Graph Kernels for Molecular and Reduced Graphs

A. Demco and C. Saunders

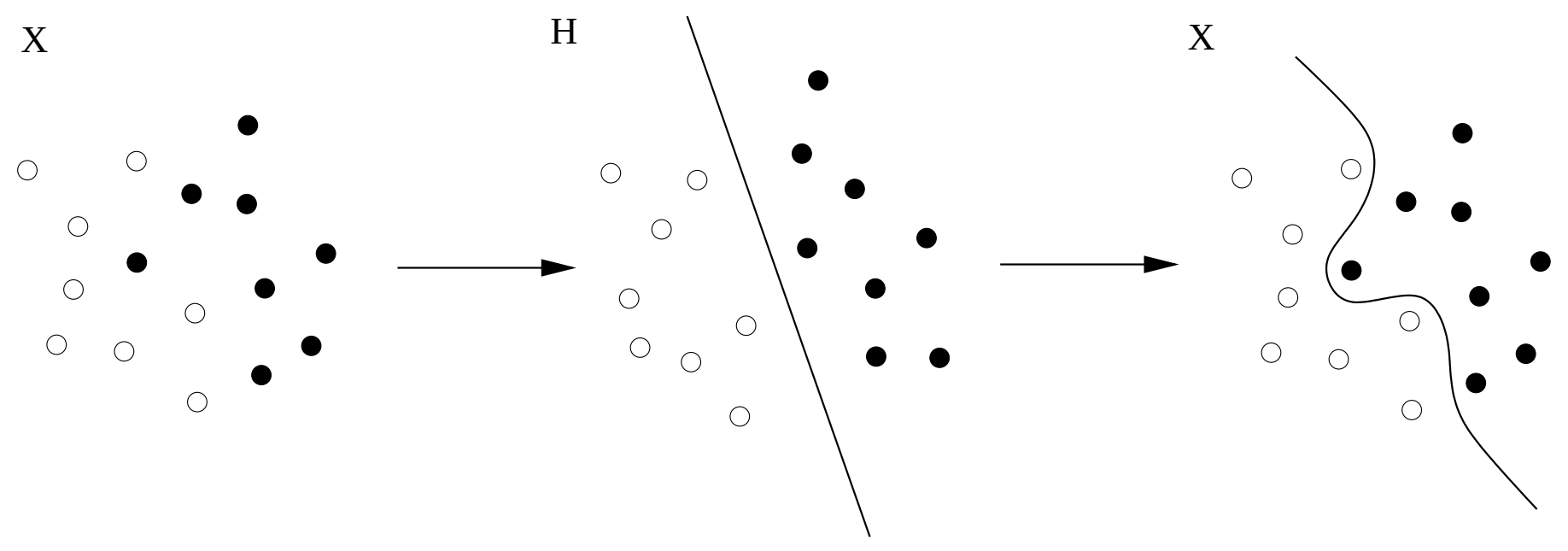
ISIS Group, School of Electronics and Computer Science, University of Southampton, Southampton, UK

Abstract

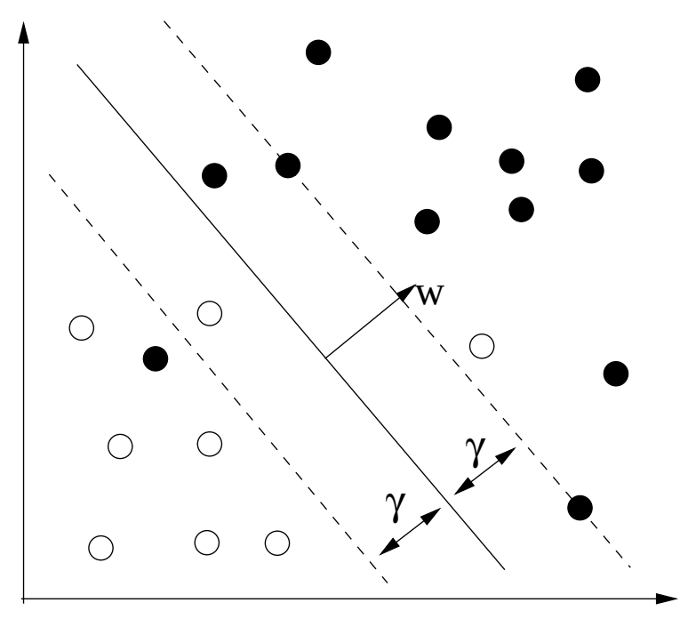
The development of structured kernel methods has had a significant impact on a number of domains and allows for a wide range of analysis including classification, clustering and ranking. In particular defining kernel functions between graphs provides the basis of an efficient algorithm to compare and classify molecular structures. Molecular data is naturally represented as a graph where nodes represent atom types and edges represent bond types. Furthermore, reduced graphs can be derived from molecular graphs, and present an alternate graph-based representation. Graph kernels can be constructed in many different ways e.g. using features such as walks, cycles and trees. Here we develop two new graph kernels which extend walk-based graph kernels and allow soft matching and gaps to be considered. These extensions have particular application to molecular data because walks which do not have matching atom-labels but contain matching topological pharmacophore labels can be included but down-weighted appropriately. Here we test these kernels using known extensions such as coloring and non-tottering. Both of our extensions increase the flexibility and the applicability of graph kernels to structured data. Specifically here we show that the classification performance for both extensions on the MuTag and NCI-HIV datasets is consistently superior to standard walk-based graph kernels.

Introduction

A set of algorithms, including Support Vector Machines (SVM)s, only use the inner-product of examples. It is possible to replace these inner-products with positive semi-definite kernel functions (a.k.a. the kernel trick). This is equivalent to transforming all input vectors $x_1, \ldots, x_\ell$ to $\phi(x_1), \ldots, \phi(x_\ell) \in \mathcal{H}$, where ϕ is a mapping from the input space $\mathcal{X}$ to the feature space $\mathcal{H}$ ($\phi : \mathcal{X} \to \mathcal{H}$). Next, in this feature space a linear learning algorithm may learn a decision function, which is non-linear in the original feature space. Please see figure below.



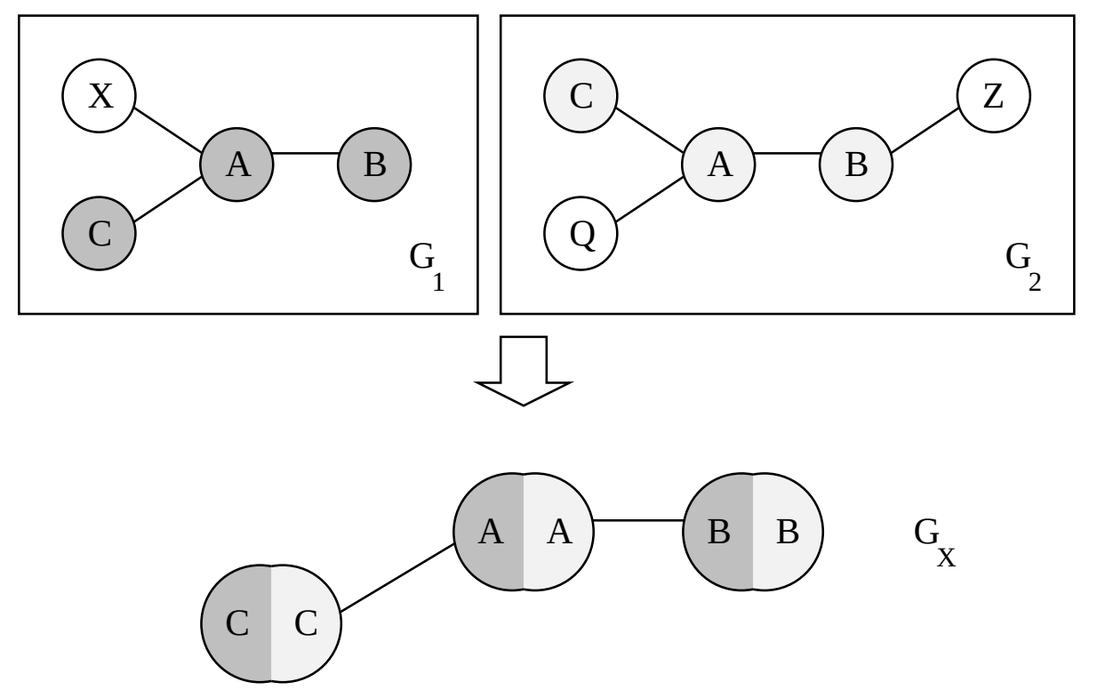
We used SVMs in order to perform binary classification (drug/non-drug) of the MuTag and NCI-HIV datasets. An SVM identifies the maximal-margin hyperplane. As most QSAR problems are non linearly separable, slack parameters are added to the SVM optimization problem to allow some misclassification. Below, γ identifies the margin, and w defines the hyperplane.



A kernel function can be seen as a similarity measure, as the larger the value the closer the two objects are in the feature space. Therefore, expert knowledge can be incorporated when deciding which features to match in the kernel function.

Graph Kernels

A graph kernel is a kernel function which gives the similarity between two graphs. Graph kernels are a type of structured kernel method, which count the number of sub ‘parts’ in common between two structures. Graph kernels exist which use different features including cycles, trees and walks. Using all sub-graphs is too slow [GFW03]. A popular and successful approach uses walks as features. This has been derived as both a marginalized kernel [KTI03] and using product graphs [GFW03]. Both approaches are equivalent as they count the number of walks in common between two-graphs, up to infinite length, down-weighting longer paths. In the following figure we construct a product graph from graphs G_1 and G_2. Note that all walks `CAB` in the product graph, $G_\times$ exist in G_1 and G_2.



First let us consider the adjacency matrix of the product graph $G_\times$, which we denote by $E_\times$. The i, j-th element of the matrix has an entry of 1 if vertex v_i is connected vertex v_j and is zero otherwise. If the matrix is taken to power n, E^n, each i, jth element now lists the number of walks of length n starting at vertex v_i and ending at v_j. Gärtner [GFW03] proposes counting all walks of all lengths (downweighted by a factor λ^n where n is the length of the walk) by using the following kernel:

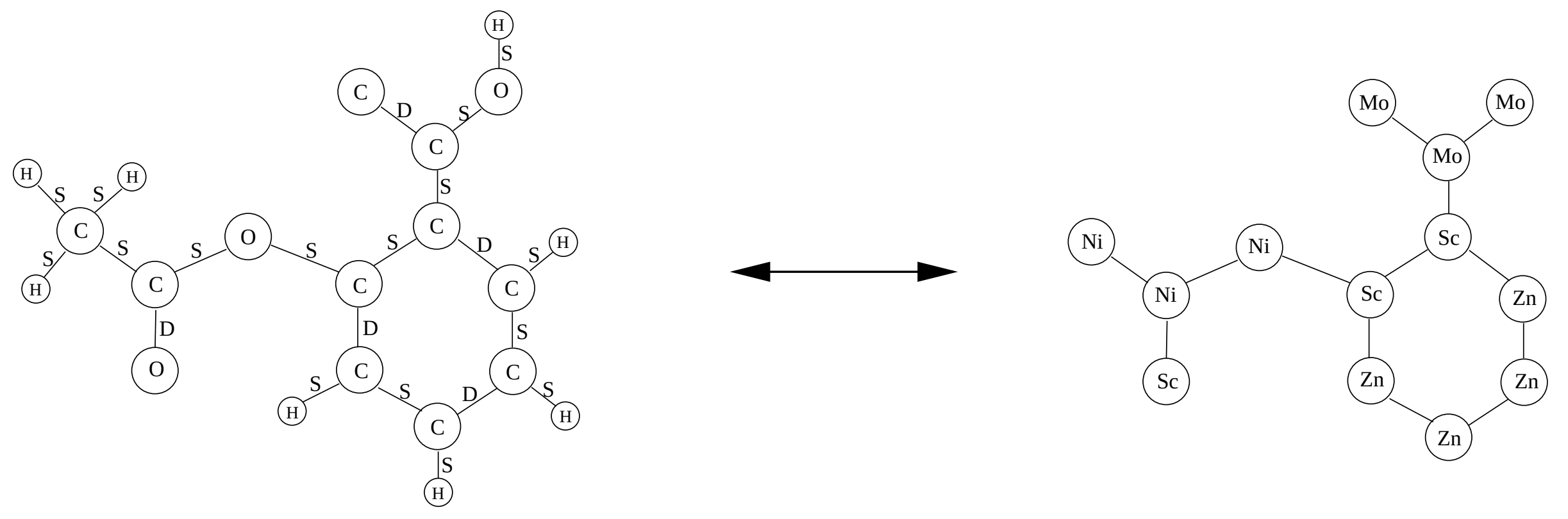
$$K_\times(G_1, G_2) = \sum_{i,j=1}^{|\nu_\times|} \left[\sum_{n=0}^{\infty} \lambda_n E_\times^n \right]_{ij}, \tag{1}$$

which can be calculated by solving using the inverse relation $K_\times(G_1, G_2) = (\boldsymbol{I} - \gamma E_\times)^{-1}$. It takes approximately $O(|\mathcal{V}_\times|^3)$ complexity for this calculation. By calculating the inverse using finite-length approximations we could add in new features including soft-matching and gaps.

Soft-matching walks in graph kernels

Soft-matching has particular application to chemical data sets. There are a range of alternate labels for atoms and bonds (we focus on topological pharmacophore (TP) labels here) that can be used for soft matching. Different matching weights can be assigned based on the importance of the descriptor. When soft-matching with TP labels, this allows to go beyond

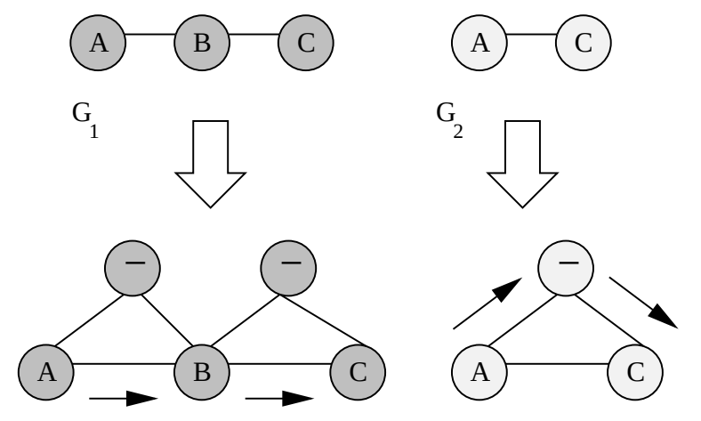
Below, the molecular graph for an aspirin molecule is given (left). Also, a graph which gives the same structure as the molecular graph and is labeled with TP labels for each atom (right). We use the following notation: Zn (linker node), Sc (Aromatic ring), Mo (Negatively ionizable), Ni (Acceptor) for TP labels.



When matching two walks, if they do not have matching atom labels (above left), they can match with a down-weight using the TP labels (above right).

Walks with gaps in graph kernels

A second extension we considered was allowing gaps in walks. A motivation for allowing gaps in walks is that often one part of a molecule can be matched to another part of the molecule which is several atoms away. If we only consider contiguous label sequences, these walks will not be included, although they may be beneficial for comparing two molecules.



Note that walk `ABC` is matched to walk `A-C` in the above graph.

Results

The MuTag dataset consists of 188 compounds along with a binary label describing its Mutagenicity for Salmonella Typhimurium. We obtained the following accuracies for soft-matching (SM) and singly gapped (SG) kernels:

λ	**SM**	**SG**
0.1	**89.4%**	**89.9%**
0.2	**89.4%**	88.8%
0.3	88.8%	**89.9%**
0.4	88.3%	87.8%
0.5	88.3%	85.1%
0.6	88.8%	84.0%
0.7	88.8%	81.4%
0.8	**89.4%**	81.9%
0.9	88.8%	80.9%

We compared our results to the infinite walk-length kernel [MUAV04] which gives the state-of-the-art result on the MuTag dataset. A grid search was conducted over a range of values for the SVM c parameters and the stopping probability p_q. We found that setting $c = 10$ and $p_q = 0.6$ gives the best results and achieves an accuracy of 88.3%.

The NCI-HIV dataset contains 42,689 molecules and a label describing whether its active, moderately active or inactive against the HIV virus. For all experiments, we split all actives into 5 folds and matched these with 500 inactives and performed a cross-validation. Using reduced graphs with a contiguous walk kernel we achieved an AUC of 0.906 ± 0.029. With soft-matching to any label we achieved an AUC of 0.916 ± 0.026. With a singly gapped kernel we achieved an AUC of 0.899 ± 0.040.

Both soft-matching and gaps improve the classification performance using the NCI-HIV and MuTag datasets. This work was submitted as a journal paper [DS07]. In future work we plan to experiment with different combinations of soft-matching and gaps.

References and Acknowledgements

References

[DS07] A. Demco and C. Saunders. Soft and gappy graph kernels for molecular data. *Machine Learning Journal Special Issue Graph Kernels*, 2007. Submitted.

[GFW03] T. Gärtner, P. A. Flach, and S. Wrobel. On graph kernels: Hardness results and efficient alternatives. In *LTKM*, volume 2843, pages 129–143. Springer Verlag, 2003.

[KTI03] H. Kashima, K. Tsuda, and A. Inokuchi. Marginalized kernels between labeled graphs. In *Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003)*, Washington DC, USA, 2003.

[MUAV04] P. Mahé, N. Ueda, T. Akutsu, and J. Vert. Extensions of marginalized graph kernels. In *Proceedings of the 21st International Conference on Machine Learning (ICML-2004)*, Banff, Alberta, Canada, 2004.

Acknowledgements

The first author was supported by a studentship funded by GlaxoSmithKline (GSK).