A user's guide to logic

January 3, 2015

1 Translating from natural language to formal logic

One of the most important skills we will learn in this course is to translate between formal logic and informal statements in a natural language, e.g., English. This is particularly important in that a programmer needs to be able to take an informal specification of a problem and come up with code that correctly solves the problem . A first step towards doing this is to translate the informal problem specification into a precise, mathematically stated specification. To do this requires the ability to capture the underlying logical structure of informal statements in formal mathematical terms. Another reason to learn this skill is because, in textbooks, technical papers, and code documentation, computer scientists communicate by alternatively using natural language (when writing for easiest reader comprehension) and formal mathematics (when writing to maximize precision). To read and write such documents requires the ability to "shift gears" between natural language and formal mathematics,

Instead of trying to understand the entire logic of a sentence holistically, it is usually better to capture the logical structure of a sentence from the "outside-in". First, identify key words and phrases in the sentence that determine its logical structure. The "outside" logical keyword is the one that applies to the entire sentence, rather than to just some part.

If the sentence is saying something is a general rule (true for all objects of a certain type) then the outermost logical structure is a $\forall$. If it is saying that an object with certain properties exists, then it is a $\exists$. The other logical key words are propositional connective phrases. Such connective phrases build complex sentences from sub-sentences called clauses. If there are several such connectives, the "outside"-most connective is the one whose removal most neatly divides the sentence into relevant and coherrent sub-sentences. For example, "If she gets home on time, then she will either eat an early dinner or watch television." has two connective phrases "if..then" and "either ..or" and three clauses, "she gets home on time", "she will eat an early dinner" "she will watch television". Note that because English abbreviates clauses that overlap, the last clause does not appear word for word in the sentence. If we remove the "either ..or" it divides up the sentence into "If she gets home on time, then she can eat an early dinner" and "she can watch television". But you can't piece together the truth of the whole statement from the truth of these sub-sentences. If we remove the "if then", the sub-sentences are "She gets home on time" and "she can either eat an early dinner or watch television". If we knew these facts, we could determine whether the original statement was true. So the outermost connective is the "if, then".

Once you've isolated the outermost connective, identify the corresponding propositional logic operation. The following table shows several common English connective phrases and their corresponding logical operation.

1. If the statement is a general rule, then the logical structure of the statement is $\forall$ (variable) (something is true of the variable.) However, before you translate the (something) that is true of the variable, think whether there are any *conditions* that must be satisfied for the rule to hold, and whether there are any *exceptions* to the rule. Conditions can be implicitly listed by adjectives, rather than explicitly stated.

   For example in the statement: "For any prime number p, and any $1 \leq x \leq p-1$, $x^{p-1}-1$ is divisible by p", x and p are universally quantified, but with two conditions: p is prime, and $1 \leq x \leq p$.

   In general, the full translation would be $\forall$ (variable) [if ((the variable meets the condition) and (the variable is not an exception)) then (the rule is true of the variable)].

   Here are some common constructions of universal statements:

   (a) All (something's) are (something else). Translation: $\forall x[(x$ is a something$) \rightarrow (x$ is a something else$)]$. e.g., All men are mortals $= \forall x[x$ is a man $\rightarrow x$ is a mortal$]$.

   (b) For all x, (something happens). Translation: substitute $\forall$ for the words "For all"

   (c) The only (somethings) are (exceptions). Rule: x is *not* a something. Exception: x is an exception. Translation: $\forall x, [(x$ is not an exception$) \rightarrow (x$ is not a something$)]$, or equivalently, $\forall x, [(x$ is a something$) \rightarrow (x$ is an exception$)]$. Example: The only factors of p are 1 and p. Translation: $\forall x, [(x \neq 1 \wedge x \neq p) \rightarrow (x$ is not a factor of $p)]$. or equivalently, $\forall x[(x$ is a factor of $p) \rightarrow (x = 1 \vee x = p)]$.

   (d) Let x be any (something). Then (something happens concerning x). or, For any (something) x, (something happens concerning x). Translation : $\forall x[(x$ is a something$)\rightarrow$ (something happens concerning x)]. Example: For any simple, undirected graph G, G has an even number of odd degree verticies. Translation: $\forall G \in \{graphs\}[(G$ is simple and undirected) $\rightarrow$ (G has an even number of odd degree vertices.)].

(e) It is never true that (something). or There are no (somethings). Rule: not something. Translation:$\forall x$(x is not a something.)

2. If the statement is saying that something with certain properties exists, the overall structure is $\exists x$. Note that the properties may be implicitly listed as adjectives, or explicitly given. Common forms for saying something exists are:

   (a) There is a (something) with (property). or For some (somethings), (property), or For at least one (something), (property). Translation: $\exists x$[(x is a something) $\wedge$(x has the property)]. Example: There is a prime number larger than 10, or Some prime numbers are larger than 10, or At least one prime number is larger than 10. Translation: $\exists x$[(x is prime) $\wedge(x > 10)$].

   (b) x can be written as (expression involving y). Translation: $\exists y[x =$ (expression involving$y$)] Example: A rational number x can be written as p/q for integers p, q. Translation: $\forall x$[(x is rational) $\rightarrow$ ($\exists p, q \in \mathbf{Z}[x = p/q]$)].

3. A sentence with a "not" or a contraction with "not" (e.g., "isn't") is the negation ($\neg$) of the sentence with the "not" removed.

4. The phrase "Neither p nor q" translates as $\neg p \wedge \neg q$.

5. Usually a sentence with an "or" or the phrase "either p or q" would be translated as $p \vee q$. Occasionally, it is clear from context that exactly one of the two is true, in which case it can be translated $p \oplus q$. For example, "I will wear a red or orange shirt today" excludes the possibility that I would wear both a red and an orange shirt. Some people hold that "either..or" automatically excludes the case that both, but I am not one of these people.

6. "Unless p, q" translates to $q \vee p$.

7. The phrase "p and q" is translanslated as $p \wedge q$.

8. The phrase "p, but q" is translanslated as $p \wedge q$. Although "but" has a different connotation from "and", to say "p, but q" is to claim that , somewhat suprisingly, both are true.

9. Similarly with other words that mean "but", such as "p; however, q" and "although p, q".

10. "If p then q" translates as $p \rightarrow q$.

11. “p suffices for q” translates as $p \to q$”
12. “p is necessary for q” translates as $q \to p$. Note that, unlike in English, $p \to q$ does not connote that p *causes* q. It means that “If I know p, then I can safely conclude q .” For example, “It is necessary for a seed to dry out for it to grow into a tree” . This translates as “If a seed grows into a tree, then it dried out (previously)” although the act of drying out causes growth, not vice versa. If you observe that the seed has grown into a tree, you can safely conclude it dried out.
13. “For p to happen, q must happen”. $p \to q$ See the above item for an explanation.

Finally, after you isolate and identify the outside connective phrase, see how removing the phrase breaks the sentence into clauses. If you introduced a quantifier, you need to substitute the variable name throughout what is left. Note that this may involve some modifications of the text as written. For example, in the sentence “She will either watch television or eat dinner”, when you remove the words “either” and “or”, you have to put the words “She will” into both sub-sentences. Analyze the clauses using the same method, or, in computer science lingo, *recursively.*

2 Formality vs. Informality in Proofs

What is a proof? A proof is an argument that should convince any mathematically trained reader who knows the definitions of the concepts involved. A mathematician being paid to poke holes in your argument should fail, no matter how brilliant she is. Here are some guidelines for recognizing when an informal proof is valid.

1. Address your proof not to myself or the TA, but to another undergraduate who is competent at mathematical reasoning but who has not solved this particular problem. Would such a person be able to understand what you have written? Would she be compelled to accept your conclusion by the argument?
2. In class we will occasionally omit some of the more tedious steps in a proof. You may also want to skip some of the more technical steps as being "obvious". However, a rule of thumb is that you should only omit steps because it would be too boring for both reader and writer to include them, not because you do not know how to fill in those details. When in doubt, put details in. This is particularly important while we're gettting comfortable with proofs, especially since we will frequently be proving very obvious statements as an exercise in the mechanics of proofs.

3. Steps in a proof need to do more than just give a true conclusion; they must be logically valid. In other words, they must be true in any conceivable circumstance. If the reasoning you want to use is true because of something about the particular problem, but not in general, then you need to think about exactly what it is about the particular circumstance that makes that reasoning correct, and state it explicitly.

3 Solow's backwards-forwards method summarized

This section is paraphrased from: How to Read and Do Proofs: An Introduction to Mathematical Thought Processes by Daniel Solow, and published by Wiley. If you want more details, or are uncomfortable with proofs, I strongly recommend this book.

Although there is no mechanical procedure for determing exactly how a proof should go, there are several rules (or at least rules-of-thumb) that determine the overall structure of the proof. It is at least possible to use these rules to clear away the mechanical aspects of a proof, and get down to the real issues. (Once you've reached the real issues, it is frequently and unfortunately necessary to *think hard.*) You should always carefully distinguish between what is *known* to be true at that point in the proof; and what the *goal* is at that point in the proof.

3.1 The Backwards Stage

In the backwards-forwards method, the prover first uses the logical structure of the *goal* to guide the formulation of a proof strategy, breaking it down into simpler *sub-goals*, She then further breaks down the sub-goals recursively. This is called the "backwards" stage by Solow, since you are working *backwards* from your goal.

Note that in the backwards stage, you are formulating a proof strategy, not starting to write down a proof. Keep notes on the steps you use in this stage *on a seperate piece of paper*. When you are finished with the proofs of the simple sub-goals, go back to your notes to see how this finishes the proof of the goal. The finished proof will have the steps *in reverse order* from how they are written in your notes. This is another reason this stage is called the "backwards" stage.

This is particularly confusing for beginners, because most students were taught to do the opposite in Algebra, using algebraic rules to "simplify" the goal. Algebraic rules, unlike logical rules, are almost all reversible, so if you can go forward from one equation to another by algebraic rules, you can also go back. Although this is picky, we am going to insist that such moves be written in the logically correct order, even if they are reversible. This is because when we do more complicated reasoning, writing things in the wrong order leads to "circular arguments", where you accidentally confuse the goal with the known, and then use it to prove itself!

The current goal is used to guide the proof strategy as follows. Note that only the "outside-most" logical operation in the goal needs to be looked at. The backwards method will strip away these operations from outermost to innermost, always making the sub-goal simpler.

1. If your goal is of the form $p \wedge q$, you need to prove p and then prove q, or maybe prove q first and then p. So make one of p, q your sub-goal, and then the other.

2. If your goal is of the form $p \vee q$, you need to prove one or the other. However, the tricky part is that usually, p will be true in some circumstances and q in others. Thus, it is not always possible to just prove p or just prove q.

   A mechantical way of handling $\vee$ is to convert $p \vee q$ into $\neg(p) \to q$, and then use the syllogism method from the next item.

   A way that requires more thinking is to use a *proof by cases.* You can legally use a proof by cases to prove any statement, but this proof structure seems particularly helpful for proving "or" statements.

   Think about when p is true, and when q is true. (They don't have to be mutually exclusive.) For example, say you know, as part of what you are given to work from or from a previous step in the proof, that $r \vee s$. By thinking about the problem, you decide that when r is true, so is p, and when s is true so is q.

   The general format for the proof by cases would be: |" Case 1: Assume r; (Proof where we add r to the "known" and our sub-goal is p.) Therefore, $p \vee q$.

   Case 2: Assume s; (Proof where we add s to the "known" and our sub-goal is q.) Therefore, $p \vee q$.

   Thus, in either case $p \vee q$ (Reason: refer to the line where we know or prove $r \vee s$, last statement of Case 1, last statement of Case 2; using the rule, "proof by cases") "

   Note that when you have a line in a proof where you make an *assumption*, you are starting a *self-contained sub-proof.* Statements proved in the sub-proof are only necessarily so *if the assumption is true.* Thus, you cannot later use a line in the part of a proof based on an assumption in the part of the proof not using the assumption.

3. If your goal is of the form $p \to q$, use a *syllogism* proof. Assume p. Now your sub-goal is q. If by making the assumption p, we reach the conclusion q, then "if p then q".

Note that when you have a line in a proof where you make an *assumption*, you are starting a *self-contained sub-proof.* Statements proved in the sub-proof are only necessarily so *if the assumption is true.* Thus, you cannot later use a line in the part of a proof based on an assumption in the part of the proof not using the assumption.

4. If your goal is $p \iff q$, then that's the same as $p \rightarrow q \wedge q \rightarrow p$. So you need to do two proofs, one $p \rightarrow q$, so assume p and prove q, the other $q \rightarrow p$, so in this part, assume q and prove sub-goal p.

5. If your goal is of the form $\neg p$, usually you should use a negation rule to convert your goal into an equivalent goal of another form. For example, say your goal is "It is not the case that both x is prime and y divides x.". This is equivalent, by De Morgan's Law, to "Either x is not prime or y does not divide x." Now your sub-goal is an "or", so use the rule for "or". Note this simplification and the rule used on your separate paper. After you prove the "or", you still need to include in your proof a line going from the "or" to the equivalent negation of the "and" justified by De Morgan's Law.

6. You can also use a "proof by contradiction" at any time. To use a proof by contradiction, *assume the negation of your goal*, adding it to your list of "known" facts, and attempt to get a contradiction. If the negation of your goal yields a contradiction, then your goal must be incorrect. The format for a proof by contradiction looks like: "Assume $\neg(goal)$ [Proof where we add $\neg(goal)$ to our known facts, and our sub-goal is to prove a contradiction, e.g., prove both r and $\neg(r)$] From this contradiction, the assumption $\neg(goal)$ must be incorrect, so $goal$. (Reason: cite the line where you got the contradiction, the line where you make the assumption, and use the rule "proof by contradiction".)"

   If you use a proof by contradiction, go immediately to the forward stage.

7. If your goal is of the form $\forall x p(x)$, you usually use Universal Generalization. The next line of the proof should be "Let a (or other new variable name) be an arbitrary (thing of the right type)." Now your sub-goal is $p(a)$. Once you prove this subgoal, you can use Universal Generalization to justify your conclusion that $\forall x p(x)$.

8. If your goal is of the form $\exists x p(x)$, you need to find a particular value c and prove $p(c)$. The heart of the argument is finding the right c, and usually you should start working from the "Known" and stop working from the "Goal" at this point. Once you find the right c, and prove $p(c)$, then you can conclude $\exists x p(x)$ using the rule of "Existence by Example".

9. If your goal is an algebraic equation, you can do algebraic simplifications to get your new goal. However, be careful when writing up the proof: Prove

the simplified version, then do the simplifications in REVERSE order to get your original goal from the simplified version.

10. If your goal has no obvious logical connectives, it might still be of one of the above forms because it involves concepts DEFINED using logical connectives. So replace all or some of the DEFINED symbols with their definitions. For example, if the goal is $A \subseteq B$, replace this with the defintion of one set being a subset of the other: $\forall x[x \in A \rightarrow x \in B]$. Now use the rule for statements which are of the form $\forall$, since this is the "outermost" logical symbol. Be sure to write up this in the proper order; you should first prove the translated goal, and then conclude your original goal in the next line, with the justification, "definition of (concept)".

3.2 The forward stage

Once you've simplified your goal as much as possible using the backwards method, start the proof by going *forwards* from what you know. Usually, you have to use every fact you know at least once, and you can start making deductions "stream of consciousness" style. Then go back and eliminate the part of the proof you don't actually need.

1. What is the "known"? You start knowing all premises (givens). Any line in the proof becomes "known" and remains known throughout the proof. The exception to this is if you "assume" something, as in a proof by syllogism, cases, or contradiction. Then the assumption becomes "known" during the sub-proof where you are making that assumption, but then reverts to not "known" after you are done proving that case. All lines in the sub-proof also are no longer "known" outside the sub-proof, since they were based on the assumption.

2. If you know a statement of the form $p \wedge q$, then you know p and you know q.

3. One way to use a known statement of the form $p \vee q$ is in a proof by cases. Assume p, and prove your goal. Then go back and assume q and prove your goal. Note that the two have to be seperate sub-proofs, since we can't use a conclusion based on an assumption outside the sub-proof for the assumption.

   Alternatively, if you later find out $\neg p$, then you can use the knowledge $p \vee q$ to conlude q.

4. If you know $p \rightarrow q$, then if you later prove p, you can conclude q using the rule "modus ponens". If you later prove $\neg q$, you can conclude $\neg(p)$ using the rule "modus tollens".

   Alternatively, convert it to $q \vee \neg p$ and use a proof by cases as in the previous item.

5. If what you know is of the form $\neg p$, it frequently helps to use negation rules to convert it to an equivalent statement of one of the other forms.
6. If what you know is of the form $p \iff q$, then you can substitute q for p throughout the proof. In particular, if you prove one, you can conclude the other, and if you prove the negation of one, you can conclude the negation of the other.
7. If what you know is of the form $\exists x p(x)$, you introduce a new variable, say a, by "Pick a so that $p(a)$. Now you know $p(a)$, but nothing else about a. This is called the "Rule of Choice" by me (and probably no one else.)
8. If what you know is of the form $\forall x p(x)$, and c is any variable or expression of the same "type" as x, you also know $p(c)$ by the rule of Universal Specification.
9. If what you know is an algebraic statement, you can do any of the normal algebraic simplifications or moves.
10. If what you know doesn't seem to be of any of the above types, it might still really be of one of the types because of DEFINED symbols or words. Translate some or all of the defined symbols using their definitions. For example if you know A is not a subset of B, then you know $\neg(\forall x[x \in A \rightarrow x \in B]$ which is logically equivalent to $\exists x[x \in B \wedge x \notin A]$, so you now should use the rule of choice.

4 Proofs by contradiction

It is never absolutely necessary to do a proof by contradiction, so if it makes you uncomfortable don't do it. What a proof by contradiction allows us to do is translate our goals into knowns, so that we can do the whole proof working from knowns, rather than switching between goals to knowns. If our goal is p, we assume $\neg p$. Now $\neg p$ is a known, and our goal is a contradiction. Note that the form of the goal switches when you negate it. If your goal was an "or", it will become a known that's an "and" by DeMorgan's Law and vice versa, and a goal that's a "for all" becomes a known that's an "exists" and vice versa. A goal that's an implication becomes a known that's the and of the hypothesis and the negation of the conclusion. "Assume that it's not true that $(p \rightarrow q)$" is the same as "Assume p, BUT $\neg q$". "But" and "and" have the same logical meaning, it's only their connotations that are different.

5 Informal proofs

: Most of the time, working mathematicians and computer science do not provide detailed step-by-step explanations of their proofs like I said to do above.

However, the same reasoning steps are in fact used. The difference is just that things are not completely spelled out: steps are combined or left to the reader to fill in. As we progress in this class, we will move from the very formal proof style above to more informal styles. However, you need to walk before you can run, so I would like you to practice the formal proofs at the beginning of the class. I think they will be a big help when we are trying to be less formal later.