5.E: the ***PhilosophyIdeas*** overview of

Structures of Logic

In order to understand human reason, we want to get at the logic which underlies rational discussion in ordinary language. It is proposed that sentences have a **logical form**, and that if we identify this form, it can be expressed symbolically, and the structures of sentences can be assembled using a set of clearly defined ingredients which follow exact rules. The most basic components are the symbols for the logical connectives, for variables, for constants, and for predicates. More complex forms add functions, relations, and theories.

In ordinary language the ‘**connectives**’ are the words which relate other words together, rather than referring to features of reality. Some of them, like ‘so’, invoke a logical relationship, but others like ‘and’ and ‘or’ provide more neutral links. In formal logic both types are treated as logical, because we can also draw inferences from the more neutral relations. Classical logic focuses on the connectives which are **truth-functional**, meaning that together they fix the ‘truth-value’ (usually T or F) of the sentences they compose. The ordinary connective ‘because’ is not truth-functional, because you can’t know ‘p because q’ is true just by knowing whether p and q are true; ‘p and q’ and ‘p or q’, on the other hand, have their truth or falsehood settled by the truth-values of p and q. The truth-functional connectives have roles described by **truth-tables**, which fully map the T and F inputs and outputs for each one. The fact that ‘and’ and ‘but’ have the same truth-table shows that logic loses some of the nuances of ordinary language. Similarly, if I write ‘he came in *and* sat down’, the ‘and’ implies a chronology, but the logical ‘∧’ is timeless.

The main philosophical debate about the connectives is whether they have an intrinsic nature which we can attempt to define, or whether their nature wholly consists of the rules for their use, or their role in sentences. If they have intrinsic natures, then they seem to embody truths either about an eternal realm of ideas, or about general facts concerning the world. These views are no longer popular, and the focus is now on rules and roles. If the connectives are mainly syntactic in character, then the rules for their introduction and elimination from argument seem to wholly specify their nature. A criticism of this approach is that it doesn’t distinguish between silly rules and sensible rules. If the connectives are mainly semantic in character, then their meaning seems to reside in the truth-tables. Some accounts try to combine the two approaches.

The introduction rule for ‘and’ (‘∧’) says that if you have propositions p and q, you may combine them into a single proposition ‘p∧q’. We can see that the result has added something, because the probability of ‘p and q’ is usually lower than their separate probabilities (like winning the UK and EU lotteries on the same day). Similarly ‘p or q’ (p∨q) says more than its parts. ‘Not’ does more than ‘flip’ the truth values T and F, because words like ‘big’ and ‘small’ also have flipped values (so that ‘not’ seems to involve ‘denial’). The truth table for ‘and’ says it is only true when both components are true, and the table for ‘or’ says it is true if one or both components are true.

How many connectives are needed for a good logic? There seem to be eight or nine truth-functional connectives in ordinary language, but the number can be greatly reduced by inter-defining them, such as by writing ‘¬(¬p∧¬q)’, which means the same as ‘p∨q’, so that ‘∨’ could be eliminated. In fact just two connectives are sufficient (as long as one of them is ‘¬’). Logic is used to both to explain the logical form of sentences, and to prove things, and the first task likes lots of connectives, while proofs prefer very few, so there is no correct basic set of connectives.

Predicate Calculus has a domain of objects, and these are referred to in formulas by either constants or variables. A **constant** (usually ‘a’, ‘b’, ‘c’…) is a term which remains fixed throughout an argument, and picks out a single object, and thus works as a name. The **variables** (‘x’, ’y’, ‘z’…) are either ‘bound’ by a quantifier (‘all the xs’, or ‘some x’), or else they are ‘free’, taking any value in the domain. Variables resemble pronouns like ‘it’ and ‘that’ in ordinary language, and common nouns like ‘horse’ can be thought of as variables (taking some particular horse as the value). Some logics group variables, for separate domains, or allow them to refer to pluralities. There remain puzzles about whether we should understand variables by their references, their roles, or some intrinsic meaning.

Many languages have a subject-predicate form, which picks out an entity, and then ‘predicates’ something about it. Logics must handle this phenomenon, and usually letters (‘F’, ‘G’, ‘H’…) are assigned to the **predicates**. Logicians are inclined to say that predicates assign ‘properties’ to things, but this should be treated with care, since predicates pick out a huge array of attributes which are not normally accepted as properties. The status of what is picked out by predicates is controversial, and at the heart of many debates in metaphysics. One attempt to side-step the difficulties is to focus on the truth of the whole statement, rather than on some entity referred to by the predicate letter.

A **relation** is treated in logic as a predicate with two places, written ‘xRy’ or ‘Rxy’, meaning ‘x has the R-relation to y’. The semantics for this uses set theory, where the relation is an ordered pair <x,y>. More complex relations can be reduced to pairs, and the ordered pair can be reduced to nested sets of objects, so relations are expressed as groupings of objects in the domain. Thus ‘to the left of’ is all the pairs of objects where x is to the left of y. This is the normal ‘**extensional**’ understanding of the components of logic, which is expressed as groupings of objects to which they apply (and ignoring much of our ordinary understanding of such things).

Mathematicians build their subject around **functions**, and so they are central to mathematical logic. A function is a procedure with an output, and its hallmark is that it outputs a single value, no matter how many inputs there are. If a function (‘over’ a domain) produces a value (in a new set) for every member of the domain, it is a ‘**total**’ function; for a smaller input it is a ‘**partial**’ function (perhaps if the function is inapplicable to some members of the domain). In model theory we find different types of function (‘injective’, ‘surjective’, ‘bijective’), depending on inputs and outputs. Thus functions are also extensional, and logic expresses them as just relations between collections.

When a semantics based on set theory is added to predicate logic, the system develops into ‘model theory’. Each formula which is proved in the system is a ‘theorem’, and a ‘**theory**’ is a complete set of theorems, said to be ‘closed under consequence’. Every proposition in the system is either true, or its negation is true.