Graph Transduction as a Non-Cooperative Game

Aykut Erdem[1] and Marcello Pelillo[2]

[1] Hacettepe University, Beytepe, 06800, Ankara, Turkey
aykut.erdem@hacettepe.edu.tr
[2] "Ca' Foscari" University of Venice, Mestre, Venezia, 30172, Italy
pelillo@dsi.unive.it

Abstract. Graph transduction is a popular class of semi-supervised learning techniques, which aims to estimate a classification function defined over a graph of labeled and unlabeled data points. The general idea is to propagate the provided label information to unlabeled nodes in a consistent way. In contrast to the traditional view, in which the process of label propagation is defined as a graph Laplacian regularization, here we propose a radically different perspective that is based on game-theoretic notions. Within our framework, the transduction problem is formulated in terms of a non-cooperative multi-player game where any equilibrium of the proposed game corresponds to a consistent labeling of the data. An attractive feature of our formulation is that it is inherently a multi-class approach and imposes no constraint whatsoever on the structure of the pairwise similarity matrix, being able to naturally deal with asymmetric and negative similarities alike. We evaluated our approach on some real-world problems involving symmetric or asymmetric similarities and obtained competitive results against state-of-the-art algorithms.

1 Introduction

In the machine learning community, *semi-supervised learning* (SSL) has gained considerable popularity over the last decade [3,19] and within the existing paradigms, graph-based approaches to SSL, namely the *graph transduction* methods, constitute an important class. These approaches model the geometry of the data as a graph with nodes corresponding to the labeled and unlabeled points and edges being weighted by the similarity between points, and try to estimate the labels of unlabeled points by propagating the coarse information available at the labeled nodes to the unlabeled ones. Performing this propagation in a consistent way relies on a common a priori assumption, known as the "*cluster assumption*" [17,3], which states that (1) points which are close to each other are expected to have the same label, and (2) points in the same cluster (or on the same manifold) are expected to have the same label. Building on this assumption, traditional graph-based approaches formalize graph transduction as a regularized function estimation problem on an undirected graph [9,20,17].

In this paper, we present a novel framework for graph transduction, which is derived from a game-theoretic formulation of the competition between the multi-population of hypotheses of class membership. Specifically, we cast the problem of graph transduction as a *multi-player non-cooperative game* where the players are the data points that play a classification game over and over until an equilibrium is reached in their respective strategies. In this game, the strategies played by the labeled points are already decided at the outset, as each of them knows which class it belongs to. On the other hand, the strategies available to unlabeled points are the whole set of hypotheses of being a member of one of the provided classes. The players compete with each other by selecting their own strategies, each choice obtains support from the compatible ones and competitive pressure from all the others. In the long run, the competition will reduce the population of strategies which assume the hypotheses that do not receive strong support from the rest, while it will allow populations with strong support to flourish. In this study, this evolutionary dynamics is modeled by a classic formalization of natural selection process used in the *evolutionary game theory* [16], commonly referred to as the *replicator dynamics*. It is worth-mentioning that our formulation is intrinsically a *multi-class* approach and does not impose any constraint on the value of the payoffs (similarities); in particular, *payoffs do not have to be nonnegative or symmetric*.

The remainder of this paper is structured as follows. In Section 2, we review basic notions from non-cooperative game theory. In Section 3, we formulate graph transduction in terms of a non-cooperative multi-player game. In Section 4, we present our experimental results on a number of real-world classification problems. Finally, in Section 5, we conclude the paper with a summary and directions for future work.

2 Non-cooperative games and Nash equilibria

Following the notations used in [16], a game with many players can be expressed in normal form as a triple $G = (\mathcal{I}, S, \pi)$, where $\mathcal{I} = \{1, \ldots, n\}$, with $n \geq 2$, is the set of *players*, $S = \times_{i \in \mathcal{I}} S_i$ is the *joint strategy space* defined as the Cartesian product of the individual pure strategy sets $S_i = \{1, \ldots, m_i\}$, and $\pi : S \to \mathbb{R}^n$ is the *combined payoff function* which assigns a real valued payoff $\pi_i(s) \in \mathbb{R}$ to each *pure strategy profile* $s \in S$ and player $i \in \mathcal{I}$.

A *mixed strategy* of player $i \in \mathcal{I}$ is a probability distribution over its pure strategy set S_i, which can be described as the vector $x_i = (x_{i1}, \ldots, x_{im_i})^T$ such that each component x_{ih} denotes the probability that the player chooses to play its h^{th} pure strategy among all the available strategies. Mixed strategies for each player $i \in \mathcal{I}$ are constrained to lie in the *standard simplex* of the m_i-dimensional Euclidean space $\mathbb{R}^{m_i}$, $\Delta_i = \{x_i \in \mathbb{R}^{m_i} : \sum_{h=1}^{m_i} x_{ih} = 1, \text{ and } x_{ih} \geq 0 \text{ for all } h\}$. Accordingly, a *mixed strategy profile* $x = (x_1, \ldots, x_n)$ is defined as a vector of mixed strategies, each $x_i \in \Delta_i$ representing the mixed strategy assigned to player $i \in \mathcal{I}$, and each mixed strategy profile lives in the *mixed strategy space* of the game, given by the Cartesian product $\Theta = \times_{i \in \mathcal{I}} \Delta_i$.

For the sake of simplicity, let $z = (x_i, y_{-i}) \in \Theta$ denote the strategy profile where player i plays strategy $x_i \in \Delta_i$ whereas other players $j \in \mathcal{I} \setminus \{i\}$ play based on the strategy profile $y \in \Theta$, that is to say, $z_i = x_i$ and $z_j = y_j$ for all $j \neq i$. The expected value of the payoff that player i obtains can be determined by a weighted sum for any $i, j \in \mathcal{I}$ as

$$u_i(x) = \sum_{s \in S} x(s) \pi_i(s) = \sum_{k=1}^{m_j} u_i\left(e_j^k, x_{-j}\right) x_{jk} \tag{1}$$

where $u_i\left(e_j^k, x_{-j}\right)$ denotes the payoff that player i receives when player j adopts its k^{th} pure strategy, and $e_j^k \in \Delta_j$ stands for the *extreme mixed strategy* corresponding the vector of length m_j whose components are all zero except the k^{th} one which is equal to one.

The *mixed best replies* for player i against a mixed strategy $y \in \Theta$, denoted by $\beta_i(y)$, is the set of mixed strategies which is constructed in such a way that no other mixed strategy other than the ones included in this set gives a higher payoff to player i against strategy y, defined as the set $\beta_i(y) = \{x_i \in \Delta_i : u_i\left(x_i, y_{-i}\right) \geq u_i\left(z_i, y_{-i}\right)\ \forall z_i \in \Delta_i\}$. Subsequently, the combined mixed best replies is defined as the Cartesian product of best replies of all the players $\beta(y) = \times_{i\in\mathcal{I}}\beta_i(y) \subset \Theta$.

Definition 1. *A mixed strategy $x^* = (x_1^*, \ldots, x_n^*)$ is said to be a Nash equilibrium if it is the best reply to itself, $x^* \in \beta(x^*)$, that is*

$$u_i(x_i^*, x_{-i}^*) \geq u_i(x_i, x_{-i}^*) \tag{2}$$

for all $i \in \mathcal{I}, x_i \in \Delta_i$, and $x_i \neq x_i^*$. Furthermore, a Nash equilibrium x^* is called strict *if each x_i^* is the unique best reply to x^*, $\beta(x^*) = \{x^*\}$*

Nash equilibrium constitutes the key concept of game theory. It is proven by Nash that any non-cooperative game with finite set of strategies has at least one mixed Nash equilibrium [11]. The algorithmic issue of computing a Nash equilibria for the proposed transduction game will be discussed later in Section 3.2.

3 Graph transduction game (GTG)

Consider the following *graph transduction game*. Assume each player $i \in \mathcal{I}$ participating in the game corresponds to a particular point in a data set $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ and can choose a strategy among the set of strategies $S_i = \{1, \ldots, c\}$, each expressing a certain hypothesis about its membership to a class and $|S_i|$ being the total number of classes. Hence, the mixed strategy profile of each player $i \in \mathcal{I}$ lies in the c-dimensional simplex Δ_i. By problem definition, we can categorize the players of the game into two disjoint groups: those which already have the knowledge of their membership, which we call *determined players* and denote them with the symbol $\mathcal{I}_\mathcal{D}$, and those which don't have any idea about this in the beginning of the game, which are hence called *undetermined players* and correspondingly denoted with $\mathcal{I}_\mathcal{U}$.

The so-called determined players of the game can further be distinguished based on the strategies they follow without hesitation, coming from their membership information. In formal terms, $\mathcal{I}_\mathcal{D} = \{\mathcal{I}_{\mathcal{D}|1}, \ldots, \mathcal{I}_{\mathcal{D}|c}\}$, where each disjoint subset $\mathcal{I}_{\mathcal{D}|k}$ stands for the set of players always playing their k^{th} pure strategies. It thus follows from this statement that each player $i \in \mathcal{I}_{\mathcal{D}|k}$ plays its extreme mixed strategy $e_i^k \in \Delta_i$. In other words, x_i is constrained to belong to the minimal face of the simplex Δ_i spanned by $\{e_i^k\}$. In this regard, it can be argued that the determined players do not play the game to maximize their payoffs since they have already chosen their strategies. In fact, the transduction game can be easily reduced to a game with only undetermined players $\mathcal{I}_\mathcal{U}$

where the definite strategies of determined players $\mathcal{I}_{\mathcal{D}}$ act as bias over the choices of undetermined players.

It should be noted that any instance of the proposed transduction game will always have a Nash equilibrium in mixed strategies [11]. Recall that, for the players, such an equilibrium corresponds to a steady state such that each player plays a strategy that could yield the highest payoff when the strategies of the remaining players are kept fixed, and it provides us a globally consistent labeling of the data set. Once an equilibrium is reached, the label of a data point (player) i is simply given by the strategy with the highest probability in the equilibrium mixed strategy of player i as $y_i = \arg\max_{h \leq c} x_{ih}$.

3.1 Defining payoff functions

A crucial step in formulating transduction as a non-cooperative game is how the payoff function of the game is specified. Here, we make a simplification and assume that the payoffs associated to each player are additively separable, and this makes the proposed game a member of a special subclass of multi-player games, known as *polymatrix games* [8,7]. Formally speaking, for a pure strategy profile $s = (s_1, \ldots, s_n) \in S$, the payoff function of every player $i \in \mathcal{I}$ is in the form:

$$\pi_i(s) = \sum_{j=1}^{n} A_{ij}(s_i, s_j) \tag{3}$$

where $A_{ij} \in \mathbb{R}^{c \times c}$ is the *partial payoff* matrix between players i and j. It follows that, in terms of a mixed strategy profile $x = (x_1, \ldots, x_n)$, the payoffs are computed as $u_i(e_i^h) = \sum_{j=1}^{n} (A_{ij} x_j)_h$ and $u_i(x) = \sum_{j=1}^{n} x_i^T A_{ij} x_j$.

In an instance of the transduction game, since each determined player is restricted to play a definite strategy of its own, all of these fixed choices can be reflected directly in the payoff function of a undetermined player $i \in \mathcal{I}_{\mathcal{U}}$ as follows:

$$u_i(e_i^h) = \sum_{j \in \mathcal{I}_{\mathcal{U}}} (A_{ij} x_j)_h + \sum_{k=1}^{c} \sum_{j \in \mathcal{I}_{\mathcal{D}|k}} A_{ij}(h, k) \tag{4}$$

$$u_i(x) = \sum_{j \in \mathcal{I}_{\mathcal{U}}} x_i^T A_{ij} x_j + \sum_{k=1}^{c} \sum_{j \in \mathcal{I}_{\mathcal{D}|k}} x_i^T (A_{ij})_k \tag{5}$$

Now, we are left with specifying the partial payoff matrices between each pair of players. Let the geometry of the data be modeled with a weighted graph $\mathcal{G} = (\mathcal{X}, \mathcal{E}, w)$ in which $\mathcal{X}$ is the set of nodes representing both labeled and unlabeled points, and $w : \mathcal{E} \rightarrow \mathbb{R}$ is a weight function assigning a similarity value to each edge $e \in \mathcal{E}$. Representing the graph with its weighted adjacency matrix $W = (w_{ij})$, we set the partial payoff matrix between two players i and j as $A_{ij} = I_c \times w_{ij}$ where I_c is the identity matrix of size c[3]. Note that when partial payoff matrices are represented in block form

[3] The rationale for specifying partial payoffs in this way depends on the analysis of graph transduction on a unweighted undirected graph. Due to the page limit, the details will be reported in a longer version.

as $A = (A_{ij})$, the matrix A is given by the Kronecker product $A = I_c \otimes W$. Our experiments demonstrate that in specifying the payoffs, it is preferable to use the normalized similarity data matrix $\widehat{W} = D^{-1/2} W D^{-1/2}$ where $D = (d_{ii})$ is the diagonal degree matrix of W with its elements given by $d_{ii} = \sum_j w_{ij}$.

3.2 Computing Nash equilibria

In the recent years, there has been a growing interest in the computational aspects of Nash equilibria. The general problem of computing a Nash equilibrium is shown to belong to the complexity class PPAD-complete, a newly defined subclass of NP [4]. Nevertheless, there are many refinements and extensions of Nash equilibria which can be computed efficiently and moreover, the former result does not apply to certain classes of games. Here, we restrict ourselves to the well-established evolutionary approach [16], initiated by J. Maynard Smith [10]. This dynamic interpretation of the concept imagines that the game is played repeatedly, generation after generation, during which a selection process acts on the multi-population of strategies, thereby resulting in the evolution of the fittest strategies. The selection dynamics is commonly modeled by the following set of ordinary differential equations:

$$\dot{x}_{ih} = g_{ih}(x) x_{ih} \tag{6}$$

where a dot signifies derivative with respect to time, and $g(x) = (g_1(x), \ldots, g_n(x))$ is the growth rate function with open domain containing $\Theta = \times_{i \in \mathcal{I}} \Delta_i$, each component $g_i(x)$ being a vector-valued growth rate function for player i. Hence, g_{ih} specifies the growth rate at which player i's pure strategy h replicates. It is generally required that the function g be *regular* [16], *i.e.* (1) g is Lipschitz continuous and (2) $g_i(x) \cdot x_i = 0$ for all $x \in \Theta$ and players $i \in \mathcal{I}$. While the first condition guarantees that the system (6) has a unique solution through every initial state, the condition $g_i(x) \cdot x_i = 0$ ensures that the simplex Δ_i is invariant under (6).

The class of regular selection dynamics includes a wide subclass known as *payoff monotonic dynamics*, in which the ratio of strategies with a higher payoff increase at a higher rate. Formally, a regular selection dynamics (6) is said to be payoff monotonic if

$$u_i\left(e_i^h, x_{-i}\right) > u_i\left(e_i^k, x_{-i}\right) \Leftrightarrow g_{ih}(x) > g_{ik}(x) \tag{7}$$

for all $x \in \Theta$, $i \in \mathcal{I}$ and pure strategies $h, k \in S_i$.

A particular subclass of payoff monotonic dynamics, which is used to model the evolution of behavior by imitation processes, is given by

$$\dot{x}_{ih} = x_{ih} \left[\sum_{l \in S_i} x_{il} \Big(\phi_i\left[u_i\left(e_i^h - e_i^l, x_{-i}\right)\right] - \phi_i\left[u_i\left(e_i^l - e_i^h, x_{-i}\right)\right] \Big) \right] \tag{8}$$

where $\phi_i(u_i)$ is a strictly increasing function of u_i. When ϕ_i is taken as the identity function, *i.e.* $\phi_i(u_i) = u_i$, we obtain the multi-population version of the replicator dynamics:

$$\dot{x}_{ih} = x_{ih} \left(u_i(e_i^h, x_{-i}) - u_i(x) \right) \tag{9}$$

The following theorem states that the fixed points of (9) are Nash equilibria.

Theorem 1. *A point $x \in \Theta$ is the limit of a trajectory of (9) starting from the interior of Θ if and only if x is a Nash equilibrium. Further, if point $x \in \Theta$ is a strict Nash equilibrium then it is asymptotically stable, additionally implying that the trajectories starting from all nearby states converge to x.*

Proof. See [16].

In the experiments, we utilized the following discrete-time counterpart of (9), where we initialize the mixed strategies of each undetermined player to uniform probabilities, *i.e.* the barycenter of the simplex Δ_i.

$$x_{ih}(t+1) = x_{ih}(t)\frac{u_i(e_i^h)}{u_i(x(t))} \tag{10}$$

The discrete-time replicator dynamics (10) has the same properties as the continuous version (See [12] for a detailed analysis). The computational complexity of finding a Nash equilibrium of a transduction game using (10) can be given by $\mathcal{O}(kcn^2)$, where n is the number of players (data points), c is the number of pure strategies (classes) and k is the number of iterations needed to converge. In theory, it is difficult to predict the number of required iterations, but experimentally, we noticed that it typically grows linearly on the number of data points[4]. We note that the complexity of popular graph transduction methods such as [20,17] is also close to $\mathcal{O}(n^3)$.

4 Experimental Results

Our experimental evaluation is divided into two groups based on the structure of similarities that arise in the problems. Basically, we test our approach on some real-world problems involving *symmetric* or *asymmetric* similarities. It is noteworthy to mention that the standard methods are restricted to work with *symmetric* and *non-negative* similarities but our game-theoretic interpretation imposes no constraint whatsoever, being able to naturally deal with *asymmetric* and *negative* similarities alike.

4.1 Experiments with symmetric similarities

We conducted experiments on three well-known data sets: *USPS*[5], *YaleB* [5] and *20-news*[6]. *USPS* contains images of hand-written digits 0-9 down-sampled to 16×16 pixels and it has 7291 training and 2007 test examples. As used in [17], we only selected the digits 1 to 4 from the training and test sets, which gave us a total of 3874 data points. *YaleB* is composed of face images of 10 subjects captured under varying

[4] We observed that the dynamics always converged to a fixed point in our experiments with symmetric and asymmetric similarities. It should be added that in the asymmetric case, the convergence is in fact not guaranteed since there is no Lyapunov function for the dynamics. Still, by Theorem 1, if the dynamics converges to a fixed point, it will definitely be a Nash equilibrium.

[5] http://www-stat.stanford.edu/~tibs/ElemStatLearn/

[6] http://people.csail.mit.edu/jrennie/20newsgroups/

poses and illumination conditions. As in [2], we down-sampled each image to 30×40 pixels and considered a subset of 1755 images which corresponds to the individuals 2, 5 and 8. *20-news* is the text classification data set used in [17], which contains 3970 newsgroup articles selected from the 20-newsgroups data set, all belonging to the topic `rec` which is composed of the subjects `autos, motorcycles, sport.baseball` and `sport.hockey`. As described in [17], each article is represented in 8014-dimensional space based on the TFIDF representation scheme.

For *USPS* and *YaleB*, we treated each image pixel as a single feature, thus each example was represented in 256-, and 1200-dimensional space, respectively. We computed the similarity between two examples $\mathbf{x}_i$ and $\mathbf{x}_j$ using the Gaussian kernel as $w_{ij} = exp(-\frac{d(\mathbf{x}_i,\mathbf{x}_j)^2}{2\sigma^2})$ where $d(\mathbf{x}_i, \mathbf{x}_j)$ is the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ and σ is the kernel width parameter. Among several choices for the distance measure $d(\cdot)$, we evaluated the Euclidean distance $\|\mathbf{x}_i - \mathbf{x}_j\|$ for *USPS* and *YaleB*, and the cosine distance $d(\mathbf{x}_i, \mathbf{x}_j) = 1 - \frac{\langle \mathbf{x}_i,\mathbf{x}_j \rangle}{\|\mathbf{x}_i\|\|\mathbf{x}_j\|}$ for *20-news*.

In the experiments, we compared our approach, which we denote as GTG, against four well-known graph-based SSL algorithms, namely the Spectral Graph Transducer (SGT) [9][7], the *Gaussian fields and harmonic functions* based method (GFHF) [20][8], the *local and global consistency* method (LGC) [17][9] and Laplacian Regularized Least Squares (LapRLS) [1][10]. A crucial factor in the success of graph-based algorithms is the construction of the input graph as it represents the data manifold. To be fair in our evaluation, for all the methods, we used a fixed set of kernel widths and generated 9 different candidate 20-NN graphs by setting $w_{ij} = 0$ if x_j is not amongst the 20-nearest neighbors of x_i. In particular, the kernel width σ ranges over the set $linspace(0.1r, r, 5) \cup linspace(r, 10r, 5))$ with r being the average distance from each example to its 20^{th} nearest neighbor and $linspace(a, b, n)$ denoting the set of n linearly spaced numbers between and including a and b.

In Fig. 1, we show the test errors of all methods averaged over 100 trials with different sizes of labeled data where we randomly select labeled samples so that each set contains at least one sample from each class. As it can be seen, LapRLS method gives the best results for the relatively small data set *YaleB*. However, for the other two, its performance is poor. In general, the proposed GTG algorithm is either the best or the second best algorithm; while its success is almost identical to that of the LGC method in *USPS* and *Yale-B*, it gives superior results for *20-news*.

4.2 Experiments with asymmetric similarities

We carried out experiments on three document data sets – *Cora*, *Citeseer* [14][11], and *WebKB*[12]. *Cora* contains 2708 machine learning publications classified into seven classes,

---

[7] We select the optimal value of the parameter c with the best mean performance from the set $\{400, 800, 1600, 3200, 6400, 12800\}$.

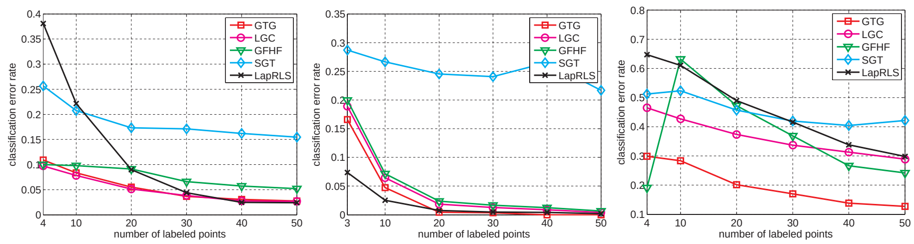
[8] In obtaining the hard labels, we employ the *class mass normalization* step suggested in [20].

[9] As in [17], we set the parameter α as 0.99.

[10] We select the optimal values of the extrinsic and intrinsic regularization parameters γ_A and γ_I from the set $\{10^{-6}, 10^{-4}, 10^{-2}, 1\}$ for the best mean performance.

[11] Both data sets are available at `http://www.cs.umd.edu/projects/linqs/projects/lbc/`

[12] Available at `http://www.nec-labs.com/~zsh/files/link-fact-data.zip`



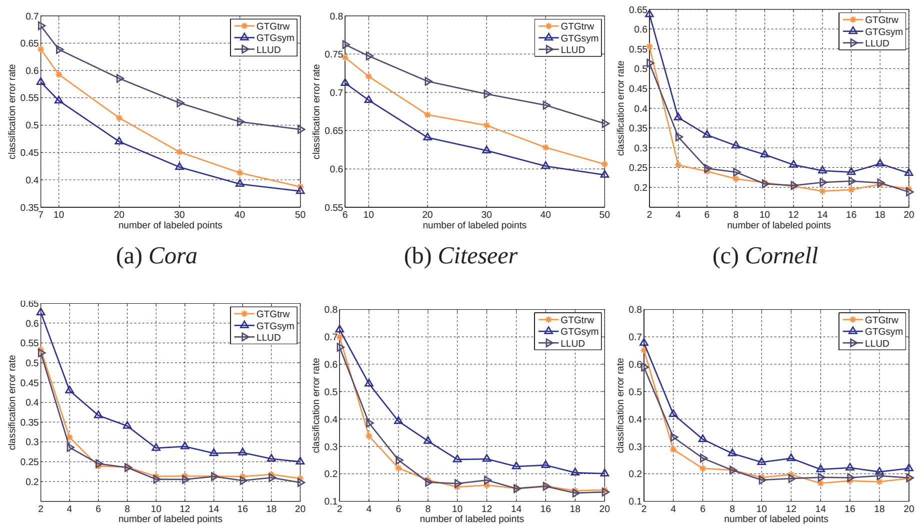
(a) *USPS* (b) *YaleB* (c) *20-news*

Fig. 1: Performance comparisons on classification problems with *symmetric* similarities.

and there are 5429 citations between the publications. *Citeseer* consists of 3312 scientific publications, each of which belongs to one of six classes, and there are a total of 4732 links. *WebKB* contains webpages collected from computer science departments of four universities (*Cornell*, *Texas*, *Washington* and *Wisconsin*), and each classified into seven categories. Following the setup in [18], here we concentrate on classifying student pages from the others. Each subset respectively contains 827, 814, 1166 and 1210 webpages and 1626, 1480, 2218 and 3200 links. In our experiments, as in [18], we only considered the citation structure, even though one can also assign some weights by utilizing the textual content of the documents. Specifically, we worked on the link matrix $W = (w_{ij})$, where $w_{ij} = 1$ if document i cites document j and $w_{ij} = 0$ otherwise.

Unlike our approach, the standard methods mentioned before, namely SGT, GFHF, LGC and LapRLS, are subject to symmetric similarities. Hence, in this context, they can be applied only after rendering the similarities symmetric but this could result in loss of relevant information in some cases. In our evaluation, we restrict ourselves to the graph-based methods which can directly deal with asymmetric similarities. Specifically, we compared our game-theoretic approach against our implementation of the method in [18], denoted here with LLUD. We note that LLUD is based on the notion of random walks on directed graphs and it reduces to LGC in the case of symmetric similarities. However, it assumes the input similarity graph to be strongly connected, so in [18] the authors consider the *teleporting random walk (trw)* transition matrix as input, which is given by $P^\eta = \eta P + (1-\eta)P^u$ where $P = D^{-1}W$ and P^u is the uniform transition matrix. For asymmetric similarity data, we also define the payoffs in terms of this transition matrix and denote this version with GTGtrw. In the experiments, we fixed $\eta = 0.99$ for both LLUD and GTGtrw. To provide a baseline, we also report the results of our approach that works on the symmetrized similarity matrices, denoted with GTGsym. For that case, we used the transformation $\widetilde{W} = 0.5 \times (W + W^T)$ for *SCOP*, and the symmetrized link matrix $\widetilde{W} = (w_{ij})$ for the others, where $w_{ij} = 1$ if either document i cites document j or vice versa, and $w_{ij} = 0$ otherwise.

The test errors averaged over 100 trials are shown in Fig. 2. Notice that the performances of GTGtrw and LLUD are quite similar on the classification problems in *WebKB* data sets. On the other hand, GTGtrw is superior in the multi-class problems



(a) *Cora* (b) *Citeseer* (c) *Cornell*

(d) *Texas* (e) *Washington* (f) *Wisconsin*

Fig. 2: Performance comparisons on classification problems with *asymmetric* similarities.

of *Cora* and *Citeseer*. We should add that symmetrization sometimes can provide good results. In *Cora* and *Citeseer*, GTGsym performs better than the other two methods.

5 Summary and Discussion

In this paper, we provided a game-theoretic interpretation to graph transduction. In the suggested approach, the problem of transduction is formulated in terms of a multi-player non-cooperative game where any equilibrium of the game coincides with the notion of a consistent labeling of the data. As compared to existing approaches, the main advantage of the proposed framework is that there is no restriction on the pair-wise relationships among data points; similarities and thus the payoffs can be negative or asymmetric. The experimental results show that the our approach is not only more general but also competitive with standard approaches. In the future, we plan to continue exploring the generality of our approach when both similarity and dissimilarity relations exist in data [15,6]. Another possible future direction is to focus on improving the efficiency. In our current implementation, we use the standard replicator dynamics to reach an equilibrium but we can study other selection dynamics that are much faster [13].

Acknowledgments

We acknowledge financial support from the FET programme within EU FP7, under the SIMBAD project (contract 213250).

References

1. Belkin, M., Niyogi, P., Sindhwani, V.: Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. J. Mach. Learn. Res. 7, 2399–2434 (2006) 7
2. Breitenbach, M., Grudic, G.Z.: Clustering through ranking on manifolds. In: ICML. pp. 73–80 (2005) 7
3. Chapelle, O., Schölkopf, B., Zien, A. (eds.): Semi-Supervised Learning. MIT Press, Cambridge, MA (2006) 1
4. Daskalakis, C., Goldberg, P.W., Papadimitriou, C.H.: The complexity of computing a Nash equilibrium. Commun. ACM 52(2), 89–97 (2009) 5
5. Georghiades, A., Belhumeur, P., Kriegman, D.: From few to many: Illumination cone models for face recognition under variable lighting and pose. IEEE Trans. on PAMI 23(6), 643–660 (2001) 6
6. Goldberg, A., Zhu, X., Wright, S.: Dissimilarity in graph-based semi-supervised classification. In: AISTATS (2007) 9
7. Howson, J.T.: Equilibria of polymatrix games. Management Science 18(5), 312–318 (1972) 4
8. Janovskaya, E.B.: Equilibrium points in polymatrix games. (in Russian) Litovskii Matematicheskii Sbornik 8, 381–384 (1968), (Math. Reviews 39 #3831.) 4
9. Joachims, T.: Transductive learning via spectral graph partitioning. In: ICML. pp. 290–297 (2003) 1, 7
10. Maynard Smith, J.: Evolution and the theory of games. Cambridge University Press, Cambridge, UK (1982) 5
11. Nash, J.: Non-cooperative games. The Annals of Mathematics 54(2), 286–295 (1951) 3, 4
12. Pelillo, M.: The dynamics of nonlinear relaxation labeling processes. J. Math. Imaging Vis. 7(4), 309–323 (1997) 6
13. Rota Bulò, S., Bomze, I.M.: Infection and immunization: a new class of evolutionary game dynamics. Games and Economic Behaviour (Special issue in honor of John F. Nash, jr.) 71, 193–211 (2011) 9
14. Sen, P., Namata, G.M., Bilgic, M., Getoor, L., Gallagher, B., Eliassi-Rad, T.: Collective classification in network data. AI Magazine 29(3), 93–106 (2008) 7
15. Tong, W., Jin, R.: Semi-supervised learning by mixed label propagation. In: AAAI. pp. 651–656 (2007) 9
16. Weibull, J.W.: Evolutionary Game Theory. MIT Press, Cambridge, MA (1995) 2, 5, 6
17. Zhou, D., Bousquet, O., Lal, T.N., Weston, J., Schölkopf, B.: Learning with local and global consistency. In: NIPS. pp. 321–328 (2004) 1, 6, 7
18. Zhou, D., Huang, J., Schölkopf, B.: Learning from labeled and unlabeled data on a directed graph. In: ICML. pp. 1036–1043 (2005) 8
19. Zhu, X.: Semi-supervised learning literature survey. Tech. Rep. 1530, Computer Sciences, University of Wisconsin-Madison (2005) 1
20. Zhu, X., Ghahramani, Z., Lafferty, J.: Semi-supervised learning using Gaussian fields and harmonic functions. In: ICML. pp. 912–919 (2003) 1, 6, 7