GroupAgentInterpretation.from_pickle('data/cartpole_dqn', 'dqn_PriorityExperienceReplay_FEED_TYPE_STATE')
        per_group_interp.add_interpretation(group_interp)
        per_group_interp.plot_reward_bounds(per_episode=True, smooth_groups=10)
```



Now to coding



fast.ai has a MOOC!



Do you want live notifications when people reply to your posts? [Enable Notifications...](#)

all categories +

Categories

Latest

New (21)

Unread (2)

Top

My Posts

Bookmarks

+ New Topic

Category

Topics

Latest

Part 1 (2019)

153 / month

2 unread

0 new

This topic is for anyone to chat about anything you want, as long as it is at least somewhat related to the course! (It's fine if you drift off topic a bit though.)

Advanced (Part 1 v2)

San Francisco

UTC-8 Pacific

UTC-5 Eastern

UTC-3 5th America

UTC-8 UK / EU West

UTC+1 Africa

UTC+2 EU East

UTC+3 Russia

UTC+5 India

UTC+8 China / SE Asia

UTC+10 Aust / NZ

Part 2 (2019)

10 / month

1 new

Welcome to part 2 (2019)! Please ensure that you've completed part 1 (2019) before the first lesson. If you haven't looked at the course for a while, I'd strongly suggest reviewing the lessons, since we'll be diving deep right from the first day of the course!

fastai users

107 / month

10 new

This is for help with installing and using the [fastai v1](#) library, for any level of user.

dev projects

fastai-v2

nbdev & blogging

Deep Learning

47 / month

1 new

Use this category to discuss anything to do with deep learning that's not related to a fastai course (each of those has its own category) - including stuff that's not related to fastai at all!

SwiftAI

1 / month

Forum for discussion of higher-level APIs for S4TF. This is a wiki post - feel free to edit it to add any high-level resources that everyone here should be aware of.

J

Welcome to forums.fast.ai

158

2d

Part 1 (2017)

P

Fastai v2 chat

723

12m

fastai-v2

Fastai v2 text

139

21m

fastai-v2

Nbdev and other git repos

10

36m

nbdev & blogging

Fastai v4 vourse

3

1h

fastai users

A walk with fastai2 - Study Group and Online Lectures Megathread

723

1h

fastai-v2

Fastai v2 vision

309

1h

fastai-v2

[fastpages] GitHub Pages Blog Using Nbdev

28

2h

nbdev & blogging

S

Deep Learning with Audio Thread

501

2h

Part 1 (2018)

UNIVERSITY OF SAN FRANCISCO

CHANGE THE WORLD FROM HERE

Admissions

News & Media

Calendar

Library

GIVE

Alumni

Parents & Families

myUSF

ABOUT USF • ACADEMICS • ADMISSION

STUDENT LIFE • SAN FRANCISCO ADVANTAGE

THE DATA INSTITUTE / CERTIFICATES / DEEP LEARNING CERTIFICATE PART I

OFFERED IN

Download

You'll learn the latest cutting-edge deep learning techniques in a practical, "top down" way, and train your own world-class models right from the first lesson. By the end of the course you'll have learned:

- How to create state of the art models in computer vision, natural language processing (NLP), recommendation systems, and tabular and time series data analysis
- How to use the brand new fastai v2 library along with PyTorch (the most popular software amongst top deep learning researchers)
- The foundations of deep learning: what is a neural network, how are they trained, and how do they make predictions
- How to turn your model into a real web application and how to debug your model if it goes wrong.

THE DATA INSTITUTE +

MEMBERSHIPS

INITIATIVES

PRACTICUMS

CERTIFICATES

- Deep Learning Part I
- Deep Learning Part II
- Exploratory Data Analysis Using R
- Fundamentals of Deep Learning using PyTorch
- Introduction to Data Ethics
- Introduction to Machine Learning
- Machine Learning for Business
- SQL for Data Analysis
- FAQs

[1]: <https://forums.fast.ai/>  
 [2]: <https://www.usfca.edu/data-institute/certificates/deep-learning-part-one>

(19)

\*From Stephen Welch's 2/7/2020 CAIR presentation

1. Form a **fastai study group**, contribute to the fastai library
2. Making kick-ass RL tools, push RL research forward– work with Josiah on fastrl!
3. Unsupervised & one-class learning in manufacturing on the MVTech dataset. [Stephen]  
I'm personally offering a \$250 bounty on this.
4. Open problem in manufacturing and autonomous driving – supervised model health monitoring  
- how do we know when a model is no longer performing well without labeling new data?
5. Create a CAIR Kaggle Team
6. Amazon DeepRacer team – I believe Dr. Shin is forming a team
7. Create high quality educational AI resources (videos, blogs, books)- happy to help as I have time.
8. Summer internship working with Stephen Welch & Josiah at Mariner
9. Summer internship at Atrium working on AI for medical applications

