Kernel on Bag of Paths For Measuring Similarity of Shapes

F. Suard, A. Rakotomamonjy and A. Bensrhair

Laboratoire ITIS EA 4051
INSA/Universite de Rouen
Avenue de l'universite, 76800 Saint Etienne du Rouvray

Abstract.   A common approach for classifying shock graphs is to use a dissimilarity measure on graphs and a distance based classifier. In this paper, we propose the use of kernel functions for data mining problems on shock graphs. The first contribution of the paper is to extend the class of graph kernel by proposing kernels based on bag of paths. Then, we propose a methodology for using these kernels for shock graphs retrieval. Our experimental results show that our approach is very competitive compared to graph matching approaches and is rather robust.

1   Introduction

Object recognition is still a difficult and challenging problem. For solving such problem, several cues are integrated, which give statistical and structural information about the object. In this paper we will focus on an important cue, which is the object shape. Like many real-world data, as texts or molecular structures, shapes can be represented as graphs. Graphs are obtained after an appropriate skeletonization of the shape [4]. Applying high-level algorithms for classification or clustering suppose the definition of similarity on graphs.

While addressing the problem of shape matching, several approaches have been used to define similarity on graphs [1], like edit-distance [9] or *maximum common subgraph*. We propose to address this problem of measuring shape similarity through the theory of positive definite kernel. Using such theory, it becomes possible to define a kernel function that acts as an inner product on the graph space.

In this paper, we first show that many graph-based kernels are built upon two ingredients : path generation and set of paths similarity measures. After, having highlighted this point, we propose other graph-based kernels that differ in how the similarity between set of paths is computed. Then we use these kernels for addressing a problem of shape retrieval, and we show that this approach compares favorably to current approach while opening interesting perspectives for statistical and structural shape classification.

2 Bag-of-paths based graph kernel

Let us introduce the notation that will be used throughout the paper. Define V as a finite set of vertices and $E \subset V \times V$ a set of edges. A graph G is defined as $G = (V, E)$. For a labeled graph, a labeling function is also defined. This function $l : V \cup E \to \mathcal{X}$ assigns a label $l(x)$ to any vertices or edges x.

Here, all steps concerning the shape transformation in a graph have been borrowed from the literature. Shape skeletonization has been performed by means of Dimitrov et al. algorithm [4] while the skeleton to graph transformation we applied is rather similar to the one proposed by Di Ruberto [7].

2.1 Set of paths

A path h of length n on a graph G can be defined by a finite-length sequence of vertices as $h = (v_1, \cdots, v_n)$ with $\forall i \in [1, n-1], (v_i, v_{i+1}) \in E$. So h can be defined as a subgraph of G and can result from the transversal graph with a random walk or by any other methods such as collecting all the paths between any two pairs of vertices. We can then represent G with a set of m paths : $\{h_1, h_2, ..., h_m\}$.

This new representation seems to be less informative since some structural information are lost, however, the advantage is that similarity between graphs can be based on the similarity between these bags of paths.

The similarity between paths is defined with the kernel function $K_L(h, h')$. Usually, $K_L(h, h') = 0$ if their lengths are different and otherwise we have :

$$K_L(h, h') = K_v(l(v_1), l(v_1')) \prod_{i=2}^{n} K_e(l(v_{i-1}, v_i), l(v_{i-1}', v_i')) K_v(l(v_i), l(v_i')) \quad (1)$$

This latter equation suggests that the kernel on path K_L needs the definition of kernels on vertex label K_v and edge label K_e. For this work, when considering kernel on paths, we will use equation 1.

In this work, we propose to represent a graph according to the set of shortest paths between each vertex instead of using all paths obtained by random walks. The problem of finding the shortest path between two vertices is already a classical result in the domain of graph theory. Throughout this paper, we will use the Dijsktra's algorithm.

Since all paths are preprocessed, it becomes easy to discard some paths from the set to reduce the computational time. For instance, we can handle the maximal path length easily. The last advantage is that a set of paths is of finite cardinality and thus, no convergence problem appears. Furthermore, considering shortest paths between vertices naturally prevents from tottering phenomenom.

2.2 Merging path similarity measures

One possible approach [11] for building a kernel on sets is to compute a similarity score between elements of each set according to a so-called minor kernel and then

is to merge the resulting scores into a higher level similarity score that defines
the inner product between the two sets.

The graph kernel of Kashima et al.[6] fits into this bag-of-paths kernel frame-
work. In fact, for this kernel each graph is represented by a set of paths (of
possibly infinite dimension) obtained through a random walk on the graph.

The graph kernel is obtained by merging all pair of paths similarity mea-
sures into a single score, using a weighted averaged approach : $K(G_1, G_2) = \sum_{h_1, h_2 \in V_1^\star, V_2^\star} p_1(h_1) p_2(h_2) K_L(h_1, h_2)$, where p_1 and p_2 are some probability dis-
tributions on the set of finite-length sequences of vertices $V_1^\star$ and $V_2^\star$. Actually,
only paths have positive probabilities under p_1 and p_2 and these probabilities
distributions are defined according to the path generation on the graph.

In the following paragraph, we look for different methods to merge path
similarity measures. Supposing that each graph G_1, G_2 has been transformed
into a set of paths respectively P_1 and P_2. A first easy way for obtaining a graph
kernel would be the mean average kernel :

$$K(G_1, G_2) = K(P_1, P_2) = \frac{1}{N_1} \frac{1}{N_2} \sum_{i:h_i \in P_1} \sum_{j:h_j \in P_2} K_L(h_i, h_j) \qquad (2)$$

where N_1 and N_2 are respectively the cardinality of the sets P_1 and P_2. This
kernel is very simple but has the disadvantage of using all pairs of similarity
between paths. Hence, a large number of pairs of paths with low similarity
measures can "hide" a large similarity between two paths. For addressing such
problem, it is possible to consider a matching kernel [11] :

$$K(G_1, G_2) = K(P_1, P_2) = \frac{1}{2}[\hat{K}(P_1, P_2) + \hat{K}(P_2, P_1)] \qquad (3)$$

with $\hat{K}(P_1, P_2) = \frac{1}{|P_1|} \sum_{i:h_i \in P_1} \max_{j:h_j \in P_2} K_L(h_i, h_j)$. This kernel aims at match-
ing each path of P_1 with a path of P_2, which is an interesting approach but leads
to a non positive-definite kernel. Although such non-positive kernel can be used
for learning, we propose here a positive-definite approximation of this matching
kernel. $\max_{j:h_j \in P_2} K_L(h_i, h_j)$ can be approximate with $\sum_{j:h_j \in P_2} K_{d_L}(h_i, h_j)$
[5]. Indeed, the kernel $K_{d_L}(h_1, h_2) = \exp\left(-\frac{d_L(h_1, h_2)^2}{2\sigma^2}\right)$ is positive definite for
all $\sigma > 0$. d_L is the distance induced by the kernel K_L. Our positive-definite
matching kernel is then :

$$\hat{K}(P_1, P_2) = \frac{1}{|P_1|} \frac{1}{|P_2|} \sum_{i:h_i \in P_1} \sum_{j:h_j \in P_2} K_{d_L}(h_i, h_j) \qquad (4)$$

With this approximation, we include in the kernel values all the similarity mea-
sures between pairs of paths, but with an exponentially decreasing influence as
the distance d_L between the two paths increases. Hence, the contribution of the
most matching path is still large compared to other paths.

Another approach is based on the kernel on sets described by Desobry et al
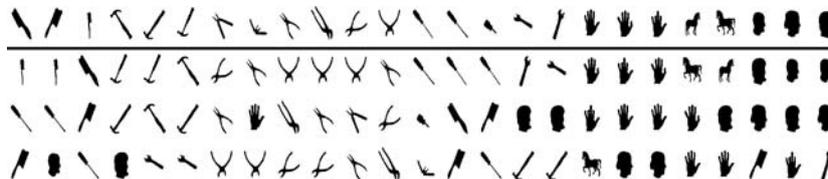[3]. Two graphs can be considered as similar if their respective sets of paths

Fig. 1: Query result with the Rudger tools dataset. First line images represent the query and then we have from up to bottom the most similar objects according to a path level-set kernel.

occupy the same part of the space of finite-length sequences. According to Desobry and Davy, this boils down to measure the similarity of the two sets of path P_1 and P_2 by comparing the support of the probability distribution of each set of paths. Since, we have to deal with estimating such support, Desobry et al. propose to use a one-class SVM. The estimation of the distribution support is obtained as a solution of the optimization problem :

$$\min_{f \in \mathcal{H}, b} \frac{1}{2} \|f\|_{\mathcal{H}}^2 + \frac{1}{\nu n} \sum_i \max(0, b - f(x_i)) - b$$

where $\mathcal{H}$ is the reproducing kernel Hilbert space induced by the kernel on path K_L, which we suppose is so that $K_L(h, h) = 1$, $\nu \in [0, 1]$ is a regularization parameter that is directly related to a given level-set of the distribution support [8]. The distribution support of each set of paths S_1 and S_2 of are obtained by applying the one-class svm to each set and the contour of S_1 and S_2 are respectively : $f_{P_1}(h) = \sum_i \alpha_i^{P_1} K_L(h_i, h) - b_{P_1}$ and $f_{P_2}(h) = \sum_j \alpha_j^{P_2} K_L(h_j, h) - b_{P_2}$. Then, we have defined the inner product between supports and thus between the graphs generating the paths as :

$$K(G_1, G_2) = K(P_1, P_2) = \langle b_{P_1}, b_{P_2} \rangle \cdot \sum_{i: h_i \in P_1} \sum_{j: h_j \in P_2} \alpha_i^{P_1} \alpha_j^{P_2} K_L(h_i, h_j) \qquad (5)$$

According to the property of positive definite kernels, the resulting kernel is positive definite.

3 Application to Shock Graphs Mining

We have tested our graph kernel approach on a shape retrieval problem. We compare our approach to state-of-the art algorithm for graph matching [10, 2, 9] on the Rudgers tool database. This database contains 25 objects separated in 8 different classes five of which can be categorized as "tool" whereas the three other are biological shapes.

The shape retrieval problem is the following : each of the 25 objects has been used as a shape query and for each query, we rank the 24 other shapes by
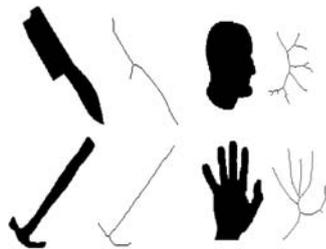
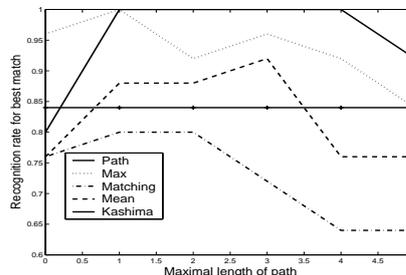Fig. 2: Examples of shapes and associated skeleton graphs.

Fig. 3: Performance on the best match for different graph kernels according to the length of path.

decreasing similarity to the query shape (the similarity being defined according to the distance induced by the graph kernel). For an ideal similarity measure, supposing that the query shape belongs to a class with n elements, the $n-1$ first ranked shapes should belong to the same class of the query.

We have used the distance to the center of the mass and the distance to the nearest shape edge as node labels and only the normalized length as edge label. All the features have been normalized and belonging to the interval $[0, 1]$. Several parameters have to be fixed for our similarity measure which are the minor kernel for edges and vertices. In all our experiment, we have used a gaussian kernel for all these kernels. Thus, for this comparison, we have fixed the gaussian kernel widths σ to 0.1 after having perform a complete test to choose the optimal parameter. For the parameter of the gaussian in the matching kernel we have used the same value.

When considering the best ranked shapes, Sebastian et al. [9] and Demirci et al. [2] respectively reported 1 and 1 mismatching retrieved shapes, which corresponds to 96% and 96% recognition rate. Figure 1 reports our results for different kernels and for different lengths of paths used for processing the kernel.

When path length is equal 0, the max matching kernel performs very well with a recognition performance of 96% whereas other path-based kernel gives performances lower than 90%. As the considered path length increases, the recognition rate increases then decreases when a given path length is reached. Kernels for path length equal to 0 have been computed considering only node similarities. We can conclude that the path length can bring discriminant informations on the shape. For this problem, it seems that a path length of 2 or 3 seems to be a good compromise.

We can also note that, although the labels used for nodes and edges are rather simple, the path level-set based kernel (eq. 5) and the max matching kernel (eq.3) are able to retrieve a correct shape on all the queries. For this experiment, the path level-set kernel seems to be robust to the path length since it provides perfect recognition rate for a path length from 1 to 4. Figure 1 clearly shows that the matching kernel (eq. 4) and the mean kernel (eq. 2) performs

poorly compared to the max matching kernel that they approximated. This lack of performance may be due to bad kernel parameters. It illustrates the price that has to be paid for having a positive definite approximation of a kernel.

Figure 1 gives an example of shape retrieval for the path level-set kernel for a maximal path length of 2. We can see that the best matches are all correct but several incorrect matches have been obtained for the second best matches. These errors are essentially due to the incapacity to make the difference between the class "brush" and "screwdriver". A rationale for this may be that labels used for nodes and edges are not discriminating enough for these two classes.

4   Conclusions

In this paper, we have shown that using graph kernels can be a good alternative to graph matching algorithm for measure graph similarity. At first, we have shown that many graph kernels can be considered as a bag of paths kernel and from this observation, we have provided another graph kernel. We then used this kernel for measuring shape similarities.

Results prove that this approach is very promising since owing to graph kernel and the kernel on paths, the similarity measure can be enriched with statistical information about the object. For instance, vertices can be labeled by local histograms or by texture features extracted locally in a window centered on a vertex of the skeleton.

References

[1] H. Bunke and K. Shearer. A graph distance metric based on the maximal common subgraph. *Pattern Recognition Letters*, 19:255–259, 1998.

[2] F. Demirci, A. Shokoufandeh, L. Bretzner, and S. Dickinson. Object recognition as many-to-many feature matching. *Internation Journal of Computer Vision*, 69(2):203–222, 2006.

[3] F. Desobry, M. Davy, and W.J. Fitzgerald. A class of kernels for sets of vectors. In *Proceedings of the 13th European Symposium on Artificial Neural Networks*, 2005.

[4] P. Dimitrov, C. Phillips, and K. Siddiqi. Robust and efficient skeletal graphs. In *Conference on Computer Vision and Pattern Recognition*, 2000.

[5] B. Haasdonk and C. Bahlmann. Learning with distance substitution kernels. In Springer, editor, *Pattern Recognition - Proc. of the 26th DAGM Symposium*, 2004.

[6] H. Kashima, K. Tsuda, and A. Inokuchi. Marginalized kernels between labeled graphs. In *Proceedings of the Twentieh International Conference on Machine Learning*, 2003.

[7] C. Di Ruberto. Recognition of shapes by attributed skeletal graphs. *Pattern Recognition*, 37(1):21–31, 2004.

[8] B. Scholkopf and A. Smola. *Learning with Kernels*. MIT Press, 2001.

[9] T. Sebastian, P. Klein, and B. Kimia. Recognition of shapes by editing shock graphs. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(5):550–571, 2001.

[10] K. Siddiqi, A. Shokoufandeh, S. J. Dickinson, and S. W. Zucker. Shock graphs and shape matching. *International Journal of Computer Vision*, 35:13–32, 1999.

[11] C. Wallraven, B.Caputo, and A. Graf. Recognition with local features: the kernel recipe. In *Proceedings of International Conference on Computer Vision*, pages 257–264, 2003.