How to Model Uncertainty in Rules – A Generic Framework

Introduction

Suppose I tell you I have a yellow object and ask you to identify it.
It could be (among many other things)

1. A banana
2. A lemon
3. A submarine (if you are old enough to remember the Beatles)

But it's considerably less likely that it's a submarine
But if also tell you it's long then it's less likely to be a lemon.
And if I say that it's small, this adds more weight to it being a banana
But at no point can we conclusively say what it is.
So how can we model rules that deal with this uncertainty

A Simple Approach

One approach to this is to construct a rule sheet like this

Conditions	1	2	3	4	5	6	7
What color is it?	'yellow'	'yellow'	'yellow'	'red'	'green'	'yellow'	-
What shape is it?	'long'	'oval'	'long'	'round'	'oval'	-	'oval'
How big is it?	'small'	'small'	'large'	'small'	'small'	-	-
How heavy is it?	'light'	'light'	'heavy'	-	'light'	-	-
What does it taste like?	'sweet'	'sour'	'salty'	'sweet'	'sour'	-	-
Is it edible?	T	T	F	T	T	-	-
Actions							
Post Message(s)							
It could be a ...	'banana'	'lemon'	'submarine'	'raspberry'	'lime'		
With certainty...	0.7	0.8	0.6	0.5	0.4		

However, there are some problems with this approach:

1. There may be a huge number of combinations to deal with if the rules are very precise (like 1-5)
2. If you make the rules more general (like 6) then what is the conclusion? It could be many things.

The Basic Idea

The object will have a number of attributes.

Each attribute tells us something about the object (e.g. color=yellow, size=small, shape=long) and this information will increase or decrease the likelihood of the object being a particular thing (e.g. submarine or banana)

So instead of determining the one thing that the object might be, we need to be able to maintain a collection of possible things (hypotheses) that the object might be (along with a measure of certainty that it is that thing)

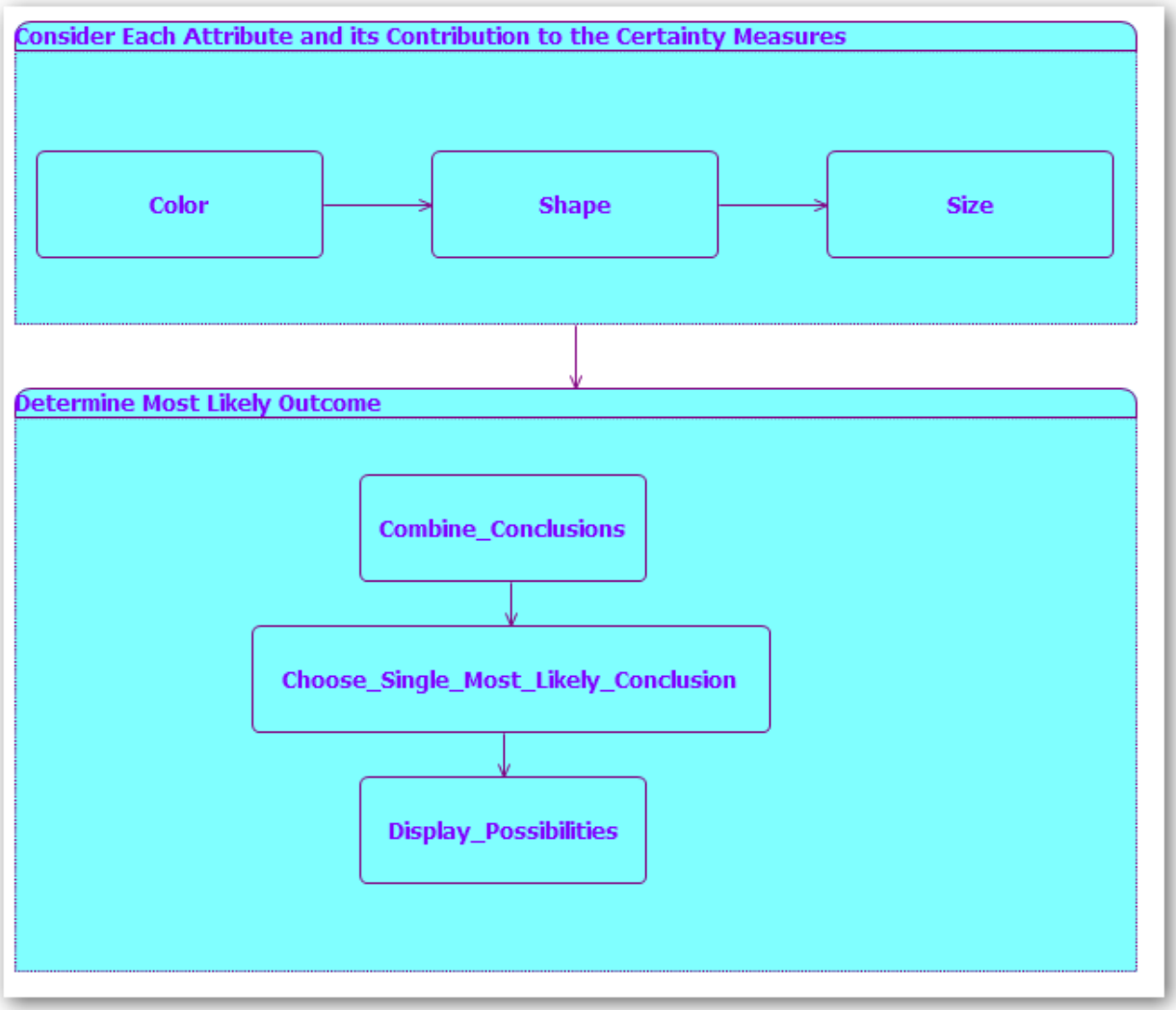
Then as other rules process the various attributes we can adjust our certainty.

At the end of this process we can select the most likely hypothesis.

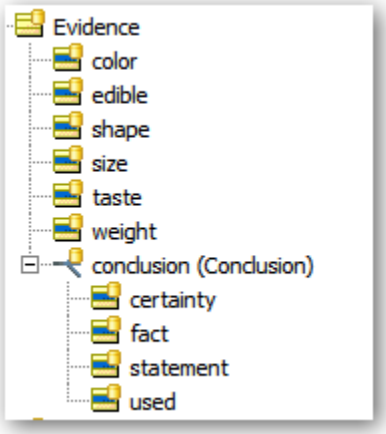
A Simple Example

If color is red then it's definitely NOT a banana
If color is yellow then it might be a banana with a certainty of .5 or a submarine with a certainty of 0.0001
But if we also know that its size is large then it's more likely to be a submarine than a banana.

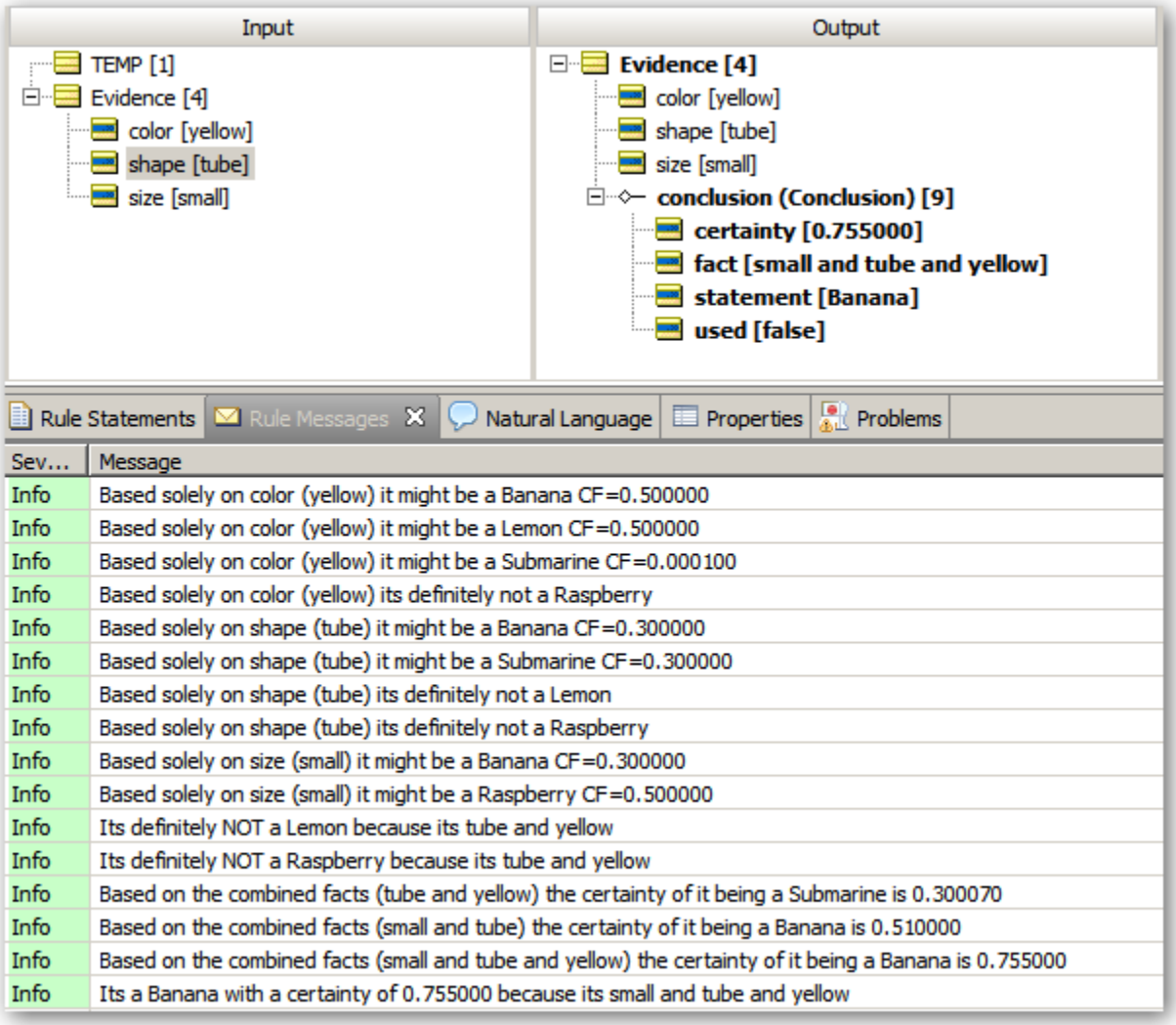
The Rule Flow



The Vocabulary



Sample Test Case



Sev...	Message
Info	Based solely on color (yellow) it might be a Banana CF=0.500000
Info	Based solely on color (yellow) it might be a Lemon CF=0.500000
Info	Based solely on color (yellow) it might be a Submarine CF=0.000100
Info	Based solely on color (yellow) its definitely not a Raspberry
Info	Based solely on shape (tube) it might be a Banana CF=0.300000
Info	Based solely on shape (tube) it might be a Submarine CF=0.300000
Info	Based solely on shape (tube) its definitely not a Lemon
Info	Based solely on shape (tube) its definitely not a Raspberry
Info	Based solely on size (small) it might be a Banana CF=0.300000
Info	Based solely on size (small) it might be a Raspberry CF=0.500000
Info	Its definitely NOT a Lemon because its tube and yellow
Info	Its definitely NOT a Raspberry because its tube and yellow
Info	Based on the combined facts (tube and yellow) the certainty of it being a Submarine is 0.300070
Info	Based on the combined facts (small and tube) the certainty of it being a Banana is 0.510000
Info	Based on the combined facts (small and tube and yellow) the certainty of it being a Banana is 0.755000
Info	Its a Banana with a certainty of 0.755000 because its small and tube and yellow

The Rules

Assigning certainty to the individual attributes considered independently

NOTE: These rules and the certainty factors are for illustration only

Shape

Conditions	1	2	3	4	5
What shape is it?	'tube'	'tube'	'oval'	'tube'	'tube'
Actions					
Post Message(s)					
It might be a	'Banana'	'Submarine'	'Lemon'	'Lemon'	'Raspberry'
With this certainty	0.3	0.3	0.4	-1	-1
Add a conclusion to the evidence	✓	✓	✓	✓	✓

Size

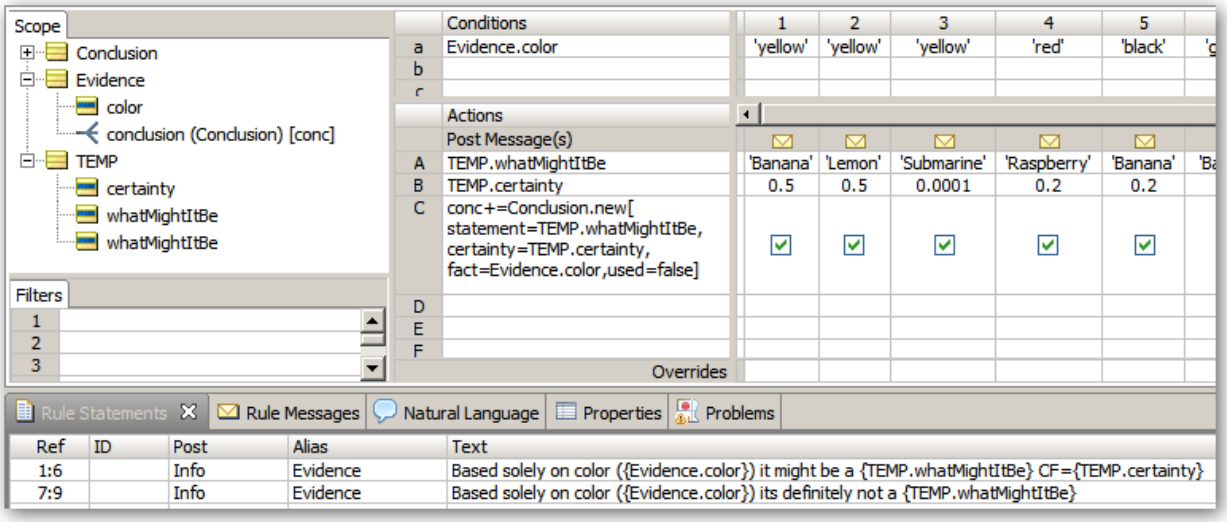
Conditions	1	2	3	4	5
What is its size?	'small'	'large'	'small'	'large'	'large'
Actions					
Post Message(s)					
It might be a	'Banana'	'Submarine'	'Raspberry'	'Banana'	'Raspberry'
With this certainty	0.3	0.3	0.5	-1.0	-1.0
Add a conclusion to the evidence	✓	✓	✓	✓	✓

Color

Conditions	1	2	3	4	5	6	7	8	9
What color is it?	'yellow'	'yellow'	'yellow'	'red'	'black'	'green'	'black'	'yellow'	'red'
Actions									
Post Message(s)									
It might be a	'Banana'	'Lemon'	'Submarine'	'Raspberry'	'Banana'	'Banana'	'Raspberry'	'Raspberry'	'Banana'
With certainty	0.5	0.5	0.0001	0.2	0.2	0.2	-1.0	-1.0	-1.0
Add a conclusion to the evidence	✓	✓	✓	✓	✓	✓	✓	✓	✓

If there were more factors then we'd just add more rule sheets.

Implementation



Incidentally this rule sheet illustrates a technique for getting around the limitation that Corticon only allows a single "cellValue" reference in a row. Instead of using a single cellValue we make use of the TEMP object (in action rows A and B) to collect up all the values that will get set in action row C. This idea can be extended to any number of values.

Combining Conclusions and Updating the Certainty

Once the initial conclusions have been made based on the attributes considered separately we then have to combine the attributes and the corresponding certainty factors.

We cannot simply add all the certainty factors that support a give outcome (otherwise we might end up with a certainty greater than 1).

So we use the following three rules for combining certainty factors:

Rule 1 The rules for adding two positive certainty factors:

CFcombine (CFa CFb) = CFa + CFb(1 - Cfa)

I.e., reduce the influence of the second certainty factor by the remaining uncertainty of the first, and add the result to the certainty of the first.

For example if the two certainties are 0.6 and 0.8 then the combined certainty factor is:

.92 = .6 + .8(1 - .6)

It does not matter which factor you start with first:

.8 + .6(1 - .8) = .6 + .8(1 - .6) = .92

Both sequences produce the same result.

Rule 2 The rule for adding two negative certainties :

Treat the two factors as positive and negate the result

CFcombine (CFe CFf) = -(CFcombine (-CFe -CFf))

Rule 3 The rule for adding positive and negative certainty factors :

CFcombine (CFg CFh) = (CFg + CFh) / (1 - min{|CFg|, |CFn|})

Thus if your certainty for an instance is 0.88 and your certainty factor against it is 0.90, the result is:

-.17 = (.88 - -.90) / (1 - min(.88, .90))

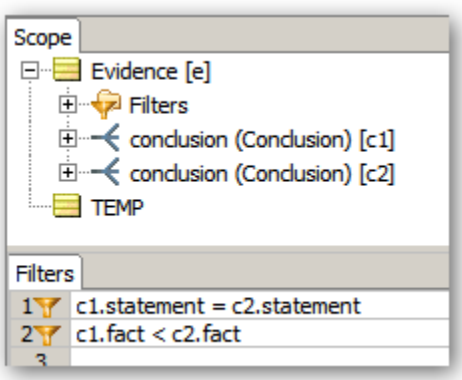
= -.02 / .12

I.e. take the difference, and then multiply that value by the reciprocal of the smallest remaining uncertainty.

These three rules provide an interval scale for certainty factors.
You will note that you cannot say that a certainty factor of 0.8 is twice the certainty of 0.4; the rules of this metric only involve those of addition and subtraction.
Here are those rule expressed in Corticon:

Conditions	1	2	3	4	5	6
c1.certainty >= 0	-	-	T	F	T	F
c2.certainty >= 0	-	-	T	F	F	T
c1.used	F	F	F	F	F	F
c2.used	F	F	F	F	F	F
c1.certainty = -1.0	T	-	F	F	F	F
c2.certainty = -1.0	-	T	F	F	F	F
Actions						
Post Message(s)			✓	✓		
c1.certainty = c1.certainty.absVal + c2.certainty.absVal * (1 - c1.certainty.absVal)			✓	✓		
c1.certainty = (c1.certainty + c2.certainty) / (1 - c1.certainty.absVal.min(c2.certainty.absVal))					✓	✓
c1.fact = c1.fact.concat(' and ').concat(c2.fact)	✓	✓	✓	✓	✓	✓
c1.certainty = -c1.certainty				✓		
c1.certainty = -1	✓	✓				
c2.used = true	✓	✓	✓	✓	✓	✓

Here is the Scope section for this rule sheet:

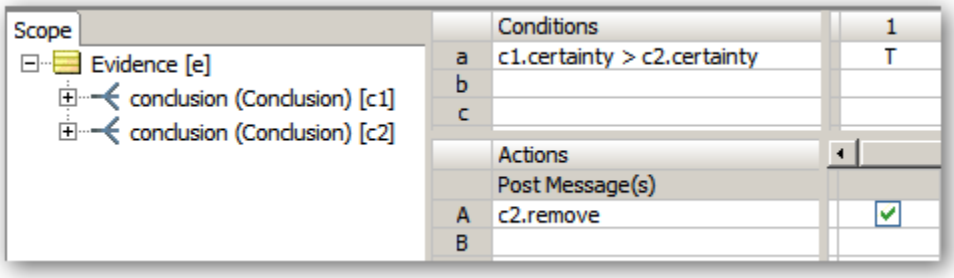


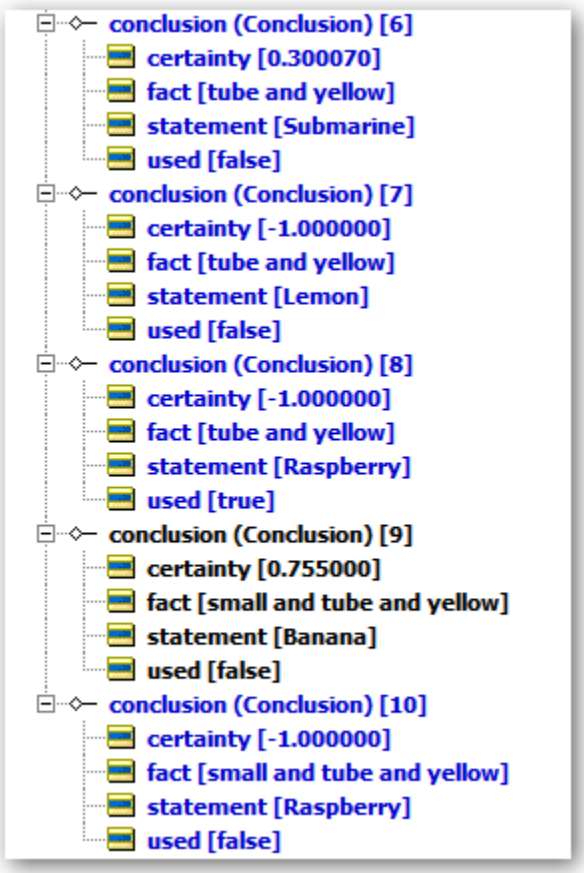
This ensures that we only combine certainty factors that support (or contradict) a given hypothesis.

Note that once we have a certainty factor of -1 (certain the evidence does not support the conclusion) then no further positive evidence can change that conclusion.

Determining the Conclusion with the highest Certainty

This can be done by comparing pairs of conclusions and removing the one with the smaller certainty:



If this rule sheet is disabled in the rule flow then you will be able to see all of the conclusions that were reached with their respective certainty factors:

Eg