[Deep Batch Active Learning by Diverse, Uncertain Gradient Lower Bounds]

Diversity ⊕ Uncertainty Approach

- Measure of Uncertainty: the gradient magnitude with respect to parameters in the final output layer, which is computed using the most likely label according to the model.
- Capturing Diversity: Collect a batch of examples where these gradients span a diverse set of directions.

** K-means $^{++}$ algorithm
- ① captures magnitude of a candidate gradient
- ② captures its distance from previously included points in the batch.

[Notation & Setting]

K : # of classes

$[K] = \{1, 2, \cdots, K\}$

$\mathcal{X}$: the instance space

$\mathcal{Y}$: the label space

D_X : the unlabeled data distribution

$D_{Y|X}$: the conditional distribution over labels given examples

$\mathbb{E}_D$: Expectation under the data distribution D

h : a classifier that maps $\mathcal{X}$ to $\mathcal{Y}$ (architectures are fixed in any given context)

$$\ell_{01}(h(x), y) = I(h(x) \neq y) \quad \begin{cases} 1 & \text{if } h(x) \neq y \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbb{E}_D[\ell_{01}(h(x), y)] = Pr_{(x,y)\sim D}(h(x) \neq y)$$

S : a set of labeled examples

$\mathbb{E}_S$: sample averages over S

$f(x; \theta) \in \mathbb{R}^K$: probability vector of scores assigned to candidate labels, given examples x and parameters θ

$$h_\theta(x) = \underset{y \in [K]}{\operatorname{argmax}} f(x; \theta)_y$$

$$\ell_{CE}(p, y) = \sum_{i=1}^{K} I(y = i) \ln \frac{1}{p_i} = \ln\left(\frac{1}{p_y}\right)$$

$$\mathcal{L}oss = \mathbb{E}_S[\ell_{CE}(f(x; \theta), y)]$$

[Algorithm]

Two main computations

① Gradient embedding computation
② Sampling computation

} ⇒ Computes
- $\hat{y}(x)$: label preferred by the current model
- g_x : gradient of the loss on $(x, \hat{y}(x))$ with respect to the parameters of the last layer of the network.

<Gradient embedding computation>

✓ "Uncertain" if knowing the label induces a large gradient of the loss with respect to the model parameters and hence a large update to the model.

✓ Compute gradient as if the model's prediction on the example is true label.

✓ The gradient norm with respect to the last layer using the label

=

a lower bound on the gradient norm induced by any other label.

→ uncertainty

→ potential update direction upon receiving a label at an example.

<Sampling Step>

Want to secure both sample magnitude & batch diversity

✓ sample using K-MEANS++ seeding algorithm

↳ selects centroids by iteratively sampling points in proportion to their squared distances from the nearest centroid that has already been chosen.

→ select diverse batch of high-magnitude samples

[Example: multi class classification with softmax activations]

f: a neural network parameterized by θ w/ last non-linearity is softmax

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

$\theta = (W, V)$

W: $\theta_{out} = (W_1, \dots, W_k)^T \in \mathbb{R}^{K \times d}$ ← weights of the last layer

V: weights of all previous layers except the last layer

$$f(x;\theta) = \sigma(W \cdot Z(x;V))$$

Z: non-linear function

$$p_i = f(x;\theta)_i$$

$$\begin{aligned}
l_{CE}(f(x;\theta), y) = l_{CE}(p, y) &= \sum_{i=1}^{K} I(y=i) \log \frac{1}{p_i} \\
&= \sum_{i=1}^{K} I(y=i) \log \frac{1}{f(x;\theta)_i} \\
&= \sum_{i=1}^{K} I(y=i) \log \frac{1}{\sigma(W \cdot Z(x;V))_i} \\
&= -\sum_{i=1}^{K} I(y=i) \log \sigma(W \cdot Z(x;V))_i \\
&= -\sum_{i=1}^{K} I(y=i) \log \frac{e^{W \cdot Z(x;V)_i}}{\sum_{j=1}^{K} e^{W \cdot z_j}} \\
&= -\sum_{i=1}^{K} I(y=i) \left\{ \log e^{W \cdot Z(x;V)_i} - \log \sum_{j=1}^{K} e^{W \cdot z_j} \right\} \\
&= -\sum_{i=1}^{K} I(y=i) \left\{ W \cdot Z(x;V)_i - \log e^{WZ_1 \times WZ_2 \times \cdots WZ_K} \right\} \\
&= -\sum_{i=1}^{K} I(y=i) \left\{ WZ(x;V)_i - \prod_{j=1}^{K} WZ(x;V)_j \right\} \\
&= -W_y Z(x;V) + \log \sum_{j=1}^{K} e^{W_y \cdot Z(x;V)_j}
\end{aligned}$$

$g_x^y = \frac{\partial}{\partial w} \ell_{CE}(f(x;\theta), y)$ for a label y

$g_x = g_x^{\hat{y}}$

$\hat{y}$: $\underset{i \in [K]}{\text{argmax}}\ p_i$

gradient corresponding to label i:

$$(g_x)_i = \frac{\partial}{\partial w_i} \ell_{CE}(f(x;\theta), \hat{y}) = (p_i - I(\hat{y}=i))z(x;V)$$

following observations:

- ✓ g_x is a scaling of $z(x;V)$
- ✓ g_x estimates the example's influence on the current model
- ✓ High confidence tends to have gradient embeddings of small magnitude.