The Relevance of Relevance *

Devika Subramanian
Department of Computer Science
Rice University
Houston, TX 77005

Russell Greiner[†]
Siemens Corporate Research, Inc
755 College Road East
Princeton, NJ 08540-6632 USA

Judea Pearl
Department of Computer Science
University of California
Los Angeles, CA

1 Introduction

With too little information, reasoning and learning systems cannot work effectively. Too much information can also cause the performance of these systems to degrade, in both accuracy and efficiency. It is therefore important to determine what information must be preserved, or more generally, *i.e.*, what information is "relevant".

Reasoning about what is and is not relevant to the goals at hand is ubiquitous in humans. Much of this meta-theoretic reasoning occurs outside of the scope of our introspection; it is believed that over 90% of the neuronal connections in our brains are inhibitory and serve to discard or suppress inputs received through our senses. Moreover, when we engage in deliberate problem-solving (*e.g.*, writing this introduction, planning a trip, or learning a buyer's profile from credit data), we explicitly reason about the relevance of entities in our conceptualization of the problem. In addition, in the context of computing within a problem formulation, we reason about the relevance of steps in that computation; *e.g.*, which move to explore in game tree search or which variable to ignore in data analysis. The object of relevance reasoning is to simplify problem formulations and to eliminate wasted effort in computation. Relevance reasoning is crucial for agents like us with limited memory and computational resources; the benefit obtained from being able to derive appropriate actions within given resource limits far outweighs the expense of such reasoning.

This special issue, motivated by the very successful AAAI Fall Symposium on this topic [1], addresses questions in automating relevance reasoning, focusing on autonomous systems with limited computational and sensory resources, embedded in dynamic environments. First, what *is* relevance reasoning? Is it different from other kinds of reasoning, such as default reasoning or common sense reasoning? Are there general definitions of relevance that transcend tasks? If relevance is task-specific, what is the relationship between a task and the kind of relevance reasoning appropriate for

it? When is relevance reasoning necessary for autonomous systems? Can we provide a knowledge-level account of relevance reasoning independent of the syntactic details of how it is implemented in particular computational systems? What are the costs and benefits of relevance reasoning, and how do we quantify them?

A key premise underlying the study of relevance is that there are general relevance criteria that can be articulated explicitly, independent of specific tasks and formulations. A related premise is that the cost of relevance reasoning will be offset by the ability to compute good action policies within resource limits. The 11 papers in this special issue put these premises to theoretical and empirical tests.

One general way to define the meta-theoretic notion of relevance is in terms of "perturbations": A specific "entity" (such as an action, training sample, attribute, background proposition, or inference step) is irrelevant to a task in some context if the appropriate response to the task does not change by an unacceptable amount if we change the entity in that context. Otherwise, we view that entity as (somewhat) relevant to the task. This view is explicitly stated in the [Galles+Pearl] paper, which deals with causality and where a perturbation corresponds to a material change in the physical world; it is also implicit in most of the other papers appearing in this specific issue. For instance, the learning papers of [Greiner+Grove+Kogan], [Kivenen+Warmuth+Auer], [Kohavi+John] and [Blum+Langley] all use the fact that an attribute is irrelevant to a classification function if the class label of an instance does not change if we change the value of that attribute. [Baum+Smith] define relevance of individual computational steps; in particular, they say that a leaf expansion in a game tree search computation is relevant if it has the potential to alter the choice of move. [Darwiche], [Levy+Fikes+Sagiv] and [Lakemeyer] provide similar definitions of when a perturbation to a proposition leaves answers to a set of goal propositions unchanged in the context of some background theory.

While there is significant agreement on the general definition of relevance, there is considerable variety in how they are instantiated in the context of specific tasks or task classes. [Baum+Smith] demonstrate the subtleties involved in instantiating this general definition to a specific computation. A leaf expansion in a game tree search could impact the utility of the move under consideration as well as the choice of future leaf expansion decisions. The implementation of their relevance criterion forces them to move away from the traditional model of making point estimates of the value of leaf positions to using parametric value distributions that can be learned during the course of a game. They show that the distinction between important and unimportant lines of play in a game can be captured using relevance measures on distributions over evaluation scores.

The nature of the learning task necessitates different choices in the structure of criteria for determining attribute relevance. Thus, while [Kivenen+Warmuth+Auer] and [Kohavi+John] assume the same attributes are relevant over all instances, [Greiner+Grove+Kogan] and [Baluja+-Pomerleau] consider tasks where different attributes are relevant for different instances, making attribute relevance a function of the specific instance rather than that of an entire instance set. The task in [Baluja+Pomerleau] is to detect lane-marking in an image to help a vehicle follow a road. This task is difficult as the image is often cluttered with distracting extraneous objects (*e.g.*, other vehicles, pedestrians, etc.) that can look superficially like the lane marking that the vehicle is seeking. To avoid these distractions, the system focuses its search on the "relevant" part of the image; this subset is based on its expectation of the lane marking positions, which is based on previous images in the sequence. In contrast, the [Kivenen+Warmuth+Auer] analysis does not require ever explicitly identifying which attributes are relevant — it only uses the observation that there is a set of k relevant attributes in its analysis. [Blum+Langley] survey work in defining and using other relevance criteria in concept learning.

The difficulty in finding realizable implementations of relevance definitions in reasoning is illustrated by [**Galles+Pearl**], [**Levy+Fikes+Sagiv**], [**Darwiche**] and [**Lakemeyer**]. [**Lakemeyer**] postulates semantic definitions of the relevance of a proposition to another in the context of a theory and shows that it is intractable to compute exact relevance relations even under propositional logic. This work motivates the need for approximations as considered by [**Khardon+Roth**] who demonstrate that reasoning can be efficient when restricted to relevant models (*e.g.*, vivid knowledge bases of Levesque) of formulas under consideration. [**Darwiche**] recasts Pearl's definition of conditional independence (irrelevance) in probabilistic reasoning [4] in the logical context and provides several efficient graph-theoretic algorithms for decomposing complex logical calculations into simpler computations. Similarly, [**Galles+Pearl**] show that most aspects of causal relevance can be captured by directed graphs and that such graphs can be used as theorem provers, to test whether new relevance relations follow from, or are consistent with, a set of relevance premises.

Can a pure knowledge level account of relevance reasoning be provided? [**Kohavi+John**] and [**Levy+Fikes+Sagiv**] provide evidence that such an account is either unlikely to be comprehensive or not be very useful. [**Kohavi+John**] empirically demonstrate that the determination of attribute relevance in an induction task cannot be made independent of the induction algorithm (which therefore necessitates a symbol level analysis). [**Levy+Fikes+Sagiv**], building on earlier work by Subramanian in [5], provide a proof-theoretic account of irrelevance in the context of efficient deductive reasoning with large Horn theories, in sharp contrast to [**Lakemeyer**]'s purely semantic account.

Almost all of the papers in the special issue provide theoretical guarantees and/or experimental results on the utility of relevance reasoning. [**Baluja+Pomerleau**] demonstrate that by using relevance analysis to focus the attention on lane pixels in an image, the accuracy of the choice of steering direction is improved by 20% — a significant performance improvement on the task. [**Baum+Smith**] conduct an extensive empirical evaluation of their policy of expanding relevant leaf nodes in the context of Othello. They show that their programs play a much stronger game than alpha-beta pruning and that the superiority of their approach increases rapidly with additional time resources. An interesting open empirical question is whether the [**Baum+Smith**] relevance criterion would benefit the game of chess.

The following papers on logical reasoning demonstrate how logical inference of various forms can be made more efficient by explicit reasoning about relevance. [**Darwiche**] provides conditions under which reasoning about logical conditional independence simplifies complex computations (belief change, entailment and satisfiability) into small sets of much simpler "local" calculations. [**Levy+Fikes+Sagiv**] demonstrates that reasoning with large Horn databases can be significantly speeded up by explicit elimination of irrelevant facts. [**Khardon+Roth**] take a very different approach, showing that models which summarize relevant information in a set of formulas can be the basis of very efficient schemes for logical as well as default reasoning.

The papers on learning also demonstrate that "less is better", *i.e.*, they show that learning algorithms produce more accurate classifiers if trained on only the subset of "relevant" training examples, or if given only the values of the "relevant" attributes of the training examples. The [**Blum+Langley**] review paper uses insights from the machine learning and the computational learning theory communities to summarize results, both characterizing the "state of the art" implemented systems (*e.g.*, into "embedded" vs "filter" vs "wrapper" approaches, contrasting feature selection with feature weighting schemes, and considering both selecting relevant examples, and relevant attributes) and providing useful theoretical explanations. [**Kohavi+John**] explain why a learner, trying to produce an accurate classifier from a limited pool of training examples, can do better if it uses only the subset of relevant attributes. They then present an extensive

empirical comparison between their approach, which involves "wrapping" the basic learner, as a black box, within an algorithm that searches for the optimal subset of features with a "filtering" algorithm, that uses other, learner-independent principles to select the subset of attributes. **[Greiner+Grove+Kogan]** present a learning model where a helpful teacher, who uses the target decision tree to label instances, also tells the learner which attributes in the target tree were used to classify the instance. They show that this relevance information can be extremely useful: *e.g.*, while there is no known algorithm that can "PAC-learn" arbitrary decision trees in the standard model (where the learner sees the complete set of attribute values for each instance) [6], this learning task become trivial if the learner is told exactly which features are relevant for each instance. Finally, Littlestone earlier presented a very clever algorithm for learning linear-seperators, called WINNOW, which had the intriguing property that it scales linearly with the number of relevant attributes but only *logarithmically* with the total number of attributes [2]; by contrast, the typical PERCEPTRON algorithm scales linearly here [3]. As most of these results were only upper bounds, it was not clear whether there *really* was a difference here. The **[Kivenen+Warmuth+Auer]** paper answers this, by showing that an adversary can force the Perceptron algorithm to make exponentially more mistakes (in the context of on-line learning) than WINNOW.

We hope these papers awaken interest in relevance reasoning and inspire new answers to the open questions posed earlier in this introduction.

References

[1] Russell Greiner and Devika Subramanian. Proceedings of the "Relevance" symposium. Technical Report FS-94-02, AAAI, Menlo Park, 1995.

[2] Nick Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning Journal*, 2:285–318, 1988.

[3] Marvin Minsky and Seymour Papert. *Perceptron: An Introduction to Computational Geometry*. The MIT Press, Cambridge, expanded edition, 1988.

[4] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, 1988.

[5] Devika Subramanian. Shift of vocabulary bias in speedup learning. *Machine Learning*, 20:155-191, 1995.

[6] Leslie G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.