
Transfer Learning Evaluation with Relational Nets

Nathaniel Love

LOVE@CS.STANFORD.EDU

Stanford Logic Group, Stanford University, 353 Serra Mall, Stanford, CA 94305

Abstract

Evaluation of transfer learning systems requires deployment of carefully structured problem instances, composed of source and target subproblems. In general, the difficulty of a transfer learning task is measured by the relationship between the source and target. Relational nets provide a formalism for rigorously defining the source-target relationship in transfer learning tasks.

1. Introduction

The performance benefits of transfer learning (TL) systems are observed by training on a set of source problems, then solving a set of target problems, where performance on the target problems can be improved by transferring learned knowledge from the source problems. If the source and target problems are closely related, TL systems have an easier time performing and benefitting from this transfer. The most challenging transfer learning tasks are characterized by significantly different source and target problems; in the degenerate case the source and target are essentially the same, and transfer learning reduces to traditional machine learning.

In order to evaluate transfer learning systems on problems of varying difficulty, emphasizing different types or degrees of transfer, metrics are needed to classify possible source-target sets. Intuitively, this metric should correspond to the amount of structure shared by the source and target problem instances. The DARPA Transfer Learning Program presented an approach to such a metric by defining eleven transfer levels, organized roughly by increasing difficulty (Defense Advanced Research Projects Agency, 2005), and the Program includes a team of external evaluators charged with the critical task of testing TL

systems across these levels. However, without a formal basis for transfer level definitions, this framework only provides guidance, and evaluators cannot precisely state if or how a source-target set actually comprises a transfer learning task at a particular transfer level.¹

Relational nets provide a representation for transfer learning problems in which mathematical definitions of source-target relationships (and thus transfer levels) can be made; both evaluators and TL systems can benefit from these definitions.

2. Transfer Learning Levels

In the DARPA TL program, transfer learning problem instances for a particular transfer level are described in terms of problem components and their configurations, as well as the solutions, sub-solutions, or action sequences relevant to the problem.

EXAMPLE

The DARPA program (Defense Advanced Research Projects Agency, 2005) defines a *composing* transfer level, in which “new problem instances consist of combinations of components from distinct component sets encountered during training.”

Even given (English) characterizations like these, it is clear that the selection, composition, and configurations of source and target problem components determine which transfer level a particular source-target pair corresponds to². Without a grounding in a formal framework, these level definitions must be interpreted by evaluators within their particular evaluation domain.

If evaluators individually interpret transfer level def-

¹For lack of a better term, “level” will be used in this paper to describe a region of a partition of the space of transfer learning tasks (source-target pairs); no ordering is implied.

²The possible solutions and action sequences are completely determined by the selection of components and their configuration.

initions within their domains of interest, comparing transfer learning results becomes difficult: results for the same TL system on a particular transfer level are incomparable across different evaluation domains. Transfer learning tasks at a particular level in distinctly different domains should share a common source-target structural relationship that does not depend on any implicit assumptions built into either domain. If they do not, evaluation results in these domains are not comparable, and TL systems using particular techniques for structural transfer may get very different results— they can transfer at level 6, but only in domain A , not in domain B .

3. Formalizing the Space of TL Tasks

Multiple definitions of transfer learning levels dependent on domain choice are clearly undesirable. The problems outlined in the previous section can be solved by adopting a structural representation framework that can be applied across domains, and using this structure to mathematically define transfer levels. The source-target relationship for a particular level can be explicitly defined using this common structural representation, and TL systems attempting to perform transfer at a given level can expect to exploit this concrete structural relationship, regardless of what particular domain they are working in. We propose to use the formalism of *relational nets* as a common structural representation framework for transfer learning evaluation.

3.1. Relational Nets

Relational nets are mathematical structures that describe the environment, agents, percepts and actions found in finite, discrete, multi-agent, dynamic systems. The simplest, most direct model for such systems is a set of interacting state machines. In a finite environment with deterministic dynamics³, one can build a state machine with a state for each distinct state of the environment, with outgoing arcs labeled with the joint moves of all agents interacting with the system, and with percepts that are output to each agent interacting with the system. This state machine representation (an example is shown in figure 1) can represent TL tasks in broad classes of domains, including those with “continuous” time and space, since within the simulators for these models these dimensions are made discrete. However, performing reasoning tasks with a state machine is impractical due to the sheer size of the representation, as well as the lack of modularity in

³Nondeterminism can be added to this system by wrapping it into the actions of other agents.

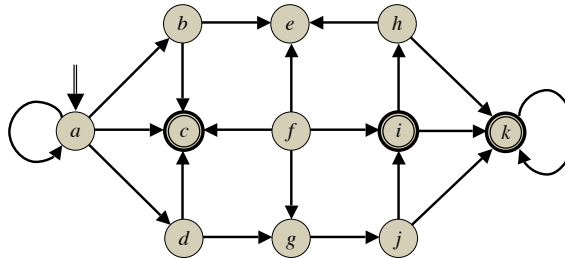


Figure 1. State Machine

the definitions of state and dynamics. A discussion of these issues, and an introduction to propositional nets (precursors to relational nets) can be found in (Love & Genesereth, 2006)

Relational nets provide a solution to these issues, as they provide modular, compact, representations of the same set of systems described by these state machines. In a relational net, the current state of a problem environment is described in terms of objects and the relationships that hold among these objects. A relational net state can be thought of a set of relational databases holding the set of facts true in the current state of the environment. Relational nets also represent the dynamics of the environment in response to actions taken by agents. These dynamics are represented as a collection of update rules (*transitions*), defining the facts true in the next state in terms of the current state and the actions taken by the agents. The relational representation allows for easy definition of goals for agents, terminating conditions, and constraints on agent actions, as each of these can be defined (as *views*) in terms of the facts true in the current state.

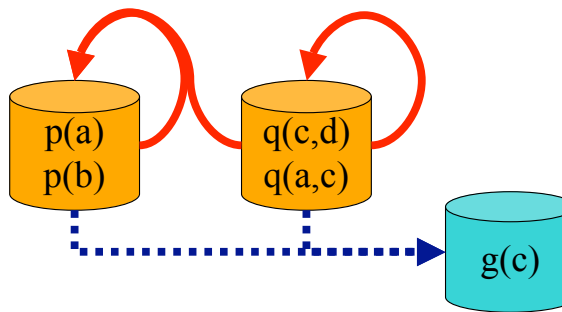


Figure 2. A relational net, with relations p , q , and the view g . The transition into p from both p and q indicates that the next state of p depends on the current states of both p and q .

Explicit definition for relational nets can be found in (Love & Genesereth, 2006). Since the state of a relational net can be represented as a database, the tran-

sitions and views can be constructed using relational algebra operators (joins, selections, and projections). Alternatively, relational nets can be represented using sentences in relational logic; one language for expressing relational nets in this manner is GDL, the Game Description Language used for General Game Playing (Love et al., 2006a). Figure 2 shows a small relational net; the explicit definitions of the transitions (solid lines) and views (dashed lines) are not shown.

3.2. Components and Configurations

The modular nature of relational nets provides a natural interpretation for the components and configurations of a particular transfer learning task. The problem environment can be partitioned into components, each captured by a certain subset of objects, the relationships among those objects, and their dynamics. The dynamics of one component may depend on the state of other components, as well as on agent actions; these dependencies between modular components contribute to the dynamics of a particular relational net. A relational net can be viewed as a union of these interacting components that describes the entire environment. Additionally, the components of a relational net can be configured to shape different problem instances by defining particular initial conditions, terminating conditions, goal conditions, and action constraints for agents.

3.3. State Graphs

A relational net can be expanded back into a *state graph*, where each node is a possible state of the relational net (again, a database consisting of the facts true in this state), and each edge is labeled with combined legal actions of all agents. A directed edge labeled \bar{m} leads from a state s of the relational net to the state s' resulting from the combined action \bar{m} (an action taken by each agent).

The resulting structure is, of course, a state machine—the formalism we began with, but rejected. However, the topology of the state graph forms an additional method of characterizing the relationship between source and target problems. Since every relational net can be expanded back into this structure, TL problem metrics like transfer levels can concretely reference source-target relationships that refer properties of the state graph. For example, when comparing solutions to a source problem to solutions to a target problem, one can compare paths and their labeled edges in the corresponding state graphs.

3.4. Transfer Levels

With concrete notions of TL task components and configurations built from the relational net representation, metrics like transfer learning levels can be explicitly defined. (Love & Tarlow, 2006) contains a selection of relational net descriptions (written in GDL) for TL tasks corresponding to DARPA transfer levels 1-8; the source-target relationships in these relational nets correspond to the definitions given in (Love et al., 2006b), which uses the same framework outlined here.

4. Extending to Evaluation Domains

Relational nets offer concrete benefits for both evaluators and TL system developers, as they provide a framework for building TL tasks based on precise metrics (i.e. transfer levels). When problem instances are already represented as declarative descriptions, the relational net framework may be directly applied; in the case of Gamemaster (Genesereth et al., 2005), a General Game Playing evaluation test bed, TL scenarios are constructed in GDL and correspond directly to relational nets.

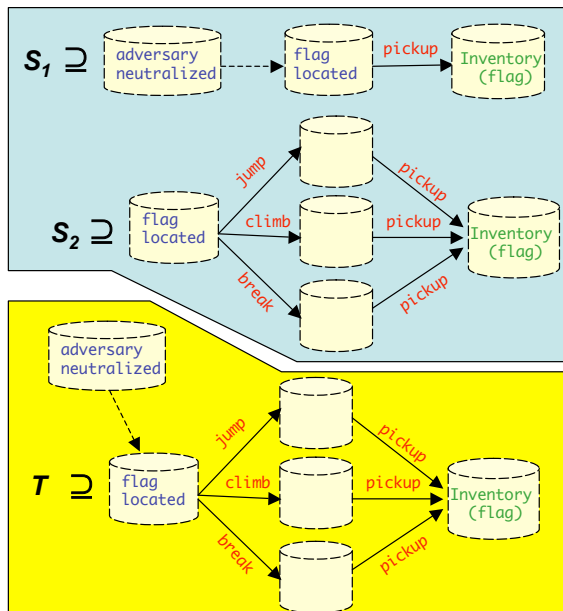


Figure 3. An abstract state graph representation of solution paths in a set of source problems $\{S_1, S_2\}$ and a target problem T in an evaluation domain using the game *Urban Combat*. The corresponding transfer level requires target solution paths to be compositions of source solution paths.

Other domains used for TL evaluation include real-time strategy games, first-person shooters, and robotic soccer or part assembly simulators— these domains

seem to have characteristics (“continuous” time and space, for example) that may make a relational net representation impractical. However, in order to leverage the benefits of the relational net formalism for TL evaluation, the entire domain need not be re-represented as a relational net. Rather, the critical objects and relationships for a particular TL task, along with their dynamics, can be modeled using relational nets, and the source-target relationship verified using the framework. Issues with continuous dimensions can be easily resolved by using qualitative discretization—differences in goals and configurations are predominantly concerned with crossing particular thresholds (moving from one side of an obstacle to the other, for example). In other cases, these conditions are entirely expressed in terms of static state (like capturing a flag). Figure 3 shows an example of a formalized transfer learning task in the game *Urban Combat*. The transfer level implemented here requires that solutions to the target problem be sequential compositions of solutions to the source problem; the relational net representation enables a formal expression of this property.

Even when deploying their systems in non-declarative, continuous game worlds, transfer learning system developers may already use internal formal, relational representations of these worlds. For example, systems leveraging relational reinforcement learning techniques may model game worlds as relational Markov Decision Processes; others may build value functions based on concepts defined in terms of the relations among objects in the current state of the game world. Such internal representations already perform the mapping of these diverse domains into a formal relational representation. Structuring TL tasks for a particular transfer level using the relational net framework may better enable TL system developers to focus their systems on performing transfer at that level.

5. Conclusion and Future Work

Relational net representations of transfer learning tasks provide a framework for a concrete partitioning of the TL space into levels. Using this framework, evaluators and TL system developers can formally decide whether a particular source/target pair form a TL problem for a particular level, and obtain results comparable across domains. Additionally, a TL system working in this level can exploit this relationship to perform transfer. At a particular transfer level, the target problem may be unknown to a TL system in advance, but the components of the source problem, their dynamics, their relationships to goals, and the properties of the state graph combine with the formal

definition of a transfer level to constrain the characteristics of the target problem.

The transfer levels discussed here are drawn from the definitions suggested in the DARPA TL program; other possibilities for partitioning the space of transfer learning tasks should be explored and considered. Because the relational net framework can be used to formalize general source-target relationships, it can be applied to new TL evaluation metrics whenever they are expressed in terms of these relationships. Some additional evaluation metrics incorporate notions of problem difficulty or amounts of background knowledge; expression of these metrics within the relational nets framework is the subject of future work. The relational net formalism has been applied as a framework for evaluation of TL systems in General Game Playing tasks (Love et al., 2006b); applying the framework to TL evaluation tasks other domains is the subject of future work.

Acknowledgments

I thank Tim Hinrichs, Michael Genesereth, Charles Petrie, John Duchi, and Danny Tarlow for their valuable suggestions and assistance. This research has been supported by the DARPA Transfer Learning Program and NRL Grant #N00173-05-1-G033.

References

- Defense Advanced Research Projects Agency (2005). Transfer learning proposer information pamphlet (PIP) for broad agency announcement 05-29.
- Genesereth, M., Love, N., Hinrichs, T., & Duchi, J. (2005). Gamemaster. <http://games.stanford.edu:4000>.
- Love, N., & Genesereth, M. (2006). *Relational nets* (Technical Report). Stanford University, Stanford, CA. Forthcoming.
- Love, N., Genesereth, M., & Hinrichs, T. (2006a). *General game playing: Game description language specification* (Technical Report LG-2006-01). Stanford University, Stanford, CA. <http://logic.stanford.edu/reports/LG-2006-01.pdf>.
- Love, N., Genesereth, M., Petrie, C., Langley, P., Asgharbeygi, N., Fawcett, T., Konik, T., & Stracuzzi, D. (2006b). Constructing transfer learning challenge problems in general game playing. Unpublished.
- Love, N., & Tarlow, D. (2006). Examples for GGP TL evaluation. <http://games.stanford.edu/transfer/>.