Weekly Report: Exploration of BCE

Junhui Chen

Parts

- Brief Introduction to global BCE and batch BCE
- Wrong properties of BCE and batch BCE **without a optimal config**
- Further experiments with a optimal config
  - reconsidered the implementation of batch BCE: group training
  - Promote to CE (useless?)
  - Testing in extreme scenarios: **Lightweight Models** and **Complex Scenarios**

Brief Introduction to BCEs

global BCE:

$$L = \lambda \log(1 + \exp(-\cos(\theta_y))) + (1-\lambda) \sum_{i \neq y}^{K} \log(1 + \exp(\cos(\theta_i)))$$

The parameter λ is **fixed**.

But in fact, for **different positive examples**, The actual proportion of positive and negative samples **is different:**

$$\hat{\lambda} = \frac{S_{pos}}{\sum S_{neg}}$$

Brief Introduction to BCEs

batch BCE*: B is the set of speaking humans that appear in the current batch, b_ i is the i th speaker in B.*

$$L = \lambda \log(1 + \exp(-\cos(\theta_y))) + (1-\lambda)\sum_{b_i}^{B} \log(1 + \exp(\cos(\theta_i)))$$

Advantages:

- Reduced the disparity in the proportion of positive and negative classes, thereby effectively reducing the sensitivity of λ.
- Compressing the 1: K-1 binary classifier into a 1: B-1 binary classifier, which reduces training difficulty and makes model convergence more stable.

Wrong Properties of BCEs **without a optimal config**

- Performance of global BCE is **sensitive to**
  - λ.
  - data distribution & cls_num.
  - batch size.

- Properties above are verified as **fake** in a optimal config!

Further experiments with a optimal config

- Although all experiments with a optimal config shows that global BCE ≥ batch BCE.
- Based on intuition, we still have confidence in the generalization performance of batch BCE.
- So we have reconsidered the implementation of batch BCE.

Reconsidered the implementation of batch BCE

- In the first version of batch BCE, we calculated the loss within the batch and subsequently updated **all parameters of the linear layer.**

- To maintain consistency with our hypothesis of **ensemble learning**, we **group the parameters of the linear layer** (emb, cls_num) with columns, while each group corresponding to a class.

- Additionally, this method can obviously be extended to CE.

Testing in extreme scenarios

- Unfortunately, the same experiments shows that the new version of batch BCE is still worse than global BCE.

- We believe that the reason why global BCE performs better in multiple classes is due to the **strong backbone(resnet34 + ASP)**

- So we hope to explore the performance of two types of BCE in extreme situations where there are
  - lightweight model (resnet34 + ASP -> resnet10 + TSP),
  - more classes (vox1 -> vox2 -> cnc),
  - less data per class (rho=1, utt_per_spk=10),
  - and more complex data (vox -> cnc).

Testing in extreme scenarios

<u>base</u>: Resnet34 + ASP
Vox1 posterior SID & close set SV

Loss	vox1	
	WA	MA
global_BCE	95.26%	95.79%
shuffle_batch_BCE_256	94.12%	94.52%
shuffle_batch_BCE_512	93.71%	94.25%
batch_BCE_256	94.15%	94.78%
batch_BCE_512	94.93%	95.37%

loss	Vox1-O		
	EER	MinDCF(0.01)	MinDCF(0.001)
global_BCE	2.970%	0.33717	0.48708
shuffle_batch_BCE_256	3.723%	0.40008	0.58797
shuffle_batch_BCE_512	3.585%	0.37923	0.54420
batch_BCE_256	3.481%	0.37142	0.55600
batch_BCE_512	3.654%	0.37764	0.52835

Testing in extreme scenarios

<u>lightweight</u>: Resnet10 + TSP

Vox1 posterior SID & close set SV

loss	Vox1	
	WA	MA
Global_BCE	90.06%	91.03%
Batch_BCE_256	89.55%	90.46%
Batch_BCE_512	89.92%	90.77%
Batch_BCE_1024	89.31%	90.31%
Global_CE_256	90.35%	91.34%
Batch_CE_256	83.20%	84.93%
Batch_CE_512	81.19%	82.83%
Batch_CE_1024	73.96%	75.96%

loss	Vox1-O		
	EER	MinDCF(0.01)	MinDCF(0.001)
Global_BCE_256	5.462%	0.50189	0.71328
Batch_BCE_256	5.760%	0.52181	0.65742
Batch_BCE_512	6.148%	0.53587	0.68948
Batch_BCE_1024	5.510%	0.51831	0.65863
Global_CE_256	4.898%	0.50473	0.67698
Batch_CE_256	6.965%	0.55812	0.71663
Batch_CE_512	6.249%	0.55639	0.74130

Note: batch CE seems useless.

Testing in extreme scenarios

<u>Less data</u>: Resnet10 + TSP

Vox1 **rho=1 utt_per_spk=10** posterior SID & close set SV

loss	Vox1	
	WA	MA
Global_BCE_256	50.64%	51.83%
Batch_BCE_256	51.38%	52.62%
Batch_BCE_512	50.16%	51.48%
Batch_BCE_1024	50.67%	51.91%
Global_CE_256	46.96%	48.47%
Batch_CE_256	21.04%	21.56%
Batch_CE_512	18.76%	18.90%
Batch_CE_1024	20.62%	20.84%

loss	Vox1-O		
	EER	MinDCF(0.01)	MinDCF(0.001)
Global_BCE_256	12.339%	0.72380	0.80346
Batch_BCE_256	12.153%	0.73575	0.76891
Batch_BCE_512	12.701%	0.73697	0.79745
Batch_BCE_1024	12.754%	0.71959	0.81459
Global_CE_256	12.366%	0.74180	0.83518
Batch_CE_256	15.762%	0.82437	0.94227
Batch_CE_512	16.841%	0.86867	0.92745
Batch_CE_1024	14.881%	0.82198	0.89911

Note: batch CE seems useless.

Testing in extreme scenarios

<u>more classes</u>: Resnet10 + TSP:
Vox2 rho=1 utt_per_spk=10 posterior SID & open set SV

loss	Vox2	
	WA	MA
Global_BCE_256	73.37%	69.87%
Batch_BCE_256	64.33%	60.98%
Batch_BCE_512	67.50%	64.13%
Batch_BCE_1024	71.39%	68.07%

loss	Vox1-O		
	EER	MinDCF(0.01)	MinDCF(0.001)
Global_BCE_256	9.026%	0.68477	0.81707
Batch_BCE_256	10.924%	0.70687	0.79623
Batch_BCE_512	10.525%	0.70287	0.79553
Batch_BCE_1024	10.100%	0.69580	0.81495

Testing in extreme scenarios

More complex data & open set: Resnet34 + TSP

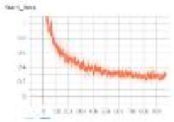
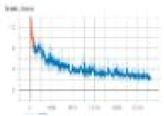
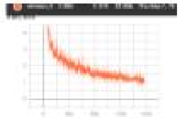
train: cn1 test: CNC-Eval-Core.lst

loss	SV CNC-Eval-Core.lst			Cos score SID ACC		
	EER	MinDCF(0.01)	MinDCF(0.001)	Top1	Top5	Top10
Global_BCE	20.698%	0.74417	0.81224	47.38%	61.16%	69.48%
Batch_BCE_256	19.640%	0.73951	0.81453	47.06%	61.15%	68.63%
Batch_BCE_512	20.873%	0.74741	0.82258	46.25%	60.11%	68.09%

Testing in extreme scenarios

<u>more data</u>: Resnet34 + TSP:

train: cn1+cn2   test: CNC-Eval-Core.lst

loss	loss pic	SV CNC-Eval-Core.lst			Cos score SID ACC		
		EER	MinDCF(0.01)	MinDCF(0.001)	Top1	Top5	Top10
Global_BCE_256		14.723%	0.62323	0.72317	58.33%	73.30%	80.08%
Batch_BCE_256		14.779%	0.65999	0.74448	54.32%	69.19%	76.55%
Batch_BCE_512		14.852%	0.65951	0.74191	54.85%	69.58%	77.01%

Conclusions

- Batch BCE can **slightly** surpass global BCE when the model is **lightweight**, the number of **classes is small**, and the **data is scarce**. At this point, batch BCE of different batch sizes will converge with global BCE.

- Except situation above, global BCE > batch BCE, and the batch size larger, the performence better.

Future work

- Perhaps we can further explore the effectiveness of batch BCE in Few-Shot Learning.