

Basic recipe for a deep learning project

Everybody can bake a cookie

Filippo Biscarini
Senior Scientist
CNR, Milan (Italy)

Nelson Nazzicari
Research fellow
CREA, Lodi (Italy)



Basic recipe

1. Define a metric
2. Define a target
3. Prepare your data
4. Do good on train set
5. Do good on val/test sets



1) Have a single metric

- Crossentropy is usually a good choice for loss
 - But real application care more about errors/accuracy
- [P] Precision = (True Positives) / (All Positives)
 - E.g. precision is 95% and you say “sick” → there’s a 95% chance the patient is actually sick
- [R] Recall = (True Positives) / (True Positive + False Negatives)
 - E.g. recall is 95% → you have found 95% of the sick patients
- Tradeoff
- $F1 = 2 * (P * R) / (P + R)$
 - “Harmonic average”
 - F2 for recall, F0.5 for precision



1b) or a “single” metric

- Multiple metrics happen
- Have N metrics:
 - Pick one to be optimizing
 - N-1 will be satisficing

| Time | Error rate |
|------|------------|
| | |



2) Human level target

- “True” target: Bayes error rate
 - The lowest possible error rate, by definition
 - Unknowable :(
- Human expert(s) error rate
 - Good proxy on well known topics
 - Hard to get by (sometimes)
- State of the art classifiers
 - Maybe good proxy? Maybe not
 - Easy to get by



3) Prepare your data

- Normalize
- Preprocess (resize, feature extraction...)
- Train/val/test sets
 - They MUST come from the same distribution
- Unbalancedness



4) Do good on the train set

- More epochs
 - Different runs
- Optimizer
- Crossvalidate hyperparam choice
 - Learning rate, #/layers, #units, THEN everything else
- More data
- Bigger network
- Error analysis
- Restate the problem (e.g. decomposition)



5) Do good on the test set

- Overfitting?
- Errors in dataset?
- Regularization



Error analysis: False positives



Error analysis: False negatives

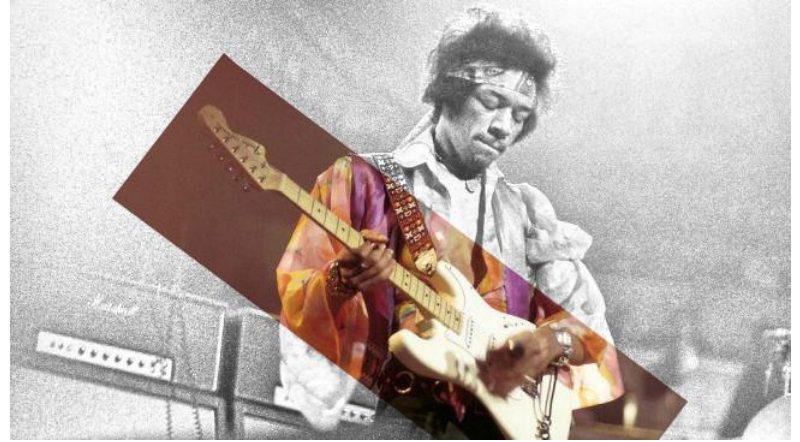


Problem decomposition

Classify the instrument



Find the instrument
Classify the instrument



Computation resources

- Google Colab
 - <https://colab.research.google.com/>
 - Remember to change runtime type
 - Factory reset is your friend
- Kaggle Notebooks (and datasets)
 - <https://www.kaggle.com/notebooks>
 - A phone number is required
 - Very fun (read: very stressing) competitions
- Binder
 - <https://mybinder.org/>
 - “Turn a Git repo into a collection of interactive notebooks”
 - A little waiting time
 - Learn about requirements.txt

