Learning Graph Structure for Multi-label Image Classification via Clique Generation

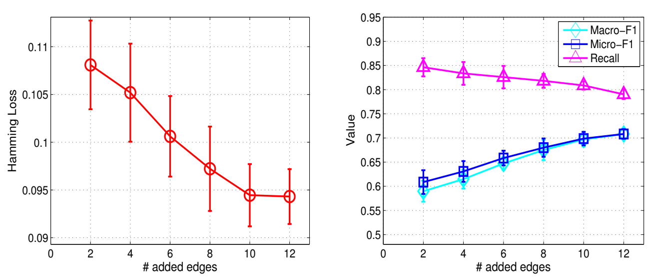
Mingkui Tan[1], Qinfeng Shi[1], Anton van den Hengel[1], Chunhua Shen[1], Junbin Gao[2], Fuyuan Hu[1], Zhen Zhang[1]

[1]School of Computer Science, The University of Adelaide, Australia. [2]School of Computing and Mathematics, Charles Sturt University, Australia.

Multi-label image classification is to predict a binary label vector to indicate the presence or absence of certain object categories in an image [1]. Exploiting label dependency can significantly boost the classification performance. For example, if an instance of the *ship* category is present in an image, it is very likely that the *water* category is also presented. To capture the label dependencies, it is a common practice to use Probabilistic Graphical Models (PGMs) [4], a standard workhorse for modelling dependencies among random variables. However, the structure of graphical models in existing methods is either determined heuristically or learned from very limited information, and it is still a challenging task to correctly and efficiently estimate a proper graph. People often use heuristics, such as manually specified graph structures based on domain knowledge, or simple rules like minimum spanning tree based on certain distance scores and the ChowLiu Tree [2, 3] which uses mutual information between labels and ignores features or visual contents of images completely.

In this paper, the graph is unknown initially, and we propose to learn the graph structure and model parameters jointly from the data by considering input features and labels. In this way, the learned graph structure and parameters will fit the data better.

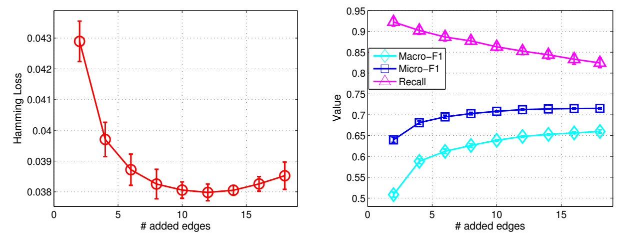
Learning a graph can be seen as selecting relevant cliques from all possible cliques. Although there might be many potential cliques, only a few of them are relevant to the output in the sense that, a label (e.g. an object) in practice is often related to only a small number of other objects. To find the relevant cliques, we introduce a 0-1 indicator vector into the SSVMs framework to index the potential cliques, and we impose an ℓ_0-norm constraint on the vector to induce sparse solutions. The resultant problem is non-convex, but we transform it into a convex programming problem through a tight convex relaxation. The relaxed problem has exponentially many constraints. To address it, we propose a cutting plane algorithm, which iteratively activates a group of cliques until the structural loss cannot decrease significantly. Our approach, which is referred to as the clique generating machine (CGM), exhibits both strong theoretical properties and a significant performance improvement over state-of-the-art methods on both synthetic and real-world data sets.



(a) Hamming loss (b) Other measures

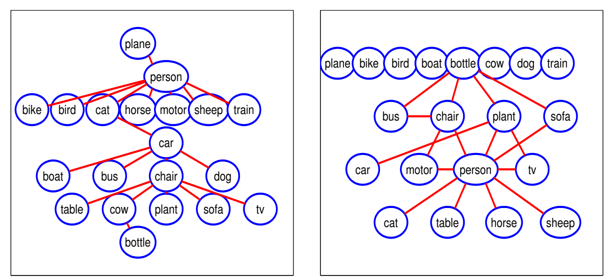
Figure 1: Performance variations v.s. # edges on Scene

In Figures 1 and 2, we show the performance changes of CGM versus the number of edge cliques on Scene and PASCAL07. From the figures, we draw two conclusions. Firstly, on the two data sets, adding relevant label dependencies improves the performance in terms of Hamming loss and F1 measures. Secondly, adding too many label dependencies does not necessarily improve the performance significantly. Actually, on Scene, the performance of CGM does not show significant improvement more than 10 edges. On PASCAL07, the performance of CGM in terms of Hamming loss degrades when there are more than 12 edges. In other words, adding too many edges (cliques), especially irrelevant edges (cliques), may degrade the performance.

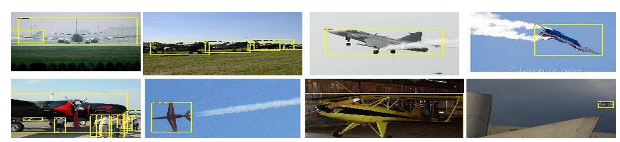


(a) Hamming loss (b) Other measures

Figure 2: Performance variations v.s. # edges on PASCAL07



(a) Graph by ChowLiu Tree [3] (b) Graph by proposed method



(c) Sample images of "plane" in PASCAL2007 database

Figure 3: Comparison of graphs obtained by ChowLiu Tree and proposed method on PASCAL2007. Unlike the graph in Figure (3(a)), our learned graph has some categories not connected with other nodes, e.g., "plane". From Figure (3(c)), in general, the presence of a "plane" in an image is independent of other objects except "sky".

An important concern for learning a graph is that, while it is essential to find the relevant dependencies (cliques), it is also very important to identify the *independent labels* and *irrelevant or false* dependencies. In Figure 3, we show the graph constructed by ChowLiu Tree [3] (Figure 3(a)) and the graph learned by our method (Figure 3(b)). Unlike the ChowLiu Tree graph wherein all the labels are connected, some labels in our graph are isolated (such as "plane"), which makes learning and inference faster and simpler.

[1] Matthew R Boutell, Jiebo Luo, Xipeng Shen, and Christopher M Brown. Learning multi-label scene classification. *Pattern Recognition*, 37(9):1757–1771, 2004.

[2] Joseph K Bradley and Carlos Guestrin. Learning tree conditional random fields. In *ICML*, 2010.

[3] C Chow and C Liu. Approximating discrete probability distributions with dependence trees. *IEEE Trans. Inform. Theory*, 14(3):462–467, 1968.

[4] D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

This is an extended abstract. The full paper is available at the Computer Vision Foundation webpage.