Raggi, D., Stockdill, A., Jamnik, M., Garcia Garcia, G., Sutherland, H. E. A., & Cheng, P. C. H. (2020). Dissecting Representations. In A.-V. Pietarinen, P. Chapman, L. Bosveld-de Smet, V. Giardino, J. Corter, & S. Linker (Eds.), Diagrammatic Representation and Inference (pp. 144-152). Cham: Springer.
doi:10.1007/978-3-030-54249-8_11

Dissecting representations*

Daniel Raggi[1], Aaron Stockdill[1], Mateja Jamnik[1],
Grecia Garcia Garcia[2], Holly E. A. Sutherland[2], and Peter C.-H. Cheng[2]

[1] University of Cambridge, UK
{daniel.raggi,aaron.stockdill,mateja.jamnik}@cl.cam.ac.uk

[2] University of Sussex, UK
{g.garcia-garcia, h.sutherland, p.c.h.cheng}@sussex.ac.uk

Abstract. Choosing effective representations for a problem and for the person solving it has benefits, including the ability or inability to solve it. We previously devised a novel framework consisting of a language to describe representations and computational methods to analyse them in terms of their formal and cognitive properties. In this paper we demonstrate the application of this framework to a variety of notations including natural languages, formal languages, and diagrams. We show how our framework, and the analysis of representations that it enables, gives us insight into how and why we can select representations which are appropriate for both the task and the user.

Keywords: Knowledge representation · Reasoning · Intelligent systems.

1 Introduction

A given problem can be represented in a variety of ways, and the choice of representation determines whether it can be solved at all, as well as influences the performance of problem solvers—either helping or hindering them. It is up to the problem solver to represent the problem appropriately before solving it. Likewise, in a tutoring setting, it is up to the tutor to select an effective representation for a given audience. But *what is an effective representation,* and *how can we tell it apart from a bad representation?*

The quality of a representation is a confluence of many factors [2, 4], including whether it expresses the necessary information, makes this accessible, enables useful inferences, and reduces the search space for the problem solver. Many of these factors are user-dependent; some representations may be ideal for expert users, but not for novices, and vice-versa. The ultimate goal of our research is to understand what makes an effective representation, computationally model this analysis, and thus enable the automation of representation selection.

In previous work we introduced a language [9] for encoding the properties of representational systems, in addition to correspondences between them [11]. The purpose was to calculate an informational measure, which, given a problem, estimates the likelihood that the important information for this problem can be

* This work was supported by the EPSRC grants EP/R030650/1, EP/R030642/1, EP/T019034/1 and EP/T019603/1. We thank Gem Stapleton for her useful comments.

expressed in any given representational system. In subsequent work we implemented algorithms for computing cognitive measures of representations. Calculating these measures requires a richer and more structured language, which we incorporated in our framework. Specifically, we introduced *attributes* which allow us to encode structural information and more detailed descriptions of the representation's components. In this paper we illustrate through examples how to use our language (including its new additions) to describe representations in the framework. Moreover, we demonstrate how the framework can be used for representation selection based on informational and cognitive measures. Our work provides novel and general computational methods for assessing and comparing sentential and diagrammatic representations that are formal and informal, general and specialised; and could thus be used for making AI systems more human-like and adaptable to the user. An Appendix for this paper can be found at https://sites.google.com/site/myrep2rep/publications/dissecting.

2 How to describe representations?

The fundamental objects that our framework aims to describe are *representational systems* (RSs). For example, Arithmetic Algebra forms an RS in which expressions are constructed using tokens (e.g., $x, 0, +, =, \leq$) with some grammatical constraints (e.g., $\_ = \_$ needs to be filled with expressions of the same type), and its expressions can be manipulated according to some rules (e.g., $x + 0 \leq y$ can be rewritten as $x \leq y$). Moreover, our framework also describes concrete instances of representations, such as *problem* formulations. For instance, the problem in Arithmetic Algebra: *assuming* $0 < x$ *and* $x \cdot y = 0$*, prove* $y = 0$.

2.1 Representational systems and problems

We characterise a representational system (RS) by its formal **components**: its *tokens*, *expressions* (which we capture by *patterns*), *types*, *tactics*, and *laws*. A component can have *attributes*, specified as a record of features associated with it. We introduced these concepts (excluding attributes) elsewhere [9], so here we only provide a brief explanation: tokens are atomic symbols from which expressions are built. Patterns are abstractions of expressions; and their attributes encode structural information (e.g., how expressions can be nested in one another). Types are a grammatically meaningful classification of expressions (e.g., the type of $\pi + 4$ is real), tactics are the possible manipulations and inferences within the system (e.g., applying the modus ponens rule), and laws are the rules or units of knowledge that enable some manipulations and inferences to be made.

A representational system is a general tool for representing many things, but we are particularly interested in its use for representing *problems*. In this paper we demonstrate how (four) different representations of the same problem can be dissected and evaluated by the tools that our framework provides. We chose to focus our analysis primarily on two RSs. The first (*Bayesian*) is a standard formal notation for conditional probability, and the second (*PS diagrams*) is a novel diagrammatic notation for probability, which has been shown to improve students' problem solving and learning [3].

Problem (Lightbulbs). *There are two lightbulb manufacturers in town. One of them is known to produce defective lightbulbs 30% of the time, whereas for the other one the percentage is 80%. You do not know which one is which. You pick one to buy a lightbulb from, and it turns out to be defective. The same manufacturer gives you a replacement. What is the probability that this one is also defective?*

The problem is presented in English (*NL: Natural Language*), which we do not analyse here, but results for its informational and cognitive measures are shown in §3. An analysis of the NL formulation can be found in Appendix.

Representation 1 (Bayesian) *Denote the manufacturers as a and b. Let d_1 and d_2 be the events of the first and second lightbulbs being defective, respectively. Clearly, d_2 is conditionally independent of d_1 given the choice of manufacturer.*

$$\textit{Assume: } b = \bar{a} \tag{1}$$
$$\Pr(a) = \Pr(b) \tag{2}$$
$$\Pr(d_2 \mid x \cap d_1) = \Pr(d_2 \mid x) \textit{ for } x \in \{a, b\} \tag{3}$$
$$\Pr(d_1 \mid a) = \Pr(d_2 \mid a) = 0.3 \tag{4}$$
$$\Pr(d_1 \mid b) = \Pr(d_2 \mid b) = 0.8 \tag{5}$$
$$\textit{Calculate: } \Pr(d_2 \mid d_1).$$

Some notable tokens here are Pr, $|$, $\cap$, a, b, d_1, d_2. Some features of these tokens can be encoded by attributes. For instance, we write

$$\text{token } a : \{\text{type} := \texttt{event},\ \text{occurrences} := 5\}$$

to indicate that a is a token with type `event`, and that it occurs 5 times in this specific representation. Moreover, we can assign more complex types, such as $\texttt{event} \times \texttt{event} \to \texttt{real}$ to Pr. In our framework this implies that there is a pattern, associated with the token Pr, encoded as follows:

$$\text{pattern } \text{patt}(\Pr) : \{\text{type} := \texttt{real},\ \text{holes} := [\texttt{event}^2],\ \text{tokens} := [\,\Pr, |, (,)\,]\}$$

Intuitively, patt(Pr) represents the expressions of the shape $\Pr(\_ \mid \_)$. The declaration above means: first, that these expressions have type `real`; second, that they are formed by plugging in two expressions of type `event` into the *holes*; and third, that they necessarily use each of the tokens Pr, $|$, (, and). In our implementation,[3] every pattern associated with a token (of nontrivial type) is generated automatically, such as for $\cap$, $=$, and $\in$.

Our framework can also encode inferential aspects (use of tactics and laws), which we illustrate by analysing the solution below (full solution in Appendix).

Solution (Bayesian) *Amongst other things, use the law of total probability (LTP), de Finetti's axiom of conditional probability (dF), Bayes' theorem (BT), arithmetic calculation (calc). For conciseness we show only a part of the solution:*

$$\begin{aligned}
\Pr(d_2 \mid d_1) &= \Pr(d_2 \cap a \mid d_1) + \Pr(d_2 \cap b \mid d_1) && \text{(by LTP, asm 1)} \\
&= \Pr(d_2 \mid a \cap d_1)\Pr(a \mid d_1) + \Pr(d_2 \mid b \cap d_1)\Pr(b \mid d_1) && \text{(by dF)} \\
&\ \ \vdots \\
&= \frac{0.3 \cdot 0.3 + 0.8 \cdot 0.8}{0.3 + 0.8} \approx 0.663 && \text{(by calc)}
\end{aligned}$$

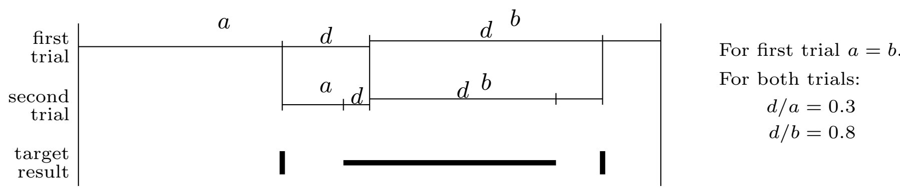
[3] https://github.com/rep2rep/robin

Every step in this solution can be characterised as an application of the tactic *rewrite*, or an arithmetic calculation. This, and more information (e.g., how many times each tactic is applied) can be encoded by attributes, as follows:

$$\begin{aligned}\text{tactic } \texttt{rewrite} : \{&\text{occurrences} := 18,\ \text{inference\_type} := \text{substitution},\\ &\text{law\_params} := 1,\ \text{pattern\_params} := 1\}\\ \text{tactic } \texttt{calculate} : \{&\text{occurrences} := 1,\ \text{inference\_type} := \text{calculation},\\ &\text{law\_params} := 0,\ \text{pattern\_params} := 1\}\end{aligned}$$

Foreshadowing what this means in terms of the cognitive cost of using this representation, calculation is in principle a more complex operation, but rewriting has a larger contribution to the breadth of the search space because it can be applied in many ways depending on the laws at hand.

Representation 2 (PS diagrams) *Below, labelled segments represent events and their lengths represent their probability. The ratio that needs to be calculated is that of the thicker line relative to the space between the thick delimiters.*



This representation has some important characteristics (segments, delimiters, proportions, etc.) that need to be captured in our description. Amongst others, some tokens are the horizontal segments (thin and thick), the vertical marks (thin and thick) and the vertical lines. Emergent components, such as a segment formed by two colinear segments, relations between components, or values thereof (e.g., length), can be expressed as patterns:

$$\begin{aligned}\text{pattern } \texttt{joint\_segments} &: \{\text{type} := \texttt{segment},\ \text{holes} := [\texttt{segment}^2],\ \ldots\}\\ \text{pattern } \texttt{aligned\_segments} &: \{\text{type} := \texttt{relationship},\ \text{holes} := [\texttt{segment}^2],\ \ldots\}\\ \text{pattern } \texttt{relative\_length} &: \{\text{type} := \texttt{real},\ \text{holes} := [\texttt{segment}^2],\ \ldots\}\end{aligned}$$

Patterns also allow us to represent emergent *gestalt* items. For instance, the 'segment' inbetween the 2 target delimiters can be encoded by:

$$\begin{aligned}\text{pattern } \texttt{segment\_from\_delimiters} : \{&\text{type} := \texttt{segment},\\ &\text{holes} := [\texttt{delimiter}^2],\ \ldots\}\end{aligned}$$

We proceed to analyse the inferential aspects of this representation by looking at a solution.

Solution (PS diagrams) *The length of the segments labelled* d *in the second trial must be* $0.3 \cdot 0.3 \cdot x$ *and* $0.8 \cdot 0.8 \cdot x$ *where* x *is the length of* a *(and* b*) in the first trial. Moreover, the length between the target delimiters must be* $0.3 \cdot x + 0.8 \cdot x$*. Thus, the desired ratio is* $\frac{0.3 \cdot 0.3 \cdot x + 0.8 \cdot 0.8 \cdot x}{0.3 \cdot x + 0.8 \cdot x}$*. This yields* ≈ 0.663*.*

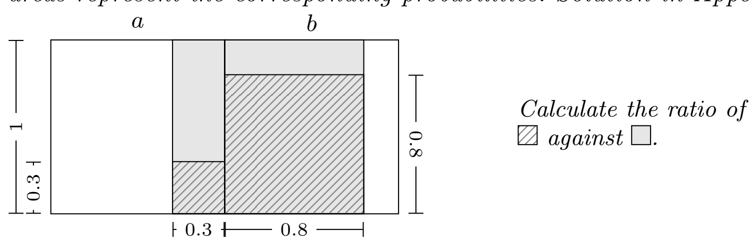
This solution is quite condensed because each inference relies on mere *observations* which are possible as an immediate consequence of having represented the assumptions [10]. Here we assume observations apply to patterns; e.g., observing the relative length is obtaining the real number that represents such relation. We can capture the notion of observation as a tactic:

$$\text{tactic } \texttt{observe} : \{\text{occurrences} := 10,\ \text{inference\_type} := \text{observation},\ \text{pattern\_params} := 1,\ \text{law\_params} := 1,\ \ldots\}$$

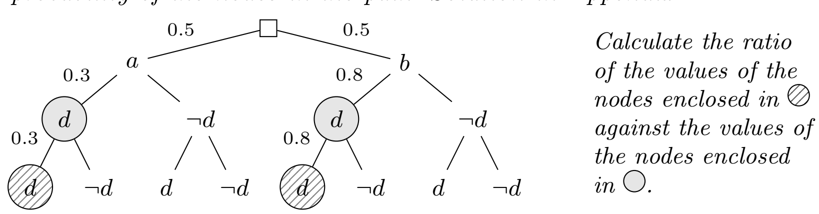
Finally, the sequence of observations leads to a ratio that the user still needs to calculate, so we need a calculation tactic similar to the Bayesian RS.

So far we analysed one sentential and one diagrammatic representation. We hope this demonstrates that our language is simple yet expressive. Below, we give alternative representations under consideration, but without analysis.

Representation 3 (Areas) *In the figure, regions represent events, and their relative areas represent the corresponding probabilities. Solution in Appendix.*



Representation 4 (Probability trees) *In the rooted tree below, the values of edges represent conditional probabilities and the values of the nodes represent the joint probability of the nodes in the path. Solution in Appendix.*



2.2 Writing RS and Q descriptions

The example RSs above demonstrate the expressiveness and intuitions behind the components of our framework. But, *what should the end result of analysing representations look like?*

An **RS description** is a collection of meaningful components of an RS: it must include tokens that typically appear in the instances of such an RS, and patterns, laws and tactics that are relevant for using such an RS. Similarly, a **Q description** (Q for *question*) is a collection of meaningful components of a problem representation in some specific RS that the question is posed in. Each component in a Q description must have an *importance* [9] value associated with it, encoding how informative this component is for finding a solution (defined in the interval between 0 for noise and 1 for maximal relevance; we use colours for discretised values). A longer discussion of importance can be found in Appendix.

See Figure 1 for two RS descriptions, and Figure 2 for a Q description (organised in a hierarchy of 4 importance values, where purple is the most important).

Bayesian	
types	event, real, formula, proof
tokens	$=$: {type := $\alpha \times \alpha \rightarrow$ `formula`}, Pr : {type := `event` $\times$ `event` $\rightarrow$ `real`, tokens := [$\mid$.(.)]}, $\cap$: {type := `event` $\times$ `event` $\rightarrow$ `event`}, Ω : {type := `event`}, ...
patterns	equality_chain : {type := `proof`, holes := [$\alpha^{O(\log n)}$], tokens := [=]}, ...
tactics	rewrite : {inference_type := subst, ...}, calculate : {inference_type := calc, ...}, lemma : {inference_type := match, ...}
laws	LTP, dF, BT, ...

PS diagrams	
types	segment, vertical_guide, delimiter, real
tokens	`$outcome_segment` : {type := `segment`}, `$target_delimiter` : {type := `delimiter`}, `$target_segment` : {type := `segment`}, ...
patterns	joint_segments : {type := `segment`, holes := [$\texttt{segment}^2$]}, relative_length : {type := `real`, holes := [$\texttt{segment}^2$]}, ...
tactics	observe : {inference_type := obs, ...}, calculate : {inference_type := calc, ...}
laws	MNR, EAS, LADJ, ...

Fig. 1. Snippets of Bayesian and PS diagrams RS descriptions. Note the prefix `$` to specify that this is a label for a non-unicode token.

Lightbulbs in NL	
answer type	ratio
types	number,event
tokens	probability : {type := N, occurrences := 1}, % : {type := `number` $\rightarrow$ `ratio`, occurrences := 2}
patterns	sequential_events : {type := `relationship`, holes := [$\texttt{event}^2$], ...}, conditionally_independent_events : {type := `relationship`, holes := [$\texttt{event}^2$], ...}
tokens	30 : {type := `number`, occurrences := 1}, 80 : {type := `number`, occurrences := 1}, percentage : {type := N, occurrences := 1}
tokens	lightbulb : {type := N, occurrences := 2}, defective : {type := N, occurrences := 3},...
patterns	SfromNPandVP : {type := S, holes := [NP, VP],... }...

Fig. 2. A section of the Q description of the Lightbulbs problem in NL.

3 Evaluating representations

We can use RS and Q descriptions to compute important measures: *informational suitability* (presented in [9]), and *cognitive cost*.

The Informational Suitability (IS) of an RS, r, given a problem q is the sum of the strengths of analogical *correspondences* [11] between components that match the source q and the target r, modulated by the importance of said components:

$$\mathrm{IS}(q, r) = \sum_{\langle a,b,s\rangle \in C} s \cdot \mathrm{importance}_q(a). \tag{6}$$

It computes the extent to which an RS can express all the relevant parts of the problem at hand. For the Lightbulb problem with 5 candidate RSs the results are shown below:

RS	Bayes	PS diag.	Areas	Pr-trees	NL
score	7.9	7.5	7.2	6.6	6.3

The Cognitive Cost encodes the RS's processing cost to the user, and is calculated by computing a set of properties of the representation, all of which can be estimated by values computed from Q descriptions (out of the scope of this paper). These properties are based on established cognitive science concepts [1, 6–8, 12, 13], presented schematically in Figure 3.

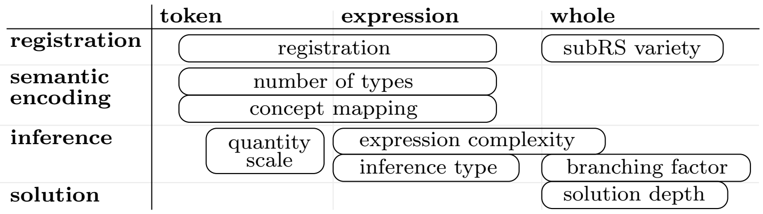


Fig. 3. Cognitive properties organised according to granularity (columns) and cognitive process level (rows).

Each of the properties is associated with a cognitive process, and thus a cognitive cost. Moreover, the user is modelled by their expertise [5], which is accounted for in two ways: by flattening importance (to model that a novel user cannot distinguish between important and unimportant properties), and by inflating the cost of higher-level cognitive processes. Given a Q description for a problem q, the costs for each cognitive property p and user u are calculated, normalised, and weighted by an expertise factor $c_p(u)$. The values for all p are summed to obtain a total cost.[4]

$$\mathrm{Cost}(q,u) = \sum_p c_p(u) \cdot \mathrm{norm}_p(\mathrm{cost}_p(q,u)). \tag{7}$$

See the rankings of RSs according to their estimated cognitive cost for the Lightbulb problem, for three different users:[5]

	Bayesian			PS diag			Areas			Pr trees			NL		
expert | avg. | novice	1	2	4	4	1	1	2	3	2	5	4	3	3	5	5

The main contributing factor to the differences in rankings between novices and experts comes from the cognitive costs associated with high granularity properties, for example: branching factor and solution depth. Because the weights associated with these costs scale with expertise, a representation like the Bayesian representation is penalised more heavily here for novices than for experts (dropping from first to fourth). Conversely, we see the Areas and PS diagrams representations have relatively low values in these cognitive costs, and as such are less penalised for novice users.

4 Conclusion and future work

We demonstrated our computational framework for analysing representations by explicitly constructing RS and Q descriptions for a particular problem and a number of candidate alternative representations. These descriptions serve as input to compute informational and cognitive measures of the suitability and the cost of using a representation by a particular user. Q and RS descriptions need

[4] These calculations rely on parameters whose values we gave provisionally based on the literature, but which need to be tuned based on empirical data.
[5] The costs, broken down per cognitive property, can be found in appendix.

to be built by an expert analyst; this includes decomposing into components, assigning importance and attributes to components, setting up correspondences with their strengths, and tuning the parameters of cognitive properties based on empirical data. Current and future work involve operationalising the process of obtaining descriptions and carrying out user studies for parameter tuning.

The generality of our approach makes our framework potentially useful for a variety of endeavours: from multi-representational tutoring systems, to user-sensitive interactive theorem provers. The ability to consider the user allows the framework to be deployed across many domains varying in their level of specialisation. The framework's descriptions are computation-friendly, creating an opportunity for diverse, diagrammatic representations to be evaluated and subsequently implemented in domains where sentential representations dominate.

References

1. J. Anderson. Spanning seven orders of magnitude: A challenge for cognitive modeling. *Cognitive Science*, 26(1):85–112, 2002.
2. A. Blackwell et al. Cognitive dimensions of notations: Design tools for cognitive technology. In *Cognitive Technology*, pp. 325–341. Springer, 2001.
3. P. Cheng. Probably good diagrams for learning: representational epistemic recodification of probability theory. *Topics in Cognitive Science*, 3(3):475–498, 2011.
4. P. Cheng. What constitutes an effective representation? In *Diagrams 2016*, pp. 17–31. Springer, 2016.
5. M. Chi. *The nature of expertise.* Lawrence Erlbaum Associates, Inc, 1988.
6. J. Larkin and H. Simon. Why a diagram is (sometimes) worth ten thousand words. *Cognitive Science*, 11(1):65–100, 1987.
7. D. Moody. The "physics" of notations: toward a scientific basis for constructing visual notations in software engineering. *IEEE Transactions on Software Engineering*, 35(6):756–779, 2009.
8. A. Newell. *Unified theories of cognition.* Harvard University Press, 1990.
9. D. Raggi et al. Inspection and selection of representations. In *CICM*, pp. 227–242. Springer, 2019.
10. G. Stapleton, M. Jamnik, and A. Shimojima. What makes an effective representation of information: a formal account of observational advantages. *Logic, Language and Information*, 26(2):143–177, 2017.
11. A. Stockdill et al. Correspondence-based analogies for choosing problem representations in mathematics and computing education. In *2020 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, 2020. In press.
12. M. Van Someren et al. *Learning with Multiple Representations. Advances in Learning and Instruction Series.* ERIC, 1998.
13. J. Zhang. A representational analysis of numeration systems. *Cognition*, 57(3):271–295, 1995.

Appendix

Dissection of Natural Language formulation of the Lightbulb problem

The tokens of NL are its words (in addition to numbers, full stops, delimiters, etc.). In this case, there are 51 unique tokens, some of which occur more than once. For instance, the word 'defective' occurs three times. The types of these tokens, insofar as NL is concerned are nouns (N), verbs, (V), adjectives (ADJ), etc. Thus, in our framework we write:

$$\text{token } \texttt{defective} : \{\text{type} := \texttt{ADJ},\ \text{occurrences} := 3\}$$

The items specified inside curly brackets (in this case *type* and *occurrences*) are called *attributes*. Note that the type is a global attribute, in the sense that 'defective' will be an adjective in any use of the NL RS, whereas occurrences is local, in the sense that *defective* happens to occur three times in this problem, but this is not the case in other uses of the RS.

Moreover, the grammar of NL is described by rules, for example, a sentence (S) can be formed from a noun phrase (NP) and a verb phrase (VP). In this example, six of the sentences are constructed from this one rule. Within our framework, we describe these rules by the use of *patterns* as follows:

$$\text{pattern } \texttt{SfromNPandVP} : \{\text{type} := \texttt{S},\ \text{holes} := [\texttt{NP}, \texttt{VP}],\ \text{occurrences} := 6\}$$

Note that the pattern type is declared as the resulting type of any expression that was constructed using the pattern. Holes are uninstantiated subexpressions.

We can discern more complex patterns too, for instance, sequential events (e.g., *the first lightbulb you buy is defective* and *you get the replacement*). Intuitively, this pattern is characterised by the relation of two *events*, so we can write:

$$\text{pattern } \texttt{sequential} : \{\text{type} := \texttt{relationship},\ \text{holes} := [\texttt{event}^2],\ \text{occurrences} := 3\}$$

We use the type `relationship` to denote patterns which refer to facts. We write $\texttt{event}^2$ to specify that two entities of that type are needed to satisfy the pattern. But where does the type `event` come from, if the entities of NL have only typical grammatical types (S, V, NP, etc.)? For this problem, the reader is expected to interpret some sentences as events, which we encode by a pattern `eventFromS`:

$$\text{pattern } \texttt{eventFromS} : \{\text{type} := \texttt{event},\ \text{holes} := [\texttt{S}],\ \text{occurrences} := 3\}.$$

Solutions in a variety of RSs

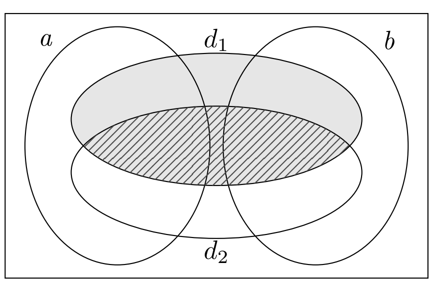
Solution (Bayesian).

$$
\begin{aligned}
\Pr(d_2 \mid d_1) &= \Pr(d_2 \cap a \mid d_1) + \Pr(d_2 \cap b \mid d_1) && \text{(by LTP, asm 1)}\\
&= \Pr(d_2 \mid a \cap d_1)\Pr(a \mid d_1) + \Pr(d_2 \mid b \cap d_1)\Pr(b \mid d_1) && \text{(by dF)}\\
&= \Pr(d_2 \mid a)\Pr(a \mid d_1) + \Pr(d_2 \mid b)\Pr(b \mid d_1) && \text{(by asm 3)}\\
&= \Pr(d_2 \mid a)\frac{\Pr(d_1 \mid a)\Pr(a)}{\Pr(d_1)} + \Pr(d_2 \mid b)\frac{\Pr(d_1 \mid b)\Pr(b)}{\Pr(d_1)} && \text{(by BT)}\\
&= \frac{\Pr(d_2 \mid a)\Pr(d_1 \mid a)\Pr(a) + \Pr(d_2 \mid b)\Pr(d_1 \mid b)\Pr(b)}{\Pr(d_1 \cap a) + \Pr(d_1 \cap b)} && \text{(by dist, LTP)}\\
&= \frac{\Pr(d_2 \mid a)\Pr(d_1 \mid a)\Pr(a) + \Pr(d_2 \mid b)\Pr(d_1 \mid b)\Pr(b)}{\Pr(d_1 \mid a)\Pr(a) + \Pr(d_1 \mid b)\Pr(b)} && \text{(by dF)}\\
&= \frac{0.3 \cdot 0.3 \cdot \Pr(a) + 0.8 \cdot 0.8 \cdot \Pr(b)}{0.3 \cdot \Pr(a) + 0.8 \cdot \Pr(b)} && \text{(by asms 4,5)}\\
&= \frac{0.3 \cdot 0.3 + 0.8 \cdot 0.8}{0.3 + 0.8} \approx 0.663 && \text{(by calc)}
\end{aligned}
$$

Solution (Areas). *The patterned region must have an area of* $0.3^2 + 0.8^2$*, and the shaded region must have an area of* $0.3 + 0.8$*. Then, the desired ratio is* $\frac{0.3^2+0.8^2}{0.3+0.8} \approx 0.663$*. The description of the inferential aspects of this representation is similar to that of PS diagrams.*

Solution (Probability trees). *According to the law of path multiplication, the values of the nodes enclosed in are* $0.5 \cdot 0.3^2$ *and* $0.5 \cdot 0.8^2$*, and the values of the nodes enclosed in are* $0.5 \cdot 0.3$ *and* $0.5 \cdot 0.8$*. Thus, the desired ratio is* $\frac{0.3^2+0.8^2}{0.3+0.8} \approx 0.663$.

Representation 5 (Euler + cardinality algebra) *We chose Euler diagrams RS for its notable inadequacy for this problem, as Euler diagrams do not specify the size of the sets in question. Thus, we need to supplement it with cardinality algebra to have something informationally comparable to the other RSs presented so far.*



Assume: $d_1 \subseteq a \cup b$

$$
\begin{aligned}
&d_2 \subseteq a \cup b\\
&|a| = |b|\\
&|d_1 \cap a| = 0.3 \cdot |a|\\
&|d_1 \cap b| = 0.8 \cdot |b|\\
&|d_2 \cap a \cap d_1| = 0.3 \cdot |a \cap d_1|\\
&|d_2 \cap b \cap d_1| = 0.8 \cdot |b \cap d_1|
\end{aligned}
$$

Calculate: $\dfrac{|d_2 \cap d_1|}{|d_1|}$

Importance

When a problem is presented, it is clear that some components used in the representation are more important than others, and some may even be irrelevant (noise). We use the notion of *importance* to express this. Clearly, the importance is strictly relevant to the task, so we express it only when describing a problem (such as the *Lightbulbs* problem above) in a particular representation (for example, in Natural Language). Importance is defined as a function from the components to the interval ranging between 0 and 1, where 0 is noise and 1 denotes a maximally informationally relevant property. For example, the token *Pr* for the Bayesian representation of the *Lightbulbs* problem is important. Assigning importance is like finding good heuristics – in our framework, the domain expert who is setting up our framework for their deployment assigns these values. In the future, we will explore if there is a principled approach to assigning these values, and if these importance parameters can be generated automatically by analysing a sufficiently large set of problems.

In practice, we distinguish between discrete classes of importance as follows: a component is *essential* (importance(x) = 1) if replacing it with something else modifies the nature of the problem. A component is *instrumental* ($0.5 \geq$ importance(x) < 1) if a solution most certainly needs the component. A component is *relevant* ($0 <$ importance(x) < 0.5) if it is useful for understanding the problem but it could be replaced for something else without affecting the structure of the solution, and we say that it is *circumstantial* or *noise* (importance(x) = 0) if removing it does not affect the solution at all.

Thus, for Q descriptions, components are classified according to their importance.

Cognitive costs for six RSs, broken down by cognitive properties

Table 1. Estimated costs, with weighted normalisation function η_p for each of the cognitive properties for *average* user ($u = 0.5$). tr = token registration, er = expression registration, tt = number of token types, et = number of expression types, cm = concept-mapping, qs = quantity scale, ec = expression complexity, it = inference type, sr = subRS variety, bf = branching factor, sd = solution depth.

	$\mathrm{norm}_p(x)$	NL	Bayesian	Pr trees	Euler+	PS diag	Areas
tr	$0.5 \cdot \eta_{\mathrm{tr}}(x)$	0	35.4	45.6	50	30.6	18.6
er	$0.5 \cdot \eta_{\mathrm{er}}(x)$	11.9	7.4	2.6	20.1	0	50
tt	$1 \cdot \eta_{\mathrm{tt}}(x)$	0	100	78.9	68.4	52.6	50
et	$1 \cdot \eta_{\mathrm{et}}(x)$	51.3	0	79.5	38.5	48.7	100
qs	$1 \cdot \eta_{\mathrm{qs}}(x)$	26	0	94	42.7	100	53.1
cm	$2 \cdot \eta_{\mathrm{cm}}(x)$	109	0	200	147	173	183
ec	$2 \cdot \eta_{\mathrm{ec}}(x)$	137	0	86.9	200	116	99.1
it	$2 \cdot \eta_{\mathrm{it}}(x)$	200	59.7	0	25.6	2.3	20.4
sr	$4 \cdot \eta_{\mathrm{sr}}(x)$	0	0	0	400	0	0
bf	$4 \cdot \eta_{\mathrm{bf}}(x)$	271	250	198	400	0	257
sd	$4 \cdot \eta_{\mathrm{sd}}(x)$	271	310	129	400	205	0
total		98	69.3	83.2	163	66.3	75.6
rank		5	2	4	6	1	3