Features of Logics

The aim of ‘seeing how the logic works’ has become central to analytic philosophy. Since any theory can be expressed as a set of sentences, which can be subjected to rigorous mathematical analysis, this suggests that logic is close to the heart of our understanding of the world. No one thinks that is all there is to philosophy, but it is an intriguing frontier of modern thought. One can study the details of proofs, characterising successful rules and valid steps, but there are also features that appear on a large scale. Much of this is studied in Model Theory, but logical languages and the theories they express have other features that are worth grasping. Philosophers are typically interested in issues such as how to reason about properties, but two other factors that impinge on the topic are the commitment of mathematicians to many vast levels of infinity (with appropriate logic and set theory), and the theoretical limits of what can be implemented on a machine (which tends to emphasise syntax rather than semantics).

The most important aspect of a logical language is **expressibility**. For example, propositional logic can express nothing about predicates, and predicate calculus can express nothing about necessity. New logics are usually created to extend expressibility. Predicate calculus is sufficiently accepted to be labelled as ‘classical’, but it comes under pressure because of its expressive limitations when handling infinities. For example, it exhibits **compactness**, which means that if an infinite set of formulas (a theory, perhaps) implies some further formula, then in a compact logic the new formula is implied by a finite subset of the formulas. This means that compact logics are graspable by the human intellect, but that the logic ‘runs out of steam’ in the infinite, because adding new formulas to the infinite set makes no difference to what they can prove. The induction axiom of Peano Arithmetic requires infinities, so mathematicians cannot be wholly content with classical logic, even though many philosophers prefer it.

Whether a theory is **consistent** is best thought of as a syntactic property, meaning that the language can never prove both some proposition and its negation. It can also be expressed semantically, if a theory has no model in which all the formulas are true. A theory is said to be ‘negation-consistent’ if every proposition or its negation can actually be proved, which is thus the strongest form of consistency. In a formal system inconsistency is a disaster, and will eventually make any computer program crash, but it is interesting that in real life we can live with inconsistency, if it lies so deep as to be almost never encountered, or is obvious enough to be easily bypassed. An extreme view suggests that reality itself is inconsistent. A surprising result in the theory of logic was a proof that no syntactic system (just involving rules) can prove its own consistency, though for most logics this becomes possible if you add a semantics (involving truth).

Once a syntactic system has acquired a semantics, we can study the match between the two. **Soundness** is usually taken to be minimum mark of respectability for a logic, because it says that its proof system works – that is, every proof preserves truth. This can be proved, but only by using a logic which is assumed to be sound! If every truth in the system can be proved, then it exhibits **completeness**, which is an attractive feature always found in slightly restricted systems like first-order logic, but completeness is sometimes sacrificed to achieve greater expressivity. If necessary, completeness can be achieved by adjusting the semantics, or expanding the language. When logics are sound and complete, the syntactic proofs match the semantic proofs, so proofs can either use rules, or models and counterexamples. A system is ‘negation-complete’ if for every sentence it can show either its truth or its falsehood. A system is ‘strongly complete’ if adding another sentence must produce inconsistency. A system is ‘omega(ω)-complete’ if all countable truths are derivable, but not the universal quantifier (which implies higher infinities). First-order logic is both complete and compact.

Ancient geometry offers the first attempt at the **axiomatisation** of a system. The aim is to find a short list of clear truths, from which (using the rules of the language) all of the truths of the system can be deduced in easily understood steps. The traditional approach is to pick out the main ingredients of the system, and then state a self-evident basic fact about their nature, with the definitions being refined after trial and error with the deductions. Modern approaches are less confident about ‘self-evidence’, and start from a working system and work back to simple axioms, which may be true or fallible or just arbitrary. Geometry, arithmetic, first-order logic and set theory have all been treated in this way, until hopes of achieving full success were dashed, first by tinkering with the geometry axioms, and then proving that the truths of arithmetic exceed what can be proved from syntactic axioms. Set theory axioms are now a trade-off between what seems sensible, and what facilitates useful proofs, with the option to add a new axiom always available.

A modern focus for discussions of logic is the limits of what can be computed. What may be dealt with by infinite models or axioms cannot be handled by a machine with only finite storage space and running time available. Compactness is obviously an attractive feature for this task. The **enumerability** of a system also becomes of interest, meaning whether the ingredients of a system can be counted, by matching them with the natural numbers. The trickiest new issue raised by computing was **decidability**, which asks whether a procedure or proof will halt after a finite or manageable number of steps. For example, positive claims are usually quicker to prove than negative ones (such as the existence or non-existence of some legendary beast). If a system is decidable then all set memberships can be settled in models of the system. The problem is approached through **recursion** theory, studying how the steps depend on each other. Classical logic and first-order arithmetic turned out to be undecidable.

When considering the link between formal logic and everyday reasoning, the feature of **montonicity** becomes important. A logic is monotonic if ‘once it is proved it stays proved’. If you prove something from a set of formulas, the addition of further formulas can make no difference. This addition is known as ‘thinning’, because you dilute the left-hand side of the proof. The best known instance of non-monotonic reasoning is induction, which is acquiring general truths from repeated evidence (though fresh evidence can always change your view). Artificial intelligence seems to need a non-monotonic logic, which is precise enough to implement on a machine, yet flexible enough to respond to new information.