Hierarchical Subquery Evaluation for Active Learning on a Graph Supplementary Material

Oisin Mac Aodha Neill D.F. Campbell Jan Kautz Gabriel J. Brostow
University College London
http://visual.cs.ucl.ac.uk/pubs/graphActiveLearning

1. Learning Curves

In Figure 1 we display the learning curves for the 13 datasets presented in Table 1 in the main paper. A subset of these results were presented in Figure 2 in the paper. Datasets are organized from smallest to largest.

2. Varying the Number of Subqueries

In Figure 2 we display learning curves with varying numbers of subqueries for our 13 test sets. Our algorithm almost always performs better than RALF, even with surprisingly few subqueries. The notable exception to this is the cropped Pascal 08 dataset. In this instance, the breadth first search is superior and the situation is exacerbated by increasing the number of subqueries. We believe this is due to the high intra-class variance within each of the Pascal classes. This makes a semi-supervised approach less valid and thus increasing the performance of a naive exploration approach since there is no potential for boundary refinement. This is supported by the improved performance of bFirst with the LGC method in Table 1 since the marginal distributions are unimportant and LGC can outperform Zhu in such situations. We have observed that by reducing the number of nearest neighbours in the graph to 5 our area under the learning curve improves to 0.217 for our algorithm and 0.208 for breadth first.

3. Comparison to LGC

In Table 1 we evaluate EER using the closed form LGC method of Zhou *et al*. [38] for label propagation. The LGC is an alternative to GRF and uses the normalized graph Laplacian, $S = D^{-1/2} W D^{-1/2}$, so

$$F = (1 - \alpha)(I - \alpha S^{-1})\, Y, \tag{1}$$

where I is the identity matrix and α is a regularization parameter. Unlike GRF, LGC does not explicitly clamp the labels provided by the oracle and as a result they may change during label propagation. We also use the class re-weighting method of [34], denoted as 'wR'. The last column displays the result of our hierarchical algorithm from the main paper which uses the GRF formulation of Zhu *et al*. [42]. We note that bFirst does best in LGC which indicates that the marginals are not good for boundary refinement.

4. Implementation Details

We can efficiently compute the inverse in equation (1) using the sparse Cholesky decomposition and we apply a small 'jitter' factor to the leading diagonal to improve stability. To cope with graphs containing multiple connected-components, we use $(D_{uu} - W_{uu} + \lambda\, I)$, where λ is small compared to the row sums in D_{uu}. Components currently containing no labels in Y_l are given a uniform output distribution across all classes - the high entropy will encourage the EER criterion to ask the user to provide labels.

5. Video

In the included video we illustrate the reason for our algorithms success by displaying the boundary refinement on a subsampled version of the oil dataset. It is worth noting that the baselines RALF and bFirst revert to density sampling and do not refine the boundaries between the classes.

Dataset	randSubLGCwR	randSubLGC	bFirstLGCwR	bFirstLGC	oursLGCwR	oursLGC	**HSE (ours)**
Glass	0.6071	0.4549	0.6430	<u>0.7305</u>	0.7229	0.4600	**0.8040**
Ecoli	0.6873	0.5661	<u>0.7749</u>	0.6622	0.6916	0.5058	**0.8330**
Segment	0.8158	0.7646	0.8107	<u>0.8292</u>	0.7942	0.7710	**0.8960**
FlickrMat	0.1287	0.1077	**0.2649**	0.2111	0.1218	0.1079	<u>0.2590</u>
Coil20	0.6953	0.6662	<u>0.7582</u>	0.7548	0.7144	0.6631	**0.7600**
LFW10	0.2045	0.3651	0.1780	<u>0.3863</u>	0.0775	0.3646	**0.4220**
UIUCSport	0.3532	0.4039	<u>0.6534</u>	0.6439	0.3642	0.3816	**0.6710**
Gait	0.4526	0.3603	<u>0.6497</u>	0.6492	0.4017	0.5002	**0.6960**
Oil	0.9108	0.9083	0.9434	<u>0.9448</u>	0.7697	0.9277	**0.9860**
Caltech4	0.9859	0.9807	**0.9957**	<u>0.9954</u>	0.9824	0.9860	0.9900
Eth80	0.5761	0.5912	<u>0.6353</u>	0.6510	0.5212	0.5300	**0.6750**
CpPascal08	0.0843	0.1487	0.1706	**0.1863**	0.0948	0.1551	<u>0.1840</u>
15Scenes	0.2468	0.3480	<u>0.5472</u>	0.5204	0.3162	0.4897	**0.5730**
Mean	0.519	0.513	0.617	<u>0.628</u>	0.506	0.526	**0.673**

Table 1. Evaluation of algorithm using the closed form LGC method of Zhou *et al*. [38] for label propagation compared to the GRF method of Zhu *et al*. [42] (last column) used in the main paper. The columns with subscript 'wR' use the class re-weighting term of [34]. For each dataset, the winner (largest area under learning curve) is marked in bold with the second place underlined.

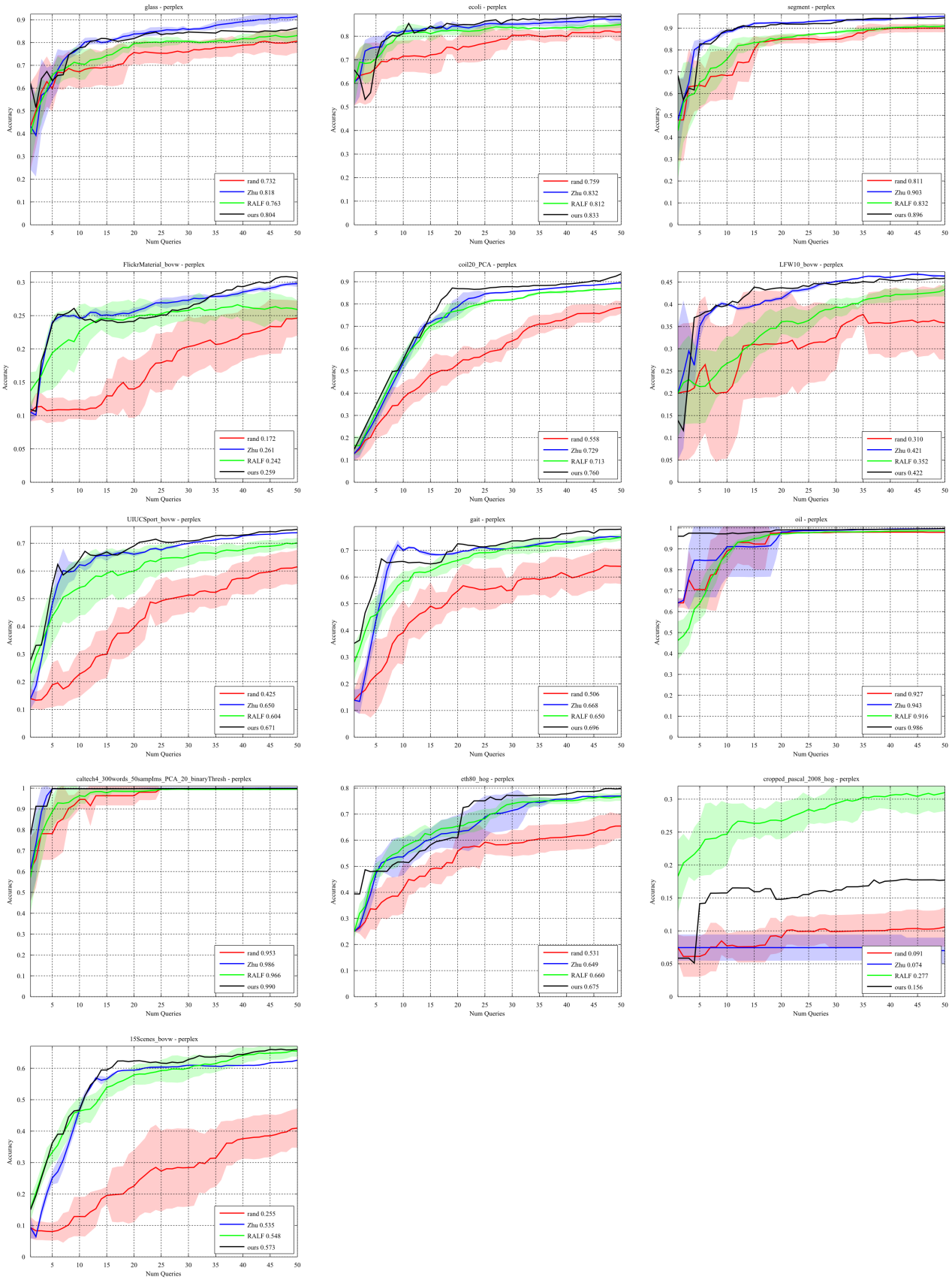

Figure 1. Learning curves illustrating the performance of our approach versus three other baselines from Table 1 in the main paper. The shaded regions around each learning curve represents one standard deviation. The area under the learning curve for each algorithm is displayed in the legend. Our method gives superior results compared to that of Zhu *et al*. [44] and as it is deterministic, results do not vary over different runs.

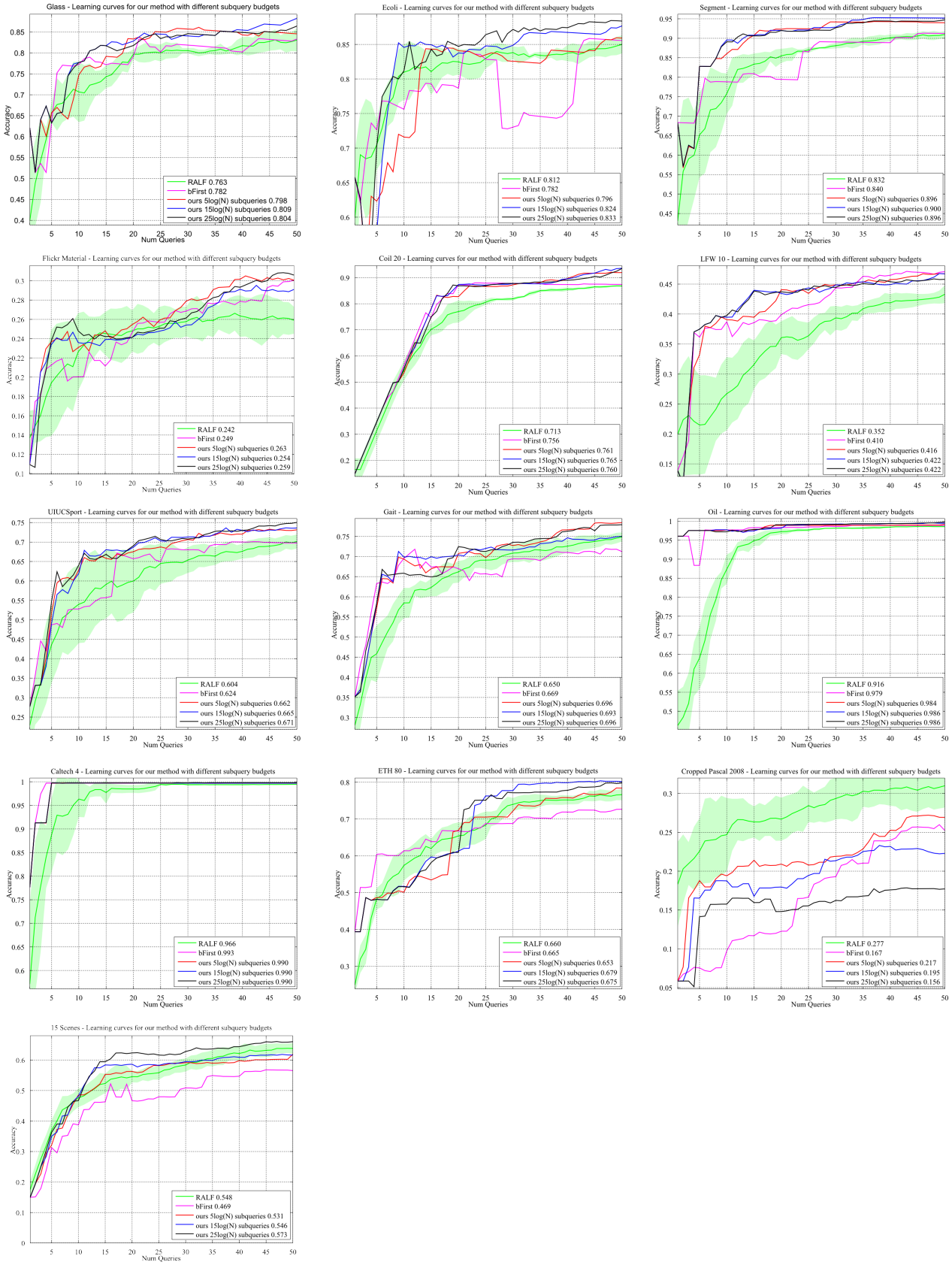

Figure 2. Learning curves comparing the effect of varying the number of subqueries. Our algorithm almost always performs better than RALF, even with surprisingly few subqueries.