Deep Transfer: A Markov Logic Approach

Jesse Davis and Pedro Domingos
Department of Computer Science and Engineering
University of Washington
Seattle, WA 98195
{jdavis, pedrod}@cs.washington.edu

Inductive learning has traditionally been defined as generalizing from training instances to test instances from the same distribution. Decades of research have produced many sophisticated techniques for solving this task. Unfortunately, in real applications, training and test data often come from different distributions. Consequently, when faced with a new problem, standard algorithms cannot reapply knowledge from previously solved tasks. This places a large burden on the modeler, who must specify the features, relations and legal search space as well as acquire an appropriate amount of training data for each new problem. Transfer learning addresses this problem by explicitly assuming that the source and target problem are different. In shallow transfer, test instances are from the same domain as the training instances, but have a different distribution. In deep transfer, test instances are from an entirely different domain (i.e., described by different predicates). Humans routinely perform deep transfer, but few learning systems, if any, are capable of it. Computationally, the missing link is the ability to discover *structural* regularities that apply to many different domains, irrespective of their superficial descriptions. This is arguably the biggest gap between current learning systems and humans. We have developed an approach, called DTM (Deep Transfer via Markov logic), based on a form of second-order Markov logic [4] for addressing this problem. It can be viewed as a way to automatically discover important structural regularities in one domain and apply them in another [1].

Markov logic unifies first-order logic and probability. It softens a logical knowledge base by associating a weight with each formula. Worlds that violate formulas become less likely, but not impossible. The logical formulas capture regularities that hold in the data for a given domain. However, the knowledge that the formulas encode is specific to the types of objects and predicates present in that domain. Deep transfer attempts to generalize learned knowledge across domains that have different types of objects and predicates. In order to abstract away the superficial domain description, DTM uses second-order Markov logic, where formulas contain predicate variables [4] to model common structures among first-order formulas. To illustrate the intuition behind DTM, consider the formulas $\mathtt{Complex}(\mathtt{z},\mathtt{y}) \wedge \mathtt{Interacts}(\mathtt{x},\mathtt{z}) \Rightarrow \mathtt{Complex}(\mathtt{x},\mathtt{y})$ and $\mathtt{Location}(\mathtt{z},\mathtt{y}) \wedge \mathtt{Interacts}(\mathtt{x},\mathtt{z}) \Rightarrow \mathtt{Location}(\mathtt{x},\mathtt{y})$. Both are instantiations of: $\mathtt{r}(\mathtt{z},\mathtt{y}) \wedge \mathtt{s}(\mathtt{x},\mathtt{z}) \Rightarrow \mathtt{r}(\mathtt{x},\mathtt{y})$, where $\mathtt{r}$ and $\mathtt{s}$ are predicate variables. Introducing predicate variables allows DTM to represent high-level structural regularities in a domain-independent fashion. This knowledge can be transferred to another problem, where the formulas are instantiated with the appropriate predicate names. The key assumption that DTM makes is that the target domain shares some second-order structure with the source domain.

DTM works with any learner than induces formulas in first-order logic. Given a set of first-order formulas, DTM converts each formula into second-order logic by replacing all predicate names with predicate variables. It then groups the second-order formulas into cliques. Two second-order formulas are assigned to the same clique if and only if they are over the same set of literals. DTM evaluates which second-order cliques represent regularities whose probability deviates significantly from independence among their subcliques. It selects the top k highest-scoring second-order cliques to transfer to the target domain. Finally, the highest scoring cliques are transferred to the target domain where they guide the structure learner to fruitful parts of the search space. Experiments in bioinformatics, Web and social network domains show that DTM outperforms standard structure learning. In addition to improved empirical performance, DTM discovered patterns that include broadly useful properties of predicates, like symmetry and transitivity, and relations among predicates, such as various forms of homophily.

One promising avenue of future research is to examine unifying DTM with cognitive models of how humans acquire theories about the world. Kemp et al. [3] have proposed an appealing language for modeling how people acquire abstract, intuitive theories. Their language is based on an extension of first-order logic and they provide a set of formulas, or "laws," that can be combined to form theories about the world. For example, given data about interactions within a company, these formulas could be combined to describe the structure of the company. The abstract nature of the theories provide a natural mechanism for transfer. For example, if the structure of a company is best described by a hierarchy, once learned, this concept can reapplied in other domains. We are currently working on combining DTM with Kemp et al.'s language. DTM treats all predicate and object variables as if they are universally quantified. One important contribution will be to consider the different logical quantifiers (e.g., exists, exists at most one, exists exactly one) that are used in Kemp et al.'s language. Doing so will allow us to more compactly represent the key structural regularities that appear in the source domain. For example, if a domain is characterized by interactions among fully connected groups of individuals (i.e., cliques), DTM would discover one set of formulas that captures each size group (e.g., a clique of three people, a clique of four people, etc.). These concepts are unlikely to generalize well to other domains as the knowledge they encode may be too specific to the structure of the source domain. Considering different logical quantifiers will allow us to learn and transfer the high-level concept of a clique, irrespective of its size. Furthermore, we would like to recast DTM within a hierarchical Bayesian framework, which will provide a sounder probabilistic interpretation of DTM's clique scoring and transfer mechanisms. Finally, Kemp et al.'s language has been used to model causality [2] and these extensions may give us the ability to learn and transfer causal structures.

References

[1] Jesse Davis and Pedro Domingos. Deep transfer via second-order Markov logic. In *Proceedings of the 26th International Conference on Machine Learning*, pages 217–224. Omnipress, 2009.

[2] Noah Goodman, Tomer Ullman, and Josh Tenenbaum. Learning a theory of causality. In *Proceedings of the 31st Annual Conference of the Cognitive Science Society*, pages 2188–2193. Cognitive Science Society, 2009.

[3] Charles Kemp, Noah Goodman, and Josh Tenenbaum. Theory acquisition and the language of thought. In *Proceedings of the 30th Annual Conference of the Cognitive Science Society*, pages 1606–1611. Cognitive Science Society, 2008.

[4] Stanley Kok and Pedro Domingos. Statistical predicate invention. In *Procedings of the 24th International Conference on Machine Learning*, pages 433–440. ACM Press, 2007.

Topics: Learning algorithms
Preference: Oral