Inference Graphs for CNN Interpretation - Supplementary Material

Yael Konforti†, Alon Shpigler†, Boaz Lerner, and Aharon Bar-Hillel

Ben-Gurion University of the Negev, Beer Sheva, Israel
{yaelkonf, alonshp}@post.bgu.ac.il; {boaz, barhille}@bgu.ac.il

Update Equations for MLP Modeling (Section 3.1)

In a batch EM formulation, model parameters are updated based on sample statistics of interest. Each statistic is defined as the sample average $\frac{1}{N}\sum_{i=1}^{N} f(X_i)$ for a function $f(X)$ of interest. In the online setting, for each such function $f(X)$, an online sample estimator is kept, denoted here by $< f(X) >$. Given a set of examples $\{X_i\}_{i=1}^{B}$, $< f(X) >$ is updated by

$$< f(X) >_{t+1} = (1-\alpha) < f(X) >_t + \frac{\alpha}{B}\sum_{i=1}^{B} f(X_i) \tag{1}$$

where α is a smoothing factor. The tracked sufficient statistics are used in the update of the model parameters as follows:

– The transition probability update $t^l_{k,k'}$ between hidden state k' in layer $l-1$ and hidden state k in layer l is given by

$$t^l_{k,k'} = \frac{< P(h^l = k, h^{l-1} = k'|X,\Theta) >}{\sum_{k=1}^{K^l} < P(h^l = k, h^{l-1} = k'|X,\Theta) >}. \tag{2}$$

The average joint distribution of clusters from consecutive layers $< P(h^l = k, h^{l-1} = k'|X,\Theta) >$ is a tracked statistic, computed for each example using the forward-backward algorithm [2]. For the first layer, t^1_k is updated analogously using $t^1_k =< P(h^1 = k|X,\Theta) >$.

– The mean $\mu^l_{d,k}$ for an estimated Gaussian before rectification (dropping the l index for notation convenience), is given by

$$\mu_{d,k} = \frac{< P(h = k|x[d],\Theta)\cdot \hat{y}[d] >}{< P\left(h = k|x[d],\Theta\right) >} \tag{3}$$

with $\hat{y}$ defined by

$$\hat{y}[d] = \begin{cases} x[d], & x[d] > 0 \\ M_1(\mu_{d,k}, \sigma_{d,k}), & x[d] = 0 \end{cases}$$

† Equal contribution
Code for inference graphs algorithm released at github.com/yaelkon/GMM-CNN

The new mean is a weighted average of all examples' y activities, with each example contributing based on its probability to belong to the cluster. When $x[d] = 0$, the expected value of the activity prior to the ReLU operation is used. It is computed using as the first moment of a rectified Gaussian $M_1(\mu_{d,k}, \sigma_{d,k})$, which has a close form solution [1] for known mean and variance:

$$M_1(\mu, \sigma) = \int_{-\infty}^{0} x \cdot G(x|\mu, \sigma)dx = \mu - \sigma \frac{(G(-\frac{\mu}{\sigma}|0, 1))}{(C(-\frac{\mu}{\sigma}|0, 1))}, \tag{4}$$

where $G(-\frac{\mu}{\sigma}|0, 1)$ and $C(-\frac{\mu}{\sigma}|0, 1)$ are the density and cumulative value of the normal distribution at $-\frac{\mu}{\sigma}$. Since $\hat{y}[d]$ has two cases, two statistics are tracked for the computation of the nominator in Eq. 3: $< P\left(h = k|x[d], \Theta\right) \cdot x[d] 1_{x[d]>0} >$ and $< P\left(h = k|x[d], \Theta\right) \cdot 1_{x[d]=0} >$.

– The variance of an estimated Gaussian density before rectification $\sigma^l_{d,k}$, dropping the l index, is updated by

$$\sigma^2_{d,k} = \frac{< P(h = k|X, \Theta)(\hat{y}[d] - \mu_{d,k})^2 > + R_{d,k}}{< h = k|X, \Theta >} \tag{5}$$

This formula can be seen as a weighted sum-of-squares and a correction factor

$$\begin{aligned} R_{d,k} =& < P(h = k|x[d], \Theta) \cdot 1_{x[d]=0} > \\ & \cdot \left(M_2(\mu_{d,k}, \sigma_{d,k}) - M_1(\mu_{d,k}, \sigma_{d,k})^2\right) \end{aligned} \tag{6}$$

The term $M_2(\mu_{d,k}, \sigma_{d,k})$ is the second moment of a censored Gaussian distribution, which also has a closed form solution [1]:

$$M_2(\mu, \sigma) = \int_{-\infty}^{0} x^2 G(x|\mu, \sigma)dx = \mu^2 + \sigma^2 - \sigma\mu \frac{(G(-\frac{\mu}{\sigma}|0, 1))}{(C(-\frac{\mu}{\sigma}|0, 1))} \tag{7}$$

Additional Inference Visualizations (Section 4)

Additional MLP Inference Path Visualization (Section 4.1)

In Fig 1, we show an additional MLP inference path for a network trained on MNIST dataset. A path of an erroneous "nine" example in the MNIST network is partially presented with full blue clusters. Correct network decisions are made in layers fc-1 and fc-2 where the network associates the example with primary "nine" sub-clusters. The wrong decision of the network is made in layer fc-3 where it decided to send the example to a "four" cluster in layer fc-4, continuing with this pattern up until the classification layer.

Additional Inference Graphs Visualization (Section 4.3)

In this section, we present additional inference graphs for the "zebra" and "tractor" classes in the VGG-16 network. As an example of feature hierarchy diagnosis, an inference graph of a successfully classified zebra image is presented in

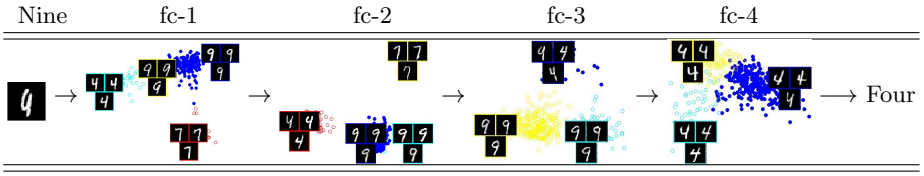

Fig. 1. MLP inference path of misclassified example on MNIST dataset. The input image traversing main decision points through layers fc-1-fc-4, where the main flawed decision is made in layer fc-3, where the open head of the image misleads the network into considering as a four digit.

Fig. 2. Focusing on the three bottom layers, the gradual development of discriminative stripe-based features can be seen. Visual words in Layer 3 (third from the bottom) are each characterized by a single orientation: vertical (left), leaning to the right (middle), or leaning to the left (right). These words, all with a similar spatial frequency and pattern, abstract over the spatial frequency by combining words from Layer 2 that mostly differ w.r.t their line spatial frequency and edge patterns, whereas words of Layer 2 compose edge feature patterns of Layer 1 (bottom). Note the green edges (and the corresponding probabilities quantifying them) connecting words of Layer 2 to words of Layer 3, and similarly between Layers 1 and 2, that demonstrate the influence of words on the creation and dominance of words in a higher layer.

Next, three inference graphs for the "tractor" class are presented: A class inference graph in Fig. 3, an inference graph for a wrongly-classified image in Fig. 4, and inference of a correctly-classified image in Fig. 5. The class inference graph (Fig. 3) demonstrates the evolvement of tractor "wheel" visual words, which end up at Layer 5 (left and right). These visual words were formed by an earlier visual word in Layer 4 (left), which represents a "curvy-edges" word, and this was partly constructed by a lower visual word in Layer 3 (left) representing a "curvy-diagonal-strips" word. The rest of the visual words in the graph present "grass" and "tree"-related words, corresponding to the natural background of a tractor.

Fig. 4 shows the inference graph for a tractor image wrongly-classified to the "snowplow" class. The reason for the wrong classification can be clearly seen in the dictionaries of visual words of multiple layers for which the tractor image was associated. Most of these words represent a combination of pale blue, gray, and white backgrounds at the upper parts of the image. While simple color features are used in low layers, these create complicated color features in Layer 5 (uppermost), focusing on combinations of vegetation with pale background (left), upper and side padding with pale background (middle), and horizontal low edges with pale background (right). The graph clearly shows the weakness

of the network with respect to the "snowplow" class, whose inference is primarily based on specific background colors.

In Fig. 5, a well classified "tractor" image is presented. The upper (fifth) layer shows that the visual words enable a successful inference because they focus on the tractor wheels creation, with the visual word of the wheel itself (left), and another two visual words representing the wheels' natural background of grass (middle) and road (right). The wheel related words in Layer 5 are composed of upper-half (right) and left-hand half (left) wheels in Layer 4. Below this layer, the most influential words chosen for presentation by the algorithm are of gray color and road texture, creating the road higher layer features.

Fig. 2. An image inference graph of a correctly classified zebra image.

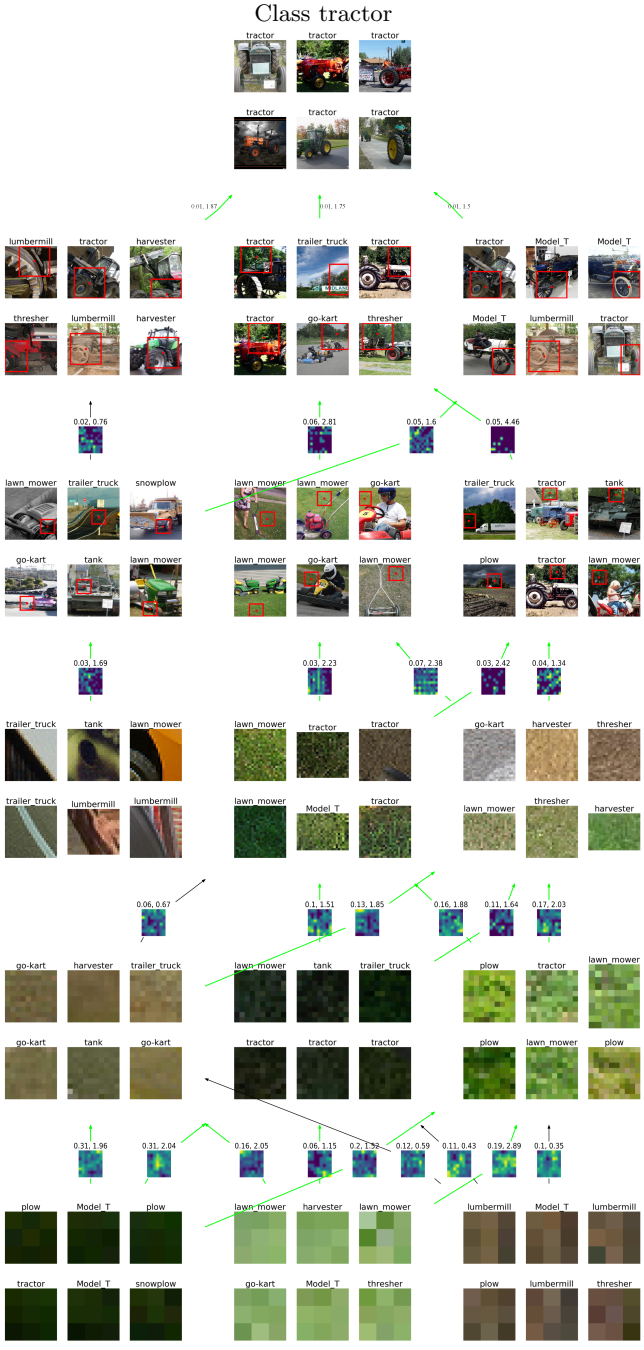

Fig. 3. Tractor inference graph.

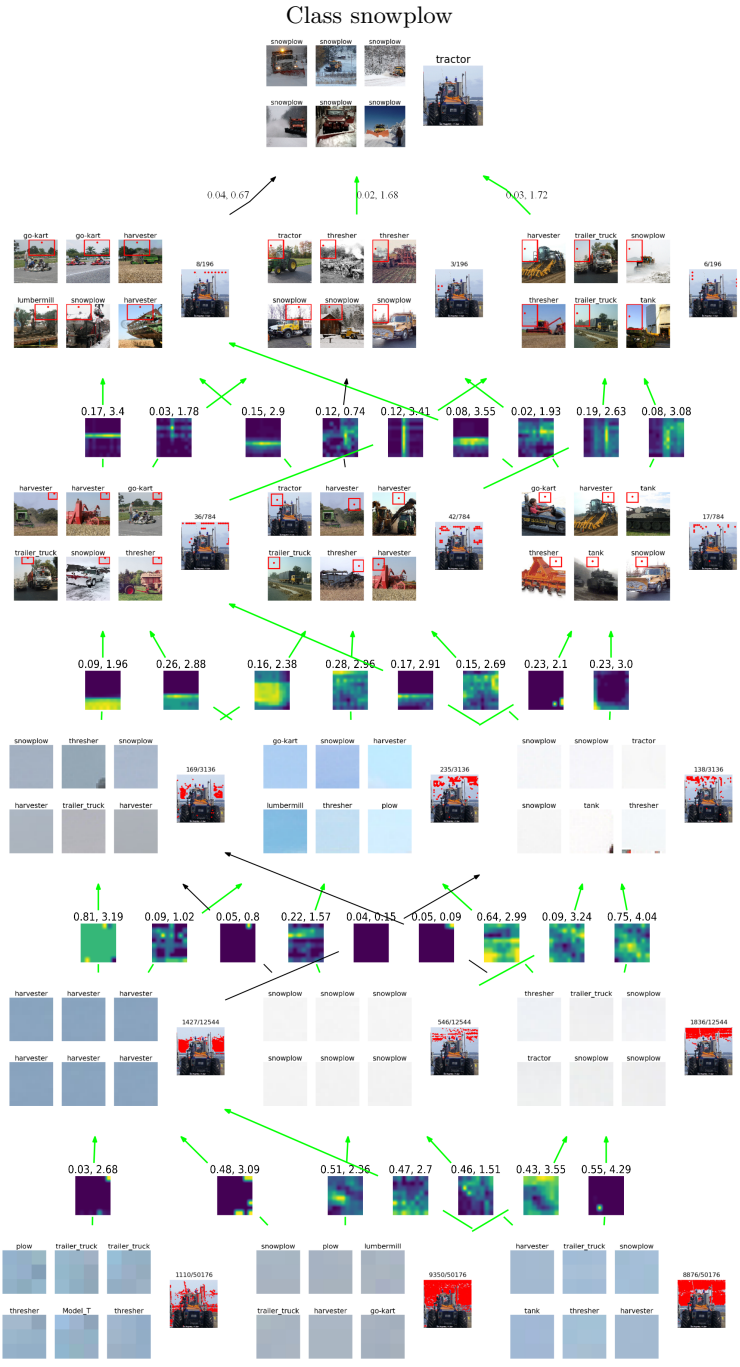

Fig. 4. An image inference graph of a wrongly classified tractor image to class "snowplow".

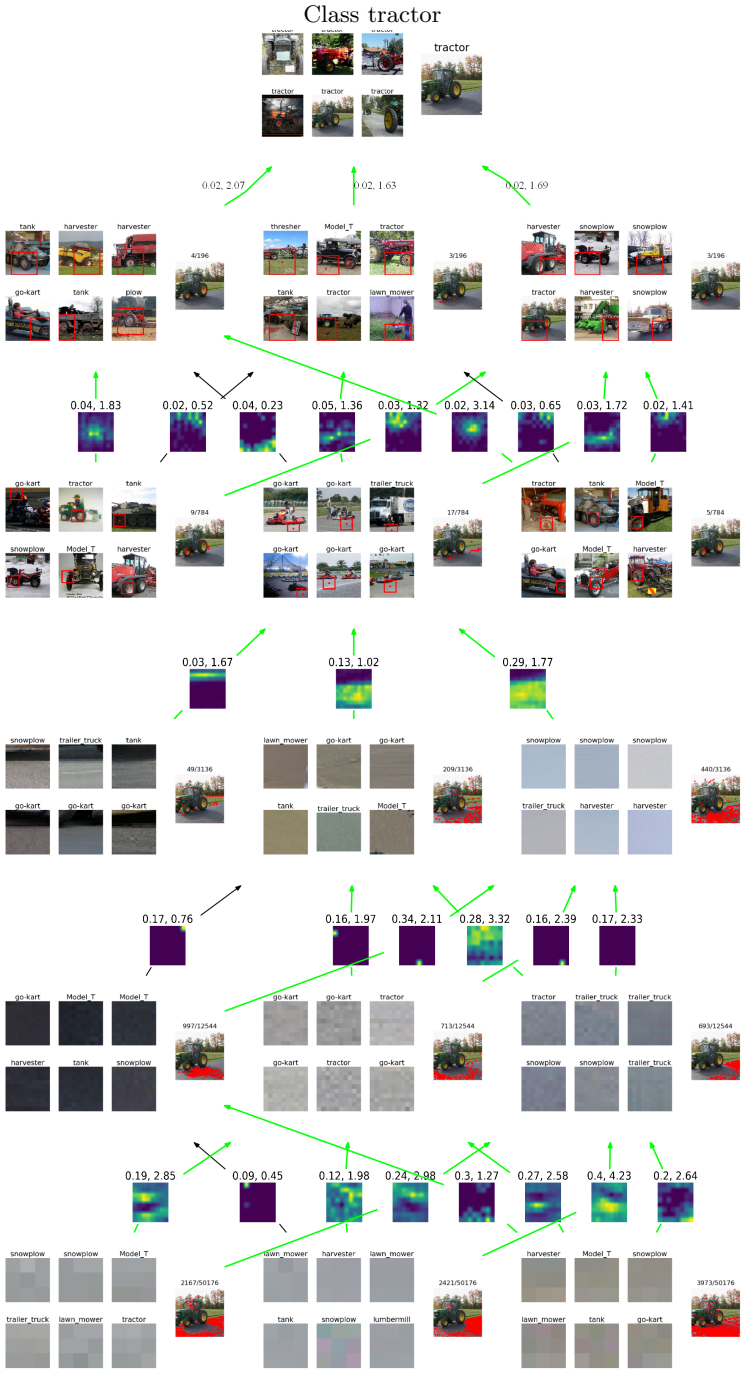

Fig. 5. An image inference graph of a correctly classified tractor image.

References

1. Lee, G., Scott, C.: EM algorithms for multivariate Gaussian mixture models with truncated and censored data. Computational Statistics & Data Analysis **56**(9) (2012)
2. Rabiner, L.R.: A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE **77**(2), 257–286 (1989)