Clustering of Attributed Graphs and unsupervised synthesis of Function-Described Graphs

Alberto Sanfeliu
Inst. de Robòtica i Informàtica
Univ. Politècnica de Catalunya
asanfeliu@iri.upc.es

Francesc Serratosa
Dept. d'Enginyeria Informàtica
Universitat Rovira i Virgili
Francesc.Serratosa@etse.urv.es

René Alquézar
Dept. Lleng. i Sist. Informàtics
Univ. Politècnica de Catalunya
alquezar@lsi.upc.es

Abstract

Function-Described Graphs (FDGs) have been introduced by the authors as a representation of an ensemble of Attributed Graphs (AGs) for structural pattern recognition alternative to first-order random graphs. Both optimal and approximate algorithms for error-tolerant graph matching, which use a distance measure between AGs and FDGs, have been reported elsewhere. Also, the supervised synthesis of FDGs from a set of graphs with a common labelling has been addressed previously. In this paper, the unsupervised synthesis of FDGs is studied in the context of clustering a set of AGs and obtaining an FDG model for each cluster. Two algorithms based on incremental and hierarchical clustering, respectively, are proposed, which are parameterized by a graph matching method. Results on 3D-object recognition show that these algorithms are effective for clustering a set of AGs and synthesising the FDGs that describe the classes.

1. Introduction

Function-Described Graphs (FDGs) have been introduced by the authors as a representation of an ensemble of *Attributed Graphs* (AGs), alternative to first-order random graphs, for structural pattern recognition [1,5]. The supervised synthesis of FDGs from a labeled set of AGs or FDGs was studied in [1]. Moreover, a distance measure between an AG and an FDG has been reported in [7] and a branch-and-bound algorithm for error-tolerant graph matching (*etgm*), which computes this distance measure has been reported recently [5]. Here, two unsupervised algorithms are given to cluster a set of AGs and synthesise the FDGs describing each cluster.

Given a set of unlabelled AGs, we do not, in general, have any way of synthesising FDGs that represent the ensemble unless we can first establish a common labelling. However, given a set of isomorphisms to a common underlying structure, an FDG can be generated from the ensemble using the process *FDG-synthesis-from-labelled-AGs* presented in [1]. We would like to choose the common labelling so as to minimise the measures of dissimilarity between the given AGs and the resulting FDGs. This global optimisation problem does not lead to a computationally practical method for choosing the labelling, because there are too many possible orientations to consider, especially when the number and order of the AGs is high. Therefore, two sub-optimal methods to synthesise FDGs from a set of unlabelled AGs are proposed. In the former, the FDGs are updated by the AGs, which are sequentially introduced, as in [4]. It has the advantage that the learning and classification process can be interleaved, i.e. the classification does not need to wait for all the input instances, but is available after processing each AG. The main drawback of this incremental approach is that, given a set of unlabelled AGs, different FDGs could be synthesised from them depending on the order of presentation of the AGs. To infer some unique FDGs, we propose a hierarchical method, which is carried out by merging pairs of FDGs with minimal distance successively, as in [6]. The drawback here is that the full ensemble of AGs is needed to generate the FDGs. Since a distance measure between FDGs has not been presented yet, the original distances between the AGs of the ensemble are used in this second approach. The hierarchy of AGs (dendogram) can be computed by a *complete method* or by a *single method*, depending on whether the similarity of an object to a cluster is taken as its similarity to the farthest or to the closest member within the cluster, respectively [2].

When the ensemble of patterns is composed of several classes, we need a clustering process to synthesise an FDG for each class. Thus, a distance measure threshold d_α is introduced to determine the splitting condition. In the following, we present the *Incremental-clustering-of-AGs* and the *Hierarchical-clustering-of-AGs* methods to cluster a set of AGs and synthesise the corresponding FDG for each class. If the given AGs are considered to belong to the same class and a single FDG is desired, these methods can be applied using a large distance threshold such that the splitting condition is never fulfilled and, therefore, only one FDG is synthesised.

2. Incremental clustering of AGs

Algorithm 1 computes the *Incremental-clustering-of-AGs*. It generates some clusters from a sequence of AGs and synthesises a set of FDGs, one for each cluster. The algorithm uses two procedures presented in [1]: *FDG-synthesis-from-labelled-AGs*, to transform an AG into an equivalent FDG, and *FDG-synthesis-from-labelled-FDGs*, to build an FDG from two FDGs with a given labelling. The procedure *Extend-labelling-AG-FDG* extends with null elements the AG and the FDG to make them structurally isomorphic and also extends the given labelling accordingly. The procedure *Update-an-FDG-with-an-AG* synthesises an FDG from a previous FDG, a new AG, and a labelling between them. The clustering method relies on (and is parameterised by) a distance threshold d_α and a matching algorithm $M(G,F)$ that is supposed to return an optimal (or a "good" suboptimal) labelling between an AG *G* and an FDG *F*, according to an appropriate distance measure (see [5]).

Algorithm 1: *Incremental-clustering-of-AGs*
Inputs: A sequence of AGs $G_1,...G_m$, $m \geq 1$, over a common domain.
A matching algorithm $M(G,F)$ between an AG and an FDG that finds an optimal or suboptimal labelling according to a distance measure [5].
A threshold d_α on the distance between an AG and an FDG.
Output: A set of FDGs $C^{F'}$ that represents the clusters of $G_1,...G_m$.
Begin
$n := 1$
{build the first FDG from G_1 using the method described in [1]}
F_n := FDG-synthesis-from-labelled-AGs(G_1)
for $i := 2$ **to** m **do**
  for $j := 1$ **to** n **do**
    let $d_j : G_i, F_j \to \Re$ and $\mu_j : G_i \to F_j$ be the distance and labelling found by $M(G_i, F_j)$
    let $d_x = min\{d_k : 1 \leq k \leq j\}$
  end-for
  if $d_x \leq d_\alpha$ **then**
    (G'_i, F'_x, μ'_x):=*Extend-labelling-AG-FDG* (G_i, F_x, μ_x)
    F_x := *Update-an-FDG-with-an-AG* (G'_i, F'_x, μ'_x)
  else
    {build F_{n+1} from G_i using the synthesis described in [1]}
    F_{n+1} := FDG-synthesis-from-labelled-AGs(G_i)
    $n := n+1$
  end-if
end-for
Let $C^{F'} = \{F_1, F_2, ..., F_n\}$
end-algorithm

Procedure *Extend-labelling-AG-FDG* (G, F, μ) **returns** (G', F', μ')
if μ is not bijective **then**
Let $\mu' : G' \to F'$ be a bijective mapping that extends μ by extending G to G' and F to F' with null vertices and arcs appropriately
  else
    $\mu' := \mu$; $G' := G$; $F' := F$
  end-if
end-procedure

Procedure *Update-an-FDG-with-an-AG* (G', F', μ') **returns** F
{build H from G' using the method described in [1]}
H := FDG-synthesis-from-labelled-AGs(G')
let $\gamma : G' \to H$ be a bijective mapping used in the previous synth.
let $\varphi : H \to F'$ be the bijective mapping determined by the composition $\mu' = \varphi \circ \gamma$
{build F using the synthesis procedure described in [1] with 2 FDGs}
F := FDG-synthesis-from-labelled-FDGs(H, F', φ)
end-procedure

3. Hierarchical clustering of AGs

Algorithm 2 computes the *Hierarchical-clustering-of-AGs*. It generates some clusters from a set of AGs by a single or complete agglomerative method and synthesises the FDGs that describe the classes obtained. The clustering method is parameterised by a matching algorithm between AGs, $M(G,G')$, a distance threshold d_α, and a boolean value indicating the type of agglomerative method. Three main procedures are used. In the first one, called *Labelling-and-distance-between-AGs*, the distance measures and labellings between AGs are obtained. In the second procedure, called *Initial-FDGs-and-labellings*, these extended AGs and labellings are transformed into equivalent FDGs and labellings. At that point, the initial patterns are described in structurally isomorphic FDGs, which represent only one AG. The distance between them is the one obtained between the AGs and their labellings are obtained as a composition of the labellings between their equivalent AGs. In the last procedure, called *Agglomerative-clustering-of-FDGs*, a set of FDGs is generated by a complete or single method. The new feature of this procedure is that, while creating the hierarchy of AGs, when two clusters are merged (i.e. F_x and F_y are synthesised together), not only the distance measure between the clusters are redefined, but also the labelling between them. Note that it is not possible to use any *average clustering method* [2] since an "average labelling" between two FDGs cannot be defined. As in the incremental approach, the procedures *FDG-synthesis-from-labelled-AGs* and *FDG-synthesis-from-labelled-FDGs* are used as well.

Algorithm 2: *Hierarchical-clustering-of-AGs*
Inputs: A set of AGs $C^G = \{G_1,...,G_m\}$, $m \ge 1$, over a common domain.
A matching algorithm $M(G,G')$ between AGs that finds an optimal or good suboptimal labelling according to a distance between AGs.
A threshold d_α on the distance between AGs.
A boolean value *complete* which is true or false depending on whether a complete or a single agglomerative method is desired.
Output: A set of FDGs $C^{F'}$ that represents the clusters of $G_1,...,G_m$.
Begin
{Find the set of distances between AGs $C^d = \{d_{i,j} : 1 \le i, j \le m : i < j\}$
and the associated labellings $C^{f'} = \{f'_{i,j} : 1 \le i, j \le m : i < j\}$,
obtain the set of structurally isomorphs AGs $C^{G'} = \{G'_i : 1 \le i \le m\}\}$
$(C^{G'}, C^{f'}, C^d)$:=*Labelling-and-distance-between-AGs*(C^G, M)
{Define the set of FDGs consisting of one AG $C^F = \{F_i : 1 \le i \le m\}$
and the bijective lab. between FDGs $C^\varphi = \{\varphi_{i,j} : 1 \le i, j \le m : i < j\}\}$
(C^F, C^φ):=*Initial-FDGs-and-labellings*$(C^{G'}, C^{f'})$
{Build an FDG F by a single or complete hierarchical method}
$C^{F'}$:=*Agglomerative-clustering-of-FDGs*$(C^F, C^\varphi, C^d, d_\alpha, \text{complete})$
end-algorithm

Proc. *Labelling-and-distance-between-AGs*(C^G, M) **return** (C^f, C^d)
for all $G_i, G_j \in C^G$
Let $f_{i,j}: G_i \to G_j$ and $d_{i,j}: G_i, G_j \to \Re$ found by $M(G_i, G_j)$
end-for-all
Let $C^f = \{f_{i,j} : 1 \le i, j \le m; i < j\}$;
Let $C^d = \{d_{i,j} : 1 \le i, j \le m; i < j\}$
end-procedure

Procedure *Initial-FDGs-and-labellings*$(C^{G'}, C^{f'})$ **returns** (C^F, C^φ)
for all $G'_i \in C^{G'}$
{build the FDG F_i from G'_i using the synthesis described in [1]}
F_i := FDG-synthesis-from-labelled-AGs(G'_i)
let $\gamma_i : G'_i \to F_i$ be the bijective mapping used in the previous synth.
end-for-all
for all $F_i, F_j : 1 \le i, j \le m : i < j$
let $\varphi_{i,j} : F_i \to F_j$ be a mapping defined by $\varphi_{i,j} = \gamma_j \circ f'_{i,j} \circ \gamma_i^{-1}$
end-for-all
Let $C^\varphi = \{\varphi_{i,j} : 1 \le i, j \le m; i < j\}$
Let $C^F = \{F_i : 1 \le i \le m\}$
end-procedure

Proc. *Agglomerative-clustering-of-FDGs*$(C^F, C^\varphi, C^d, d_\alpha, \text{complete})$ **returns** $C^{F'}$
Let $C^F = \{F_i : 1 \le i \le m\}$; Let $C^{F'} = C^F$;
Let $C^\varphi = \{\varphi_{i,j} : 1 \le i, j \le m; i < j\}$; Let $C^d = \{d_{i,j} : 1 \le i, j \le m; i < j\}$;
Let $d_{x,y}$ be the minimum distance in C^d
while $d_{x,y} \le d_\alpha$ **do**
F_y := FDG-synthesis-from-labelled-FDGs(F_x, F_y, $\varphi_{x,y}$)
Remove F_x from $C^{F'}$
for $i := 1$ **to** m **do**
if complete **then {** *complete-method* **}**
if $i < x \wedge d_{i,x} > d_{i,y}$ **then**
$d_{i,y} := d_{i,x}$; $\varphi_{i,y} := \varphi_{x,y} \circ \varphi_{i,x}$; $d_{i,x} := \infty$
end-if
if $i > x \wedge d_{x,i} > d_{y,i}$ **then**
$d_{y,i} := d_{x,i}$; $\varphi_{y,i} := \varphi_{x,i} \circ \varphi_{y,x}$; $d_{x,i} := \infty$
end-if
else { ** *single-method* **}
if $i < x \wedge d_{i,x} < d_{i,y}$ **then**
$d_{i,y} := d_{i,x}$; $\varphi_{i,y} := \varphi_{x,y} \circ \varphi_{i,x}$; $d_{i,x} := \infty$
end-if
if $i > x \wedge d_{x,i} < d_{y,i}$ **then**
$d_{y,i} := d_{x,i}$; $\varphi_{y,i} := \varphi_{x,i} \circ \varphi_{y,x}$; $d_{x,i} := \infty$
end-if
end-if
end-for
end-while
end-procedure

4. Results

The contribution of FDGs to structural pattern recognition is illustrated by the three-dimensional object recognition problem. The original data was composed by 101 AGs, which represent the semantic and structural information of the views taken from five objects. Vertices in the AGs represent the faces, with one attribute, which is the area of the face. Arcs represent the edges between faces, with one attribute, which is the length of the edge. Five FDGs were built from the AGs that represent their views using a supervised synthesis method [1]. See [5] for more details.

Some classification tests have been carried out to show the capability of the FDGs to represent an ensemble of AGs. In the first tests, we applied a supervised clustering, that is, the FDGs were synthesised using the AGs that belong to the same 3D object. We used four different methods. In the first one, the labellings were manually determined from the original data and the FDGs were synthesised using the procedure *FDG-synthesis-from-labelled-AGs* described in [1]. The number of vertices of the FDGs, shown in the first column of table 1, is similar than the number of faces of the objects. In the rest of methods, there is no a-priori information of the labellings. In the second one, the FDGs were obtained by the procedure *Incremental-clustering-of-AGs*. The number of FDG vertices is shown in the third column of table 1. The AGs were presented to the algorithm as the point of view was rotating around the object. Therefore, the new faces are gradually introduced to the system, minimising the effect of merging different faces in only one FDG vertex. The third and four methods use the algorithm *Hierarchical-clustering-of-FDGs*, with a complete and single agglomerative technique, respectively. The distance measure of Sanfeliu [3] was used to find the distance and the optimal labelling between AGs. Note that, in the hierarchical clustering the number of vertices of the FDGs were reduced (2nd column of Table 1). That means that more than one face of the object is represented by one vertex in the FDG. This is, in part, due to the fact that some faces of these objects are similar and thus, the matching algorithm between AGs matches their corresponding vertices. Then, the clustering process considers these faces the same vertex in the FDG.

# vertices	Synthesis from Labelled AGs	Hierarchical Synthesis	Incremental Synthesis
FDG 1	14	9	11
FDG 2	15	9	10
FDG 3	9	7	11
FDG 4	10	7	9
FDG 5	9	8	11

Table 1. Number of vertices of the FDGs generated by different synthesis methods

Table 2 shows the ratio of correctness of two different classifiers. The former, is based on finding the minimal distance between an AG and each of the FDGs representing the objects [5]. The latter, using the nearest neighbour classifier applying the distance measure between AGs of Sanfeliu [3] as a distance between elements. The test set was obtained from the reference set by applying a zero-mean gaussian noise that modifies the semantic information with a standard deviation of 8.0. The structure was also modified by inserting or deleting one of the vertices. The nearest neighbour classifier only obtained better results than the FDG classifier when the single method was used. The complete clustering obtained better results than the incremental clustering, although it yields FDGs with less vertices.

Supervised	**Clustering**	Correctness
FDG:	Labelled AGs	83.6%
	Incremental	74.6%
	Complete	80.3%
	Single	40.3%
Nearest	**Neighbour**	58.6%.

Table 2. Recognition ratio by supervised clustering.

Table 3 shows the ratio of correctness of the recognition process using non-supervised clustering methods. The distance threshold d_α was set to obtain five classes in all the methods. Note that the non-supervised complete clustering obtains better results (85.5%) than the supervised clustering with labelled AGs (83.6%).

Non-supervised	**Clustering**	# FDGs	d_α	Ratio
FDG:	Incremental	5	15	69.2%
	Complete	5	25	85.5%
	Single	5	25	33.1%

Table 3. Recognition ratio by non-supervised clustering.

5. Conclusions

Two methods for AG clustering using FDGs have been presented. The former is based on an incremental process, which has the advantage that the learning and classification processes can be interleaved. The latter is based on a hierarchical process; the whole ensemble of samples is needed to build the FDGs but the process does not depend on the order of presentation of the AGs as in the first method. Results on 3D-object recognition show that both methods are good to cluster an ensemble of AGs while synthesising an appropriate FDG for each class.

Acknowledgements

This work has been partially supported by the Spanish “Comisión Interministerial de Ciencia y Tecnología”, under project TAP98-0473.

References

[1] R. Alquézar, A. Sanfeliu and F. Serratosa, “Synthesis of Function-Described Graphs”. *Advances in Pattern Recognition, Proc. Joint IAPR Int. Workshops SSPR’98 and SPR’98*, Sydney, Australia, Springer LNCS-1451, pp. 112-121, 1998.

[2] A.D. Gordon, “A review of hierarchical classification”, J. R. Statist Soc, A.150, part 2, pp.119-137, 1987.

[3] A. Sanfeliu and K. Fu, “A distance measure between attributed relational graphs for pattern recognition”, *IEEE Trans. on Syst., Man and Cybern.*, vol. 13, pp. 353-362, 1983.

[4] D.S. Seong, H.S. Kim and K.H. Park, “Incremental Clustering of Attributed Graphs”, *IEEE Trans. on Syst., Man and Cybern.*, vol.23, pp. 1399-1411, 1993.

[5] F. Serratosa, R. Alquézar and A. Sanfeliu, “Function-Described Graphs for structural pattern recognition”, Submitted to *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1999.

[6] A.K.C. Wong and M. You, “Entropy and distance of random graphs with application to structural pattern recognition”, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 7, pp. 599-609, 1985.

[7] F. Serratosa, A. Sanfeliu, and R. Alquézar, “Function-Described Graphs: A measure of similarity based on probabilities”. *Proceedings NSPRIA’99, 8th National Symposium on Pattern Recognition and Image Analysis*, Bilbao, Spain, Vol. I, pp. 421-428, 1999.