Abstract. This paper shows the initial stages of development, from first principles, of a formal logic to characterise and then explore issues in a broadly defined idea of “Veracity”.

A Logic for Veracity

Steve Reeves

Department of Software Engineering, University of Waikato
Private Bag 3105, Hamilton, 3240, New Zealand

1 Introduction

When a piece of information is put out into the world it gets subjected to many attempts, both accidental and deliberate, to degrade it or tamper with it. When we are dealing with precious information, that is information which has value (cultural, monetary, scientific etc.), then having assurance that the information has stayed constant is vital. When that information is not kept hidden or otherwise protected then this becomes a very hard problem. It may even be insoluble.

Veracity seems to be a term that is widely used, but it is also hard to pin-down its meaning. In this paper I shall take it to mean, reflecting the concerns in the previous paragraph, that we have an assurance that the information has stayed constant. So, we say *a piece of information has veracity* when we can check that it has not changed.

Even though etymologically we might expect *truth* to have some role in veracity we avoid this. The main reason is that truth seems to require either reference to some authority (and we want our information to survive in an authority-free world: more on this later), or a belief in some objective and unchanging and always accessible reality against which we can always successfully measure our information and decide on its truth. This line of definition drives us towards accepting (perhaps implicitly, since we tend not to think about such things in everyday life) an idealised Platonist reality of some sort. This leads to all sorts of well rehearsed problems (we go with Dummett's analysis still on this [2]). And, of course, even if you are not a Platonist, the requirement that the reality against which you measure your information is always accessible (leaving aside decidability problems etc.) is sometimes precisely the problem; if the source has actually been obscured or lost then it is no longer accessible and truth cannot be decided. As we will see, raising the Platonist spectre does suggest an alternative.

1.1 Aims

My view, of course, is that there's a logical basis, and since we want to formalise this in the project in order to both pin-down and explore the idea of veracity, this seems the only sensible place to start.

There's a long, deep, rich heritage to truth and trust in many settings and many of them are formal, and very complicated and subtle. I want to start from scratch, not in order to just do something different, but in order to be able treat well, but lightly, those parts of veracity which can adequately be treated that way for our purposes (so, trust will be so treated). And then other parts (demonstrability, truth) will be looked at in more depth simply because they have not (as far as a literature search can show) been treated, in the setting of being a part of veracity, very much at all.

I am, therefore, not going to work through a literature review[1]. I hope that what I say below will make clear that classical approaches will not work, and trust only needs a light-touch, rather unsubtle and instrumental treatment.

[1] Work on another paper with Stephen Cranefield will cover some of this ground, anyhow.

To cut to the chase: I will look at intuitionistic logic since it seems to be clearly what's needed, as I argue below. But I'll get there by showing what does not work (pretty clearly), like classical logic or anything based on it (none of the–classical–modal logics work, for example, because of their classical basis, not because I do not like modal formalisms!). The key to seeing this is that all those classical (and classical-including) logics lose information, which is precisely what a formalisation of veracity, as a starting point, must not do, of course. Intuitionistic logic does not lose info. So, *obviously* it is the place to start.

The steps:

- We will, as the project does, take Veracity to comprise in: authenticity, truth, trust, demonstrability/verifiability;
- Try to pin down and then explore in a logical setting what this means;
- Attempt to formalise as much of veracity as we can in order to understand the way it works better.

1.2 Atomic veracity

Some statements have a sort of *immediate* veracity, in the sense that they are newly minted by me (or you) and have not passed through any other hands and have not been in any way combined with other information, so we are immediately assured that the information has not changed. The checking of this is a trivial, indeed empty, act.

Consider a couple of examples in more detail:

1. A bar code that we have ourselves just printed and associated with a physical object might be an example in a production chain: this act might generate the information that *this* bar code is stuck to *this* object that was produced by *this* person, at *this* time and *this* place, has *these* characteristics (composition, mass, etc.). We might say the information attests that"this bar code really does identify this object";
2. Or considering cultural objects, it might be an audio recording of a person giving their whakapapa together with its meta-data that we have just ourselves recorded and catalogued. Here the information is attesting to the association between the meta-data and the audio data.

These cases in some sense wear their veracity on their sleeve: it is immediate, we have "a piece of veracity", the information that *this* piece of data correctly describes *this* object since I, at just this moment, made the association. This is our *atomic veracity*. The claim or statement cannot be further analysed in terms of asking whose hands it has passed through, how it has be modified or added to since none of this has ever happened to it.

We might say (using logical terminology) that the piece of verifying information is a *witness, proof, testimony, piece of evidence* to the act of association. It is this that we want to be able to objectify and then track. This track will be what we look to when someone says "how do we know that this bar code correctly identifies this object?". Note that the information itself may be made up of many pieces of other information, or may have taken work to compile; but the information witnesses, is evidence for, an atomic claim. So, the witness may be complex, but the claim is atomic.

We might want to view this evidential information in more detail though. This will not be in the sense of more detail on the actual veracity claim itself, because this witness w, say, is already formed. But it might come with the information of who p, where l, when t, how m etc. In this case, the witness might not be an atomic name, a constant, but an "atomic" term. I.e. we might view a witness either as the atomic name w or the atomic term $w(p, l, t, m)$. So a witness, piece

of evidence, might contain a lot of information, but from the point of view of the logic it is not further analysable. Of course, as we build up non-atomic claims the witnessing information will correspondingly become both buildable and analysable in the logic.

How the data about it "sticks" to the artefact (the bar code on the car part, the meta-data to the whakapapa audio file) is not what we are concerned about here. It is another technological problem that is being worked on and is outside our scope. So, we are assuming it is possible and has been, or will be, done[2].

1.3 Other methods

Our idea of checking for assurance of veracity is different from the distributed ledger technology (DLT) way of doing it (e.g. through use of a blockchain). There the veracity isn't through checking but by making it impossible (or highly unlikely) that the information has been changed once it is put out into the world.[3]

2 Considering logics

2.1 Formalisation

We let letters like A, B, etc. stand for a claim of veracity, which is a proposition that is true when the veracity claimed is appropriately witnessed, upheld by data, by a person's statement, by direct knowledge, evidence, somehow, that the thing is what we say it is, came from where we say it came from, was grown as we say it was grown.... etc. etc.

Then a *judgement* $a \in A$ is the veracity judgement, statement[4] A has witness a. A judgement like this is upheld, or perhaps we might say that A has veracity because it is witnessed by a, when this judgement appears as the conclusion of a proof tree constructed according to the rules that follow.

There is a special veracity claim $\bot$ which has no witnesses, i.e. it is the claim that never has veracity, and a judgement that makes a claim about it can never be upheld.

This leads to our first proof rule:

$$\frac{a \in \bot}{a \in A}\,\bot^{-}$$

This rule says that if you, in the course of your reasoning, somehow have shown that the claim $\bot$ that can never have veracity does in fact have it, then you can show that *anything* has veracity. We call this rule $\bot^{-}$ for "$\bot$ elimination".

Other rules would be:

$$\frac{a \in A \quad b \in B}{(a, b) \in A \wedge B}\,\wedge^{+}$$

$$\frac{(a, b) \in A \wedge B}{a \in A}\,\wedge^{-} 1$$

[2] This might be wishful thinking, but it has such big stakes for such large companies that I think it's OK to assume it will happen one day. Whether it does or not, though, veracity is still an interesting idea to try to reason about.

[3] In the fuller version of this paper we will look at previous work on intuitionistic logic that we're drawing on: Martin-Löf [3, 4]; my work on logic from the past, in particular work with Douglas Bridges [1] too as background.

[4] Later when thinking semantically we might view A as the set of all its witnesses.

$$\frac{(a, b) \in A \wedge B}{b \in B} \wedge^{-} 2$$

Here we are formalising the idea that if two veracity claims A and B are witnessed then the combined claim that A together with B has veracity is also witnessed, and that witness we choose to denote by the pairing of the component witnesses.

Note that this is a simple use of the the idea also of information being preserved around claims and their witnesses even when they are composed together.

One immediate place where this information preservation becomes perhaps a little unfamiliar is when we try to think about what saying "we have claims A and B and we know that they each have a witness, so we know that one or the other has one: that is, a claim of A or B is witnessed". We might choose to formalise this by saying

$$\frac{a \in A \quad b \in B}{a \in A \vee B}$$

The point here is that (first) this rule has exactly the same premises as the one above, and avoiding such points of choice amongst rules is generally (for coherence) a good thing. But more importantly (at the formalisation level) is that we have lost information here. The conclusion does not record which of the alternatives we have relied on to reach it: did we justify the claim of one or the other because of the first witness, or the second?

Righting these two points means doing something like

$$\frac{a \in A}{i(a) \in A \vee B} \vee^{+} 1$$

$$\frac{b \in B}{j(b) \in A \vee B} \vee^{+} 2$$

So, if we have a witness to a claim of A then we certainly have a witness to a claim of either A or B, and we "tag" the witness in the conclusion so that we do not lose the information about which claim the claim of one or the other relies on.

Now consider the case where we know that a certain witness c upholds the claim that $A \vee B$. What can we deduce, if anything, from this?

First note that our two introduction rules mean that a witness to a claim like this must in fact have a tag since tags are introduced by the only rule that could have allowed us to deduce the claim of $A \vee B$. So, we have a case analysis to do: if the witness to this composite claim is tagged with i then we know it is A that we relied on and similarly with j and B. This preservation of all the information allows us to dismantle the composite claim:

$$\frac{i(a) \in A \vee B}{a \in A} \vee^{-} 1$$

$$\frac{j(b) \in A \vee B}{b \in B} \vee^{-} 1$$

(In fact, as we will see in the fuller paper, these rules need to be more general, but this gives the idea, I hope. We will I think need more complex ways of writing things. Cf Martin-Löf's need for the idea of canonical form, and computation (equality) rules to get to canonical.)

Finally, we denote the fact that a judgement $c \in C$ has been demonstrated to hold without assumptions, i.e. it is the conclusion of a proof tree of uses of these sorts of rules, by $\vdash c \in C$.

So think of this as saying that the judgement that c is a witness to C has been demonstrated by the rules of the logic. Later we will use this same turnstile notation to allow assumptions to appear (on the left).

Imagine that by assuming that claim A has veracity, i.e. that the judgement $x \in A$ has been shown for some arbitrary witness x, we can show that claim B has veracity, i.e. we can show $b \in B$[5].

Denote this state of affairs by *a claim that depends on an assumption*:

$$x \in A \vdash b \in B$$

Thinking about a typical logic, introduce an *implication claim* to reflect this, i.e. to discharge the assumption, so the claim becomes

$$A \Rightarrow B$$

but what would a witness to *this* claim plausibly look like?

Given any witness x to the claim A then it is possible to *construct* a witness b for the claim B. That is, there is a function which given any witness to A will compute a witness to B, so

$$\lambda(x)b \in A \Rightarrow B$$

The witness to an implicative claim like $A \Rightarrow B$ should be a function that takes a witness to the claim A and turns it into a witness for the claim B.

In general, this allows us to build a function that, given a whole set of basic veracity claims and their witnesses (the assumptions), builds for us a witness for a complex veracity claim. This function can then be read as a process to be followed which, given starting veracity claims, will assure that a complex veracity claim can be successfully and correctly made.

Implication us to define negation in terms of $\bot$: $\neg\ A$ is $A \Rightarrow \bot$. A witness to a claim of $\neg\ A$ takes a witness to A and give us a witness to $\bot$. But $\bot$ has no witness, so a witness to A is not possible, as expected by our informal understanding of saying a claim has no witnesses.

The requirement that to justify a disjunction of claims it has to be demonstrated which of the claims were justified before (which is the role that the tags on the witnesses are playing in the rules) means that, for example, the claim $A \vee \neg\ A$ is also not justifiable without saying which claim is witnessed: $A \vee \neg\ A$ doesn't survive the question: yes, but can you show, whatever A is, the witness that assures the veracity of the claim here?

And the view that witnesses to implications are functions leads us in the same direction...to the thought that this is reinterpreting intuitionistic logic

The argument so far is that the logic work above covers the verifiability (checking a proof is easy) and truth aspects of veracity. What is not yet settled is the trust aspect (the authenticity is left for now–yet to have any ideas on how it might be treated, or even what it is), and once we start to think about trust, we think about people and the relationships between them.

3 More actors

The section above works well when one person is collecting and making veracity claims. It is a one-person logic because we never mention who is making claims, so we cannot tell how many people might be, so we can only correctly assume it is one person from the form of the rules. In other words, there are no rules for combining or tracking veracity claims made by several actors.

[5] Here b is a term which may contain x free. $(x)b$ is a term with the free xs in b bound.

One way to perhaps tackle this is to add a name (of an actor) to each justified judgement. So, if actors k and l from a set Act have made claims then we might have two judgements $a^k \in A$ and $b^L \in B$, that is actor k has made claim A with witness a, and similarly for l, b and B.

This now adds a second dimension to our logic above. The first dimension dealt with one actor, so we can think of all the judgements before as being abbreviations (because there's only one actor k) of judgements of the form $a^k \in A$, so we left the k out because it never varied. Now the second dimension is around how actors become incorporated into the logic.

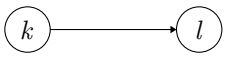
3.1 Relating actors

Having introduced more than one actor we now need to think about how, from a veracity point of view, they can be related.

Keeping to the idea that we think of simple cases to guide us rather than trying to do everything we might wish all at once, the question: what relationship between actors is a useful one (there will be others) to consider? Fundamentally, surely, is one of trust: does this actor trust that actor? Once we know who trusts who we can plausibly expect things like k trusting l means that any judgement that l has accepted allows k accept that judgement. So, roughly, we would say that $\vdash a^l \in A$ leads to $\vdash a^k \in A$ if k trusts l. If we denote the trust relation by $T \subseteq Act \times Act$ then k trusts l will be kTl we propose a rule

$$\frac{a^l \in A \quad kTl}{a^k \in A}\ trust\ T$$

and we can picture the relation as



This can be generalised to a more realistic situation, where there are various relations of trust between actors, not just T, by making clear we are parameterising the demonstration of judgements with the particular trust relationship in play and say

$$a^l \in A, kTl \vdash_T a^k \in A$$

i.e. assuming $a^l \in A$ and kTl hold for some trust relationship T, we have demonstrated $a^k \in A$.

(This is really just another way of giving the above rule, but makes it clearer (perhaps) that things are dependent on the trust relationship being used.)

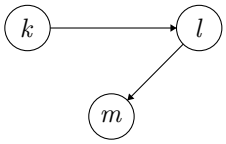
3.2 Trust relations

We can explore, even with this simple basis, how veracity works.

This is a derived rule from the simple previous one

$$a^m \in A, kTl, lTm \vdash_T a^k \in A$$

because, given T as



$$\dfrac{\dfrac{a^m \in A \quad lTm}{a^l \in A}\ trust\ T \qquad kTl}{a^k \in A}\ trust\ T$$

Given this example we might ask: can *any* binary relation between actors be a trust one? No; it surely needs to be at least reflexive and a certainly not symmetric: we trust ourselves, and if we trust someone does it follow that they should trust us? But I would hesitate to say a trust relation requires more properties.

Note that the proof above seems to show that trust is also transitive: it turns out to be a property of our simple rule. Does that call the simple rule into question, since it a stretch to accept that if I trust someone, and they trust someone else, then I should trust that someone else.

Well, I make the point that trust here is "100% trust" which explains this rule and how transitivity emerges in this pointwise way. I will return to this below.

Another derivable rule which seems to be a good thing:

if two people see veracity in two different things and one trusts the other then the first person believes the conjunction

$$a^k \in A, kTl, b^l \in B \vdash_T (a, b)^l \in A \wedge B$$

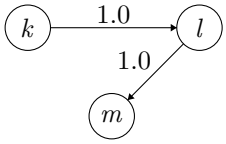
3.3 Degrees of trust

This brings the final augmentation, that we need *degrees of trust* to make things work. We write

$$a^k_{0.5} \in A$$

for k believes with strength 0.5 that a supports the claim A (and we drop the subscript in the case it's 1.0).

Then the apparent transitivity above only works if $kT_{1.0}l$ and $lT_{1.0}m$, i.e. k trusts l completely, and the same for l and m, i.e.



and that makes the apparent transitivity look reasonable

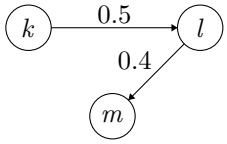
So, we recast the *trust* T rule as

$$\dfrac{kT_x l \quad a^l_y \in A}{a^k_{x.y} \in A}\ trust\ T$$

If instead $kT_{0.5}l$ and $lT_{0.4}m$ then I would say $kT_{0.2}m$ and the proof above supports this, rewritten as

$$\dfrac{kT_{0.5}l \qquad \dfrac{lT_{0.4}m \quad a^m \in A}{a^l_{0.4} \in A}\ trust\ T}{a^k_{0.2} \in A}\ trust\ T$$

i.e. if $a^m_{1.0} \in A$ and $kT_{0.5}l$ and $lT_{0.4}m$



then $a^k_{0.2} \in A$

We can also allow terms here with the variables remaining...interesting, and to be explored in the fuller paper.

4 Notes for further thought, and discussion, and data

4.1 Combining different trust relations

Most interesting, and hard, and probably not agreed on?

In particular really need some write-ups of the use cases so as to ground this.

Also industrial examples, as well as things like the work on my SfTI seed project with Āhau.

There's the question of relationships between trust relations (subset, disjoint...).

There's the other harder question of how the incompatible demonstrations of different actors interact. If the actors don't trust each other then it's clear...no merging.

If they do trust (in some direction) then (i) how do we spot incompatibility?; and (ii) how do we deal with it, if we wish to.

(i) might be formalised as combining the proof of two claims (or just two demonstrations) derives $\perp$;

(ii) That's a hard and interesting question...

4.2 Recording and dealing with disputed veracity stories

What happens if we have more than one trail of verification for a claim? Can it happen formally? Hope the previous section can deal with this in a clear way. A benefit of a simple formalisation!

Two types of statement perhaps. One for veracity and one for resolution of differences?

4.3 Authenticity

This is missing. But one thought is that once a proof tree is constructed, the open assumptions are places to look (in general) for problems in authenticity since they are the places where a judgement "hits" the real world. Authenticity is about a judgement being genuine.

4.4 Proof construction and checking

Work has started in using Isabelle to implement a proof checker for the veracity logic.

References

1. Douglas Bridges and Steve Reeves. Constructive Mathematics in Theory and Programming Practice. *Philosophia Mathematica*, 7(1):65–104, 1999.
2. Michael Dummett. *Elements of Intuitionism*. Clarendon Press, second edition edition, 2000.
3. P. Martin-Löf. *Intuitionistic Type Theory*. Bibliopolis, Naples, 1984.
4. P. Martin-Löf. Constructive Mathematics and Computer Programming. In C.A.R. Hoare and J.C. Shepherdson, editors, *Mathematical Logic and Programming Languages*. Prentice Hall International, Englewood Cliffs, N.J., 1985.