Encoding Prior Knowledge using Label-Relation Graphs

Enayat Ullah
Department of Mathematics and Statistics
Indian Institute of Technology Kanpur
Kanpur, India
enayat@iitk.ac.in

Anshul Goyal
Department of Computer Science and Engineering
Indian Institute of Technology Kanpur
Kanpur, India
anshulgo@iitk.ac.in

Abheet Aggarwal
Department of Mathematics and Statistics
Indian Institute of Technology Kanpur
Kanpur, India
abheet@iitk.ac.in

1 Introduction

With the advent of deep learning, its plausible to unearth the underlying abstractions from raw data. However, a basic existing limitation is that they do not leverage the real world knowledge to boost their performance. In most multi-label classification tasks, the label space exhibits a rich structure, and the labels posses varying degree of constraints between themselves. Traditional classification models either consider the labels mutually exclusive(softmax) or pairwise-independent(logistic regressions). However, a knowledge graph over the labels is ideal to exploit this rich structure among labels.
Various line of works are dedicated to quantify these relations in different ways. Vedantam et al made use of human generated abstract scenes made from clipart for learning semantic visual information[1]. NEIL is another such powerful work which exploits the large-scale visual data to automatically extract commonsense relationships[2]. We draw inspiration from Deng et al, wherein they proposed a probabilistic classification model which combines powerful feature extraction with a graphical structure to encode prior beliefs. They model the label constraints using a label relation graph (Hierarchical and Exclusion - HEX), and their method is shown to be at par with the state-of-the-art in ILRC and zero-shot learning tasks[3].

2 Approach

2.1 HEX

We start by formally defining the label relation graph, which is called HEX(Hierarchical and Exclusion). As the name suggests this graph contains hierarchical relations (as directed edges) and exclusion relations (as undirected edges). Labels with no edge between them are considered overlapping. As you can see in Figure 1.

Theorem 1. *Hierarchical and Exclusion (HEX) Graph: A HEX graph $G = (V, E_h, Ee)$ is a graph consisting of a set of nodes $V = \{v_1, \cdots, v_n\}$, directed edges $E_h \subseteq V \times V$, and undirected/exclusion edges $E_e \subseteq V \times V$, such that the subgraph $G_h = (V, E_h)$ is a directed acyclic graph (DAG) and the subgraph $G_e = (V, E_e)$ has no self loop, where V is set of Vertices of graph G , E_h is the hierarchical edge and E_e are the exclusive edges .*

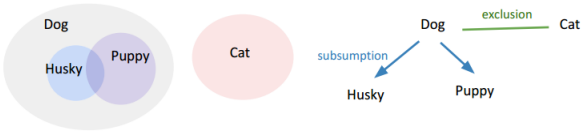


Figure 1: Heirarchy and Exclusion
Source: [3]

The restrictions imposed by hierarchical and exclusion relations between labels prune the state space and make some assignments illegal if they flout the constraints. Intuitively, it does not make sense if the child forms an exclusion edge with an ancestor. Formally, this can be put as:

Theorem 2. *A HEX graph $G = (V, E_h, E_e)$ is consistent if and only if for any label $v_i \in V, E_e \cap (\overline{\alpha}(v_i) \times \overline{\alpha}(v_i)) = \phi$ where V is set of Vertices of graph G , E_h is the hierarchical edge and E_e are the exclusive edges, v_i denotes the ancestors and the node itself.*

The joint distribution of an assignment of all labels $y \in \{0, 1\}$ is modeled as a Conditional Random Field (CRF). The scores f_i are raw classification scores which is obtained from the last layer of a deep neural network.

$$\widetilde{P}(y|x) = \Pi_i e^{f_i(x;w)[y_i=1]} \Pi_{(v_i,v_j)\in E_h}[(y_i, y_j) \neq (0, 1)] \Pi_{(v_i,v_j)\in E_e}[(y_i, y_j) \neq (1, 1)]$$

where $\widetilde{P}$ is unnormalized probability, $Pr(y|x) = \widetilde{P}(y|x)/Z(x)$, where $Z(x)$ is the partition function, $f_i(x; w)$ are raw classification scores and y_i and y_j are the current state of the node. By this probability we have made sure that all the illegal states are not considered for any analysis.

To compute the probability of a label, we need to marginalize over all labels.

Theorem 3. *The complexity of the exact inference for graph $G = (V, E_h, E_e)$ is $O(min||V|2^w, |V|^2 2^{\Omega_G})$*

Ω_G is the maximum overlap of a consistent graph $G = (V, E_H, E_e)$, i.e. $\Omega_G = max_{v\in V}| o_{\overline{G}}(v) |$, where $o_{\overline{G}}(v) = u \in V\ (u, v) \notin \overline{E_h} \wedge (v, u) \notin \overline{E_h} \wedge (u, v) \notin \overline{E_e}$ and $\overline{G} = (V, \overline{E_h}, \overline{E_e})$ and w is the width of the junction tree.

We need to bound on the complexity of the inference algorithm using the given theorem.Now the graph can either have a large treewidth but small overlap or vice-versa.

We now use the maximum overlap of a graph to bound the size of its state space :

Theorem 4. *For a consistent graph $G = (V, E_h, E_e)$, $|S_G| \leq (|V| - \Omega_G + 1)2^{\Omega_G}$*

As a matter of fact,if a HEX graph consists of a tree hierarchy and exclusion edges between all siblings, then it is easy to verify that its maximum overlap is zero and its state space size is exactly $|V|$ + 1, a tight bound in this case .

Now we would like to discuss one other property known as Joint hierarchical modeling. Our model allows flexible joint modeling of hierarchical categories,which helps us in transferring of potential knowledge. You can verify that for all graphs the marginal probability of a label depends the sum of its ancestors scores, i.e. $Pr(y_i = 1|x)$ has the term $exp(f_i + \Sigma_{v_j\in\alpha(v_i)} f_j)$, because all the ancestors have to be 1 if the value taken by label is 1.

Conversely, descendants also play role in the probability of a node. We need to marginalize over all possible states of descendants. Let's see this by an example. We have a tree hierarchy in which siblings are mutually exclusive. Then we can easily show that the unnormalized probability $\overline{P}$ of a node has simple recursive form involving its own score, unnormalized probabilities of the direct children and the anscestors scores.

In figure 2 , you can see the model difference in Softmax and HEX model.

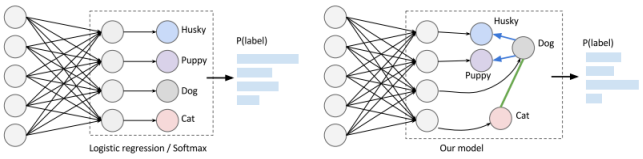


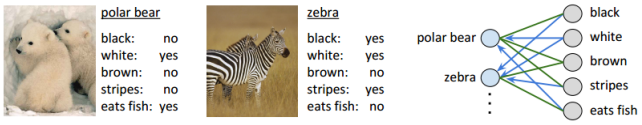
Figure 2: Softmax v/s HEX
Source: [3]

2.2 Learning

Now for learning , we are going to train our model to learn parameters such that the loss function is minimized .We formulate the loss function of the model as the negative log likelihood using the marginal probability of the ground truth labels. The weights are estimated using Stochastic Gradient Descent(SGD).

$$L(D, w) = -\Sigma_l log Pr(y^{(l)}_{g(l)}|x^{(l)}; w) = -\Sigma_l log \Sigma_{y:y_{g^{(l)}}=y^{(l)}_{g(l)}} Pr(y|x^{(l)}; w)$$

where $g(l)$ is the indices of the observed labels.



Source: [3]

2.3 Inference

We need to calculate the marginal probabilities in order to do inference. This computation is exponential in the number of labels if done by brute force. However in realistic settings, the graph is densely connected(abundance of mutually exclusive relations) and it prunes the state space considerably. The marginal here is computed by a modified junction tree algorithm which operates on (maximally) densified and sparsified graphs for tractable inference.

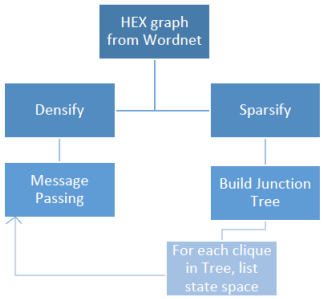


Figure 3: HEX pipeline

Moralization

Moralization is done to transform directed graph into undirected one using the following algorithm:

- For each node Xi connect all parents of Xi to a clique

- Drop orientation of edges The new undirected model represents the same distribution.

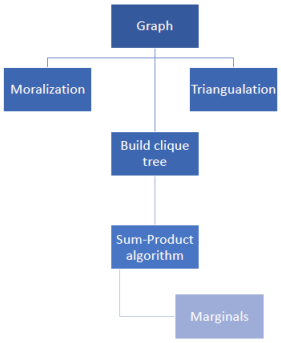
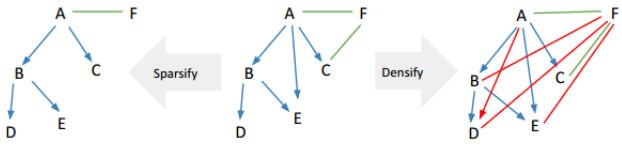


Figure 4: Junction Tree pipeline

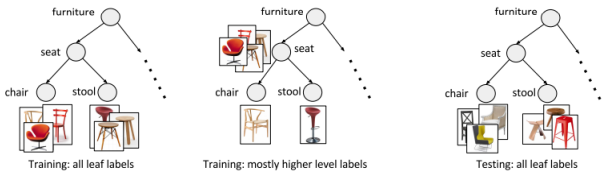
3 Performance

The HEX graph is constructed using the wordnet semantic hierarchy. "Exclusion wherever possible" principle is adopted to add the exclusion edges. This ensures that the graph is densely connected. Marginalizing over the labels by brute force incurs a time complexity exponential in the order of nodes. We resolve this using the modified junction tree algorithm presented in figure which runs in the order of size of the tree width. We tested our model on ILSVRC2012 dataset on 50 percent relabeling (leaf examples to their immediate parents) .



Source: [3]

Dataset	Softmax	HEX
ILSVRC2012	50.5	54.3
Zoo	82.43	85.56
Forest	72.39	71.11



Source: [3]

The following is an example ,figure 5, where our classifcation model predicted correct label :

Correct Label	Predicted
CAR	CAR

Figure 5

4 Conclusion

- Based on the theoretical and experimental evidence, it is apparent that the classifier leverages on the additional constraints when the label space exhibits a rich structure.
- In cases where the labels are not semantically co-related, the performance is at par with the traditional softmax, owing to the HEX's property to reduce to a softmax when the graph consists of all mutually exclusive relations.
- The implementation is task-independent. Given the feature vectors and the corresponding labels, the HEX graph is constructed without external intervention. It thus is an end-to-end module for any classification task.
- The modified junction tree algorithm leverages on the redundancy of HEX graph to perform message passing on a small treewidth graph(sparsified) with a very small state space(densified).

5 Future Work

- There are a couple of extensions to what we have achieved till now. We started with the objective of creating an end-to-end module which given the feature vectors and labels replaces the existing softmax with the automatically constructed HEX.
- We are currently operating on simple linear models, however the results are bound to improve if we put it over excellent feature extractors like Deep Neural nets.
- Integrating the model for single objects into a larger framework which can incorporate spatial relations between objects.
- Visual Question and Answering can be considered for such datasets pertaining objects classfication questions and then we can use MSCOCO dataset [4] with a k-most frequent labels constraint.

References

[1] Ramakrishna Vedantam, Xiao Lin, Tanmay Batra, C Lawrence Zitnick, and Devi Parikh. Learning common sense through visual abstraction. In *Proceedings of the IEEE International Conference on Computer Vision*.

[2] Xinlei Chen, Ashish Shrivastava, and Arpan Gupta. Neil: Extracting visual knowledge from web data. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 1409–1416. IEEE, 2013.

[3] Jia Deng, Nan Ding, Yangqing Jia, Andrea Frome, Kevin Murphy, Samy Bengio, Yuan Li, Hartmut Neven, and Hartwig Adam. Large-scale object classification using label relation graphs. In *Computer Vision–ECCV 2014*.

[4] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *International Conference on Computer Vision (ICCV)*, 2015.