Comments to implementation of PROP kernel [1] in ”Scalable kernels for graphs with continuous attributes, NIPS 2013”

Aasa Feragen, Niklas Kasenburg
Machine Learning and Computational Biology Group
Max Planck Institutes Tübingen and DIKU, University of Copenhagen
{aasa,niklas.kasenburg}@diku.dk

Jens Petersen[1], **Marleen de Bruijne**[1,2]
[1]DIKU, University of Copenhagen
[2] Erasmus Medical Center Rotterdam
{phup,marleen}@diku.dk

Karsten Borgwardt
Machine Learning and Computational Biology Group
Max Planck Institutes Tübingen
Eberhard Karls Universität Tübingen
karsten.borgwardt@tuebingen.mpg.de

In [2] the GraphHopper kernel was compared with two different instances of the PROP kernel [1] using two different propagation schemes. These instances were different from those used in [1], as the continuous label distributions described in [1] were replaced by continuous-valued features as indicated in [1, page 4, footnote 3].

In personal communication, Marion Neumann has explained that propagating continuous-valued features, as we do, leads to a smoothing, or averaging, of the continuous node features, leading to poor ability to capture graph structure. This affects our implementation of PROP-diff.

Our implementation of PROP-diff was natural given the claims of generalizability to continuous features and the explanations of how generalizations might be made in [1]. However, the PROP-diff implementation does have limitations as pointed out by Marion Neumann, and potential users should be aware of them.

PROP kernel implementation details. In the PROP-diff variant labels are propagated using the diffusion scheme (see [1]). Our implementation allows discrete or vector-valued labels on the nodes. For the discrete version the label vectors for each node are one-dimensional and contain the actual discrete integer-valued label of the node. These labels are propagated and hashed during the execution of the kernel computation. Note that this differs from the original paper, where a label distribution is used. In the non-discrete version the continuous-valued attributes of the nodes are used instead of a label distribution. So for a graph G with n nodes and d-dimensional label attributes the label matrix will be $n \times d$, where each row corresponds to the attribute of this node. Those labels are propagated with the diffusion scheme and hashed in each iteration.

In the PROP-WL variant the labels of the nodes are first discretized by hashing. Then the Weisfeiler-Lehman propagation scheme is used with those discrete labels as input. Note that if the node labels are already discretized this will be the same as applying the WL kernel directly.

For both variants two hashing functions are implemented: the total variance (TV) distance and the Hellinger distance (see [1]). However, in our results only the TV distance was used as the Hellinger distance only works for positive continuous node attributes.

Acknowledgements

We thank Marion Neumann for her clarifications.

References

[1] M. Neumann, N. Patricia, R. Garnett, and K. Kersting. Efficient graph kernels by randomization. In *ECML/PKDD (1)*, pages 378–393, 2012.

[2] A. Feragen, N. Kasenburg, J. Petersen, M. de Bruijne, and K.M. Borgwardt. Scalable kernels for graphs with continuous attributes. In *NIPS*, pages 216–224, 2013.