

Kaggle Bengali Classification Challenge

Kirill Vishnyakov

[!\[\]\(919a2cb85b99741a73c0c31a427236a8_img.jpg\) lightnezzofbeing](#)

[!\[\]\(666e09182d4cd268646ea700ea60dcdf_img.jpg\) LightnessOfBeing](#)

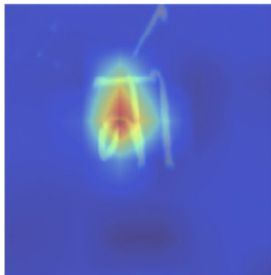
Target

- Given symbol with three components: grapheme_root (G), vowel_diacritic (V), consonant_diacritic (C).
- ~13,000 different grapheme variations.

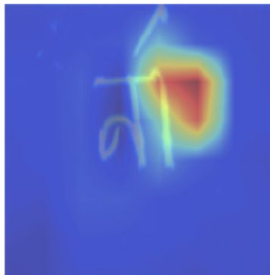
img_157847 r=81 v=1 c=2



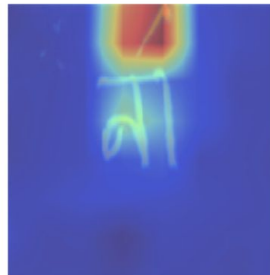
root = 81



vowel = 1



consonant = 2



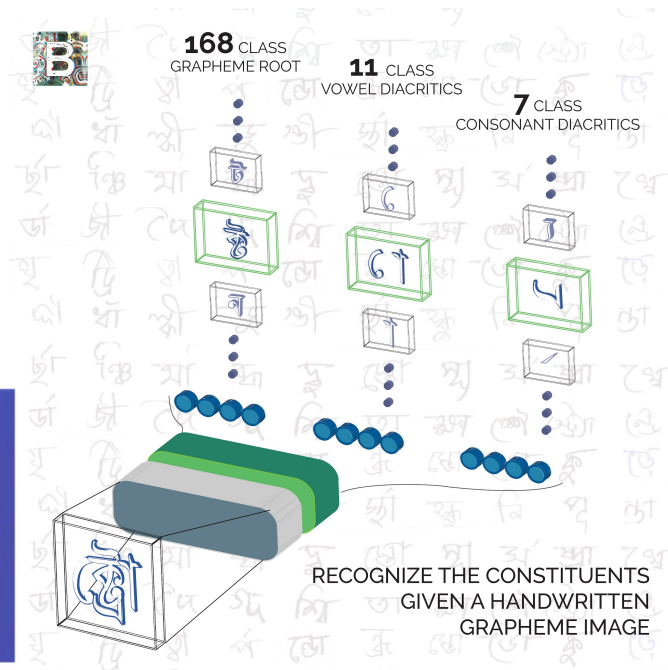
root=81



vowel=1



consonant=2



Data and metric

This dataset contains images of individual hand-written Bengali characters. Bengali characters (graphemes) are written by combining three components: a grapheme_root, vowel_diacritic, and consonant_diacritic. Your challenge is to classify the components of the grapheme in each image. There are roughly 10,000 possible graphemes, of which roughly 1,000 are represented in the training set. The test set includes some graphemes that do not exist in train but has no new grapheme components. It takes a lot of volunteers filling out sheets like this to generate a useful amount of real data; focusing the problem on the grapheme components rather than on recognizing whole graphemes should make it possible to assemble a Bengali OCR system without handwriting samples for all 10,000 graphemes.

- ~200000 images one channel images in total
- 168 G classes, 11 V classes, 7 C classes

Metric is weighted sum of recall for each component with weights:

- 2 - grapheme
- 1 - consonant diacritic
- 1 - vowel diacritic

Competition format

- Kernel only with no internet
- Public test: 12 images → no pseudo-labelling, but external data is allowed
- Inference within 2 hours on GPU

Baseline

- Resnet-50
- One model with 3 heads
- 30 epochs
- Basic augmentations: Scale, Rotate, HorizontalFlip, etc.
- CrossEntropyLoss
- Cropped images 128x128
- Random split
- Resulted in 0.95 public

Head configurations

1. Simply Linear after pooling
2. Mish -> Conv2D -> BatchNorm -> Pooling -> Linear
3. Pool -> Linear -> BatchNorm -> Dropout -> Mish

Result: 2 with GeM pooling > 2 > 3 > 1

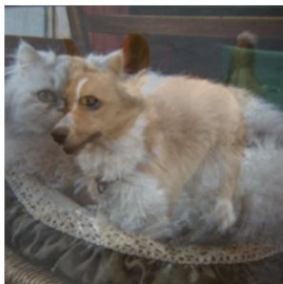
GeM pooling has slower convergence

Kaggle discussion: [Some experiments with CNN tails](#)

Augmentations

- Baseline augmentations don't work well. If we make them too aggressive the will destroy model, otherwise we will overfit.
- Mixup and Cutmix → require longer training, i.e. 100-200 epochs.
- Reasonable to resize images to 64x64.
- Need to decided on alpha parameter.

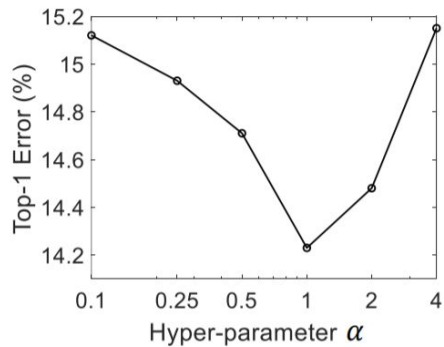
Mixup [48]



Cutout [3]



CutMix



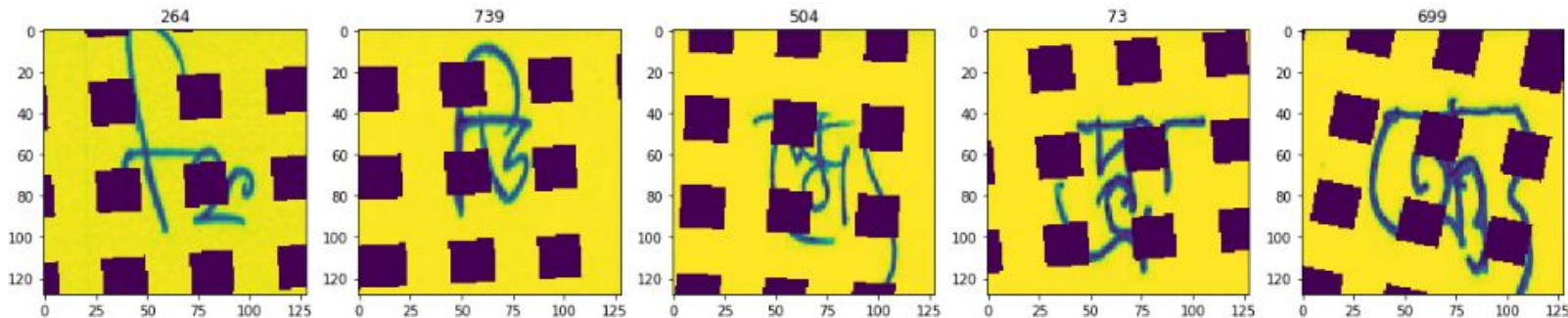
Cutmix alpha

Mixup and Cutmix

- If `random() > 0.5`: `mixup()` else `cutmix()`
- Reaching 0.96
- Hard times to fit pipeline in 9-hour kaggle kernel session with 128x128 images
- Still low score for G → assign more weight to G? → not really much of improvement
- Infeasible for higher resolution images, because high number of epochs and kaggle kernel limitations

Hard times understanding how to proceed

- Different architectures: se-resnext and efnets
- Any other augmentations except mixup and cutmix destroy model
- Single model for G performs worse than model with three heads
- FocalLoss slower than CE and performs worse, OHEM doesn't work
- Different schedulers and optimizers
- Papers suggest trying SGD → very slow compared to AdamW
- Architecture tweaks declared in papers don't work
- Some people report good performance of GridMask, CutOut and other SOTA augmentation methods → Bengali Augmentation challenge



That's basically me!



Andrew
Lukyanenko

254th place

It feels like I'm doing some basic thing very wrong

posted in [Bengali.AI Handwritten Grapheme Classification](#) a month ago



44

I started taking part in this competition some time ago and no matter what I do, I can't exceed 0.96 on the leaderboard and 0.965 on local validation.

I have tried the following things:

- various backbones from resnet50 to *seresnext50*;
- various ways to add pooling: changing pooling of the backbone or adding pooling after the backbone;
- starting from different learning rate: 0.1-0.0001;
- using no augmentations or using a lot of augmentations;
- creating images with iofass approach or simply pasting reshaped arrays on a square (`make_square` function), using different sizes from 128×128 up to 256×256;
- various optimizers and schedulers;
- using mixup callback from catalyst;
- training up to 150 epochs;
- training the whole net from the start or training only the head at the beginning and then the whole net;

And none of it helped.

Maybe I'm missing some basic thing?

You are still not there yet!



Andrew Lukyanenko

Topic Author

• (254th in this Competition)

My thanks to everyone who answered me :)

It seems that the following points were important:

- a heavier head of the net;
- bigger batch;
- better augmentation;
- longer training;
- little changes to parameters;
- teaming up;

→ Overfit

→ Depends on activations etc

→ kinda obvious

→ 40ep w 224x224 gives 9 hrs runtime

→ ???

One last try

- It might be because of misunderstanding of some Catalyst concept
- Should I rewrite it on pure Pytorch?
- Ok, the last run but with uncropped images → finally CV exceeds 0.97
- Substituting B0 with B3 gives CV 0.979/ LB 0.973


Out of ideas again

- Higher resolution (224 and 256) doesn't improve the score drastically
- Almost no difference between 80 and 140 epochs
- Gridmask and Cutout only worsen performance. In fact, any additional augmentations to mixup and cutmix don't work
- Balance sampler doesn't work

Final tweaking

- Switching to original images: 137x236 → ~0.3% of improvement
- Choosing good alpha for Cutmix ($\alpha=1$) and Mixup ($\alpha=4$) → ~0.3% of improvement
- Adjusting learning rate also helps a bit
- Weights to (7, 1, 2)
- Finetune 3 epochs without any augmentations → ~0.25% of improvement
- Single fold models 0.9855-0.9865 CV/ 0.9815-0.9825 LB
- Ensembling → ~0.985 LB

And here am I!



Qishen Ha
8th place

Have a Try on Validating With Unseen?

posted in [Bengali.AI Handwritten Grapheme Classification](#) 11 days ago



There is a risk of shake-up for this competition, which I have mentioned here:

<https://www.kaggle.com/c/bengali-ai-cv19/discussion/134035>

If we make the unseen validation splits separately, it is difficult to compare the ROBUSTNESS of models with each other.

- One week before the end
- People reporting awful results on validation with unseen
- Seems like a huge shake up is coming

Problem with Kaggle Forum

145



Best single model

DrHB 3 months ago

last comment by

Xie29 3d ago

407













- Everyone is too obsessed with architecture tweaks and hyperparameter tuning
- Nobody (almost) cares about validation and the problem itself
- High scoring guy reports he/she uses ReduceLROnPlateau → everyone jump testing
- Important information is being shadowed by topics like “Optimizer choice”

Final setup

- 137x236 original images
- Mixup + Cutmix, 0.5 probability for each
- CrossEntropyLoss with (7, 1, 2) weights
- EfficientNetB3 5-fold with stratified split
- $0.5 * (\text{AveragePooling} + \text{MaxPooling})$
- Optimizer AdamW
- Scheduler OneCycleWithWarmUp
- 100 epochs with augmentations + finetune 3 epochs without
- Training with O2 optimization level
- Ensembling of 5 folds

Results

- 41st place, Top 2%, silver medal
- 0.985 public / 0.940 private
- Huge shakeup
- Same setup, but with heavier model and longer training could reach ~20th place
- [My pipeline code](#)

| | | | | | | |
|----|-------|---------------------------|---|--------|----|-----|
| 35 | ▲194 | Dhananjay |  | 0.9405 | 46 | 6d |
| 36 | ▼24 | Kaz&CPMP |  | 0.9403 | 81 | 6d |
| 37 | ▲268 | Aryansh Omray |  | 0.9402 | 32 | 1mo |
| 38 | ▲102 | Dipam & Ram |  | 0.9402 | 73 | 6d |
| 39 | ▲1221 | Sumedh Ravi |  | 0.9402 | 3 | 2mo |
| 40 | ▲883 | NoGPU T^T |  | 0.9401 | 41 | 15d |
| 41 | ▲69 | Cyril |  | 0.9400 | 61 | 6d |
| 42 | ▲60 | Q |  | 0.9400 | 28 | 6d |
| 43 | ▲30 | AV |  | 0.9399 | 10 | 8d |
| 44 | ▲1104 | Syed Saad |  | 0.9395 | 22 | 13d |
| 45 | ▲151 | 은주니(ttagu99) |  | 0.9395 | 24 | 6d |
| 46 | ▲8 | [ods.ai] Here we go again |  | 0.9394 | 90 | 6d |

Focus on what important



Mishunyayev Nikita
254th place

We don't believe in magic, but how do you do it?

posted in [Bengali.AI Handwritten Grapheme Classification](#) 5 days ago



14

Hello, Kaggle community 🙌

Actively solving the competition last month, having spent a lot of time, at the moment we can't exceed 0.982 on the leaderboard.

- Read the description and data page
- Conduct better error-analysis
- Track your experiments (google sheets, etc)
- Think of a problem, not about optimizer choice!

Top solutions

Main Ideas

Basically we have 5 main ideas that works well in this competition as we can see now.

1. You don't give me unseen graphemes? I'll make it by myself. (1st, 2nd, 7th)
2. Figuring out unseen graphemes then predict them better. (3rd, 4th, 8th)
3. Relabel consonant diacritic (5th)
4. Using exists external data (9th)
5. Postprocessing (5th, 6th, 13th, 14th)

Q&A

Thanks for your attention!