Guide to Expressing Facts in a First-Order Language

Ernest Davis

September 28, 2015

There is no cookbook method for taking a fact expressed in natural language or any other form and expressing it in first-order logic. You have to think though the logical structure of what it is you want to say. However, the following list of suggestions and common errors may be helpful.

1 Syntax.

You must obey the syntactic restrictions of first-order logic. In particular, the arguments to functions and predicates must be terms. Predicates, Boolean operators, and quantifiers cannot appear with in the scope of a function or predicate. For instance, in a language where `Child(x,y)` and `Male(x)` are predicates, the sentences "Sam has a male child" cannot be expressed in the form

(1.1) $\texttt{Male(}\exists_{\texttt{x}}\ \texttt{Child(x,Sam))}$!!! WRONG !!!

The sentence "Mary and Ed are children of Anne" cannot be expressed in the form

(1.2) $\texttt{Child(Mary} \wedge \texttt{Ed, Anne).}$!!! WRONG !!!

(I have seen things like this, and worse, not only in students' homeworks but in published papers.)

The correct formulations of these are

(1.3) $\exists_{\texttt{x}}\ \texttt{Child(x,Sam)} \wedge \texttt{Male(x).}$
(1.4) $\texttt{Child(Mary,Anne)} \wedge \texttt{Child(Ed,Anne).}$

2 Restricted quantification.

Two common form of sentence is "All α's are β" and "Some α's are β." In mathematical notation, these are sometimes written using restricted quantification:

(2.1) $\forall_{\texttt{x}|\alpha(\texttt{x})}\ \beta(\texttt{x})$.
(2.2) $\exists_{\texttt{x}|\alpha(\texttt{x})}\ \beta(\texttt{x})$.

These translate into standard first-order logic, respectively, as

(2.3) $\forall_{\texttt{x}}\ \alpha(\texttt{x}) \Rightarrow \beta(\texttt{x})$.
(2.4) $\exists_{\texttt{x}}\ \alpha(\texttt{x}) \wedge \beta(\texttt{x})$.

For instance, the sentences "All crows are black" and "Some squirrels are black" are represented, respectively

(2.5) $\forall_{\mathtt{x}}$ `Crow(x)` $\Rightarrow$ `Black(x).`
(2.6) $\exists_{\mathtt{x}}$ `Squirrel(x)` $\wedge$ `Black(x).`

3 Two easily detected errors

If you have written a formula of the form $\forall \alpha(\mathtt{x}) \wedge \beta(\mathtt{x})$ then you have probably made a mistake. To check, translate it to the logically equivalent form $[\forall_{\mathtt{x}} \alpha(\mathtt{rxX})] \wedge [\forall_{\mathtt{y}} \beta(\mathtt{y})]$" and see if it still looks good. Both forms mean "Everything is both α and β" or equivalently "Everything is α and everything is β."

If you have written a formula of the form $\exists_{\mathtt{x}} \alpha(\mathtt{x}) \Rightarrow \beta(\mathtt{x})$ then it is 100 to 1 that you have made a mistake. Keep in mind that this is equivalent to $\exists_{\mathtt{x}} \neg \alpha(\mathtt{x}) \vee \beta(\mathtt{x})$, which is in turn equivalent to $[\exists_{\mathtt{x}} \neg \alpha(\mathtt{x})] \vee [\exists_{\mathtt{y}} \beta(\mathtt{y})]$; i.e. either there exists something that is not α or there exists something that is β. For instance the formula

(3.1) $\exists_{\mathtt{x}}$ `Crow(x)` $\Rightarrow$ `Black(x).` !!! WRONG !!!

means "Either [there exists something in the universe is not a crow] or [there exists something that is black]".

The only correct uses of the second form above I have seen have been where the variable `x` does not actually appear in α, the left hand side of the implication. For instance, the sentence "If Y is even then there exists X such that $Y = X + X$ can be correctly written

(3.2) $\forall_{\mathtt{y}}$ $\exists_{\mathtt{x}}$ `Even(x)` $\Rightarrow$ `x=y+y.`

But, unless there is some strong reason to wish all the quantifiers to be prenex, it is in my opinion much clearer to write quantifiers with the smallest possible scope, and therefore rewrite this sentence,

(3.3) $\forall_{\mathtt{x}}$ `Even(x)` $\Rightarrow$ $\exists_{\mathtt{y}}$ $\mathtt{x} = \mathtt{y} + \mathtt{y}$.

.

4 Chained implications.

In mathematical writing, it is common to say "α implies β implies γ", meaning that α implies β which implies γ, and sometimes to notate this $\alpha \Rightarrow \beta \Rightarrow \gamma$. Likewise, the statement that α, β and γ all have the same truth value is sometimes notates $\alpha \Leftrightarrow \beta \Leftrightarrow \gamma$. It is important to keep in mind that what the first of these really means is

(4.1) $[\alpha \Rightarrow \beta] \wedge [\beta \Rightarrow \gamma]$

and what the second really means is

(4.2) $[\alpha \Leftrightarrow \beta] \wedge [\beta \Leftrightarrow \gamma]$

For instance, “Squirrels are rodents, which are mammals,” is represented

(4.3) $\forall_x$ `[Squirrel(x)` $\Rightarrow$ `Rodent(x)]` $\wedge$ `[Rodent(x)` $\Rightarrow$ `Mammal(x)].`

In particular (4.1) is NOT equivalent to either

(4.4) $\alpha \Rightarrow (\beta \Rightarrow \gamma)$, nor to
(4.5) $(\alpha \Rightarrow \beta) \Rightarrow \gamma$,

and (4.2) is not equivalent to

(4.6) $\alpha \Leftrightarrow (\beta \Leftrightarrow \gamma)$, nor to
(4.7) $(\alpha \Leftrightarrow \beta) \Leftrightarrow \gamma$,

5 Don’t trust the English: Quantifiers.

Natural language were not designed by logicians, and a single English form can have very different logical meanings in different sentences.

For instance, it’s often the case that words like “some”, “something”, “someone”, and “a” correspond to an existential quantifiers. For example “Some squirrels are black,” or “Jack saw a squirrel” can be represented

(5.1) $\exists_x$ `Squirrel(x)` $\wedge$ `Black(x).`
(5.2) $\exists_x$ `Squirrel(x)` $\wedge$ `Saw(Jack,x).`

However, in the sentence “If someone is eighteen, then they are allowed to vote,” the meaning is that this implication is true of everyone. The representation is therefore

(5.3) $\forall_x$ `Geq(Age(x),Times(18,Year))` $\Rightarrow$ `Legal(Do(x,Vote)).`
NOT (5.4) $\exists_x$ `Geq(Age(x),Times(18,Year))` $\Rightarrow$ `Legal(Do(x,Vote)).` WRONG!!!
(See #3 above)
Still less (5.5) `[`$\exists_x$ `Geq(Age(x),Times(18,Year))]` $\Rightarrow$ `Legal(Do(x,Vote)).` WRONG!!!

Likewise, in the sentence “A squirrel is a rodent”, the meaning is that this is true of all squirrels.

(5.6) $\forall_x$ `Squirrel(x)` $\Rightarrow$ `Rodent(x).`

In the sentence “A vegetarian is a person who eats no meat,” the meaning of “a vegetarian” is “all vegetarians,” and the sentence as a whole is definitional; it is represented by the biconditional,

(5.7) $\forall_x$ `Vegetarian(x)` $\Leftrightarrow$ `[`$\neg\ \exists_y$ `Meat(y)` $\wedge$ `Eats(x,y)].`

Different meanings of “a” may appear together in the same sentence. For instance, the sentence “If a person has a licence, then they can legally drive,” is represented

(5.8) $\forall_x$ `[`$\exists_y$ `License(y)` $\wedge$ `Owns(x,y)]` $\Rightarrow$ `Legal(Do(x,Drive)).`

The statement is true of all people, but each person has only one license. Similarly, the sentence (the implicit content of car ads) “A worthwhile person owns an expensive car,” is represented

(5.9) $\forall_x$ `Worthwhile(x)` $\Rightarrow$ $\exists_y$ `Expensive(y)` $\wedge$ `Car(y)` $\wedge$ `Owns(x,y)`.

The statement is asserted about all people and asserts that each person, if worthwhile, owns at least one car.

The sentence, "Lucy owns a parrot that is larger than a cat," the parrot is existentially quantified (she owns one parrot) but the cat is universally quantified (the parrot is larger than any cat.) (More precisely, the meaning is probably that the parrot is larger than a *typical* cat; but that would not be expressible in first-order logic, and the universal quantifier is closer to what is meant than the existential quantifier.)

(5.10) $\exists_x$ `Parrot(x)` $\wedge$ `Owns(Lucy,x)` $\wedge$ $\forall_y$ `Cat(y)` $\Rightarrow$ `Larger(x,y)`.

6 Don't trust the English: Implication.

Ordinarily, the English form "if α then β" is correctly represented as $\alpha \Rightarrow \beta$ but not always. Also, "α only if β" is represented $\alpha \Rightarrow \beta$ and "α if β" is represented $\beta \Rightarrow \alpha$.

However, there are many exceptions. Often there is an implicit biconditional: the sentence "If the litmus paper turns red, then the substance is an acid," should probably be interpreted as "The litmus paper will turn red if and only if the substance is an acid."

Occasionally, the sentence "If α then β" is best interpreted as $\beta \Rightarrow \alpha$. For instance I recently read a paper in which the sentence "If Joe cleans up his room then he may play in the yard" was represented in the propositional calculus as `Cleans` $\Rightarrow$ `Plays` But that's wrong. Presumably, if Joe cleans up his room, he has the choice whether or not to go out and play; but if he doesn't clean his room, then he is forbidden to play. The correct interpretation is therefore `Play` $\Rightarrow$ `Clean`.

It's also the case that "if ... then ..." in English often carries with it further content which may require an extension of the language, or may be hard or impossible to express in any first-order language.

Causal or temporal structure. For example "If you push the button, the light turns on" means that pushing the button causes the light to turn on. It is tempting to represent this as

(6.1) `Push(Button)` $\Rightarrow$ `TurnOn(Light)`. !!WRONG!!

But then what would the converse `TurnOn(Light)` $\Rightarrow$ `Push(Button)` mean? One reading would be "If the light turns on then you push the button" meaning that the light turning on (should) cause you to push the button; a different reading would be "The light turns on only if you push the button" in which the pushing causes the light to turn on, as in the original sentence. Therefore, if you want to be able express both of these sentences and to distinguish between them, you need a formal language that expresses time and/or causality. For example, in a discrete model of time, one can express the above three sentences as, respectively,

(6.2) $\forall_t$ `Occurs(Push(Button),t)` $\Rightarrow$ `TurnOn(Light,Successor(t))`.
(6.3) $\forall_t$ `TurnOn(Light,t)` $\Rightarrow$ `Occurs(Push(Button),Successor(t))`. and
(6.4) $\forall_t$ `TurnOn(Light,Successor(t))` $\Rightarrow$ `Occurs(Push(Button),t)`.

Hypotheticals. Suppose the sentence "If Los Angeles had a population of ten million, it would be larger than New York," were represented as the material implication

(6.5) `Population(LA)=10,000,000` $\Rightarrow$ `Larger(LA,NYC)`. !!WRONG!!

This sentence is vacuously true because the left hand is false, but so are the sentences

(6.6) $\texttt{Population(LA)=10,000,000} \Rightarrow \texttt{Smaller(LA,NYC).}$
(6.7) $\texttt{Population(LA)=10,000,000} \Rightarrow \texttt{CanFly(Pig).}$

What we are looking for is a representation in which the sentence "If LA had a population of ten million, it would be larger than New York," is true but "If LA had a population of ten million, it would be smaller than New York," or "If LA had a population of ten million, pigs could fly," are false. The representation of this kind of conditional is a difficult problem that has only been partially solved.

7 Alternative correct representation.

Obviously, there are multiple logically equivalent ways of representing a given sentence. For instance "A vegetarian is someone who eats no meat" can be represented in any of the logically equivalent forms:

(7.1) $\forall_x$ $\texttt{Vegetarian(x)} \Leftrightarrow [\neg\exists_y\ \texttt{Meat(y)} \wedge \texttt{Eats(x,y)}].$
(7.2) $\forall_x$ $\texttt{Vegetarian(x)} \Leftrightarrow [\forall_y\ \texttt{Eats(x,y)} \Rightarrow \neg\texttt{Meat(y)}].$
(7.3) $\forall_x$ $[\exists_y\ \texttt{Meat(y)} \wedge \texttt{Eats(x,y)}] \Leftrightarrow \neg\texttt{Vegetarian(x)}.$

and many others.

Less obviously, there may be multiple correct ways of representing a given statement relative to an implicit body of background knowledge. For example, the statement "John's father is bald" may be represented either as

(7.4) $\exists_x$ $\texttt{Father(x,John)} \wedge \texttt{Bald(x).}$ or as
(7.5) $\forall_x$ $\texttt{Father(x,John)} \Rightarrow \texttt{Bald(x).}$

The two are equivalent given that John has exactly one father.

Another example: the statement "Alabama borders Mississippi" can be represented either as

(7.6) `Borders(Alabama, Mississippi).` or as
(7.7) `Borders(Mississippi, Alabama).`

given that the `Borders` relation is symmetric.

8 Limits on First-Order Representation

You should keep in mind, also, that most natural language sentences *cannot* be translated exactly to first-order logic. In many cases you have to sacrifice some of the nuance of the original text. In some cases, there is not even any reasonable approximation in first-order logic.

9 Check your answer.

When you have written down a representation in first-order logic, read it back and make sure that it means what you intended. Keep in mind that quantifiers apply to everything in the universe, and that $\alpha \Rightarrow \beta$ means just "either [not α] or β". For example if you have translated, "A squirrel is a rodent" as $\forall_{\mathtt{x}}$ `Squirrel(X)` $\Rightarrow$ `Rodent(x),` read that formula back as "For every X in the universe, either X is not a squirrel or X is a rodent" and then figure out whether this means the same thing as the original sentence.

10 Background knowledge and the translation from natural language to logic

Problem: We have said that, in order to translate a natural language sentence to a logical formula, you need to understand what the sentence means. But if you think about a *program* that is understanding natural language, "understanding what a sentence means" is exactly finding the corresponding logical formula, so this is completely circular.

Answer: There are a number of solutions. The traditional solution is that the program has a body of domain knowledge that it can call upon, to determine which of the possible logical interpretations of the sentence is most plausible. Supporting this kind of disambiguation is indeed one of the main objectives of research in representing knowledge. To date, this approach has had only limited success in practice.

A second approach is to use as many syntactic and linguistic clues as possible — and there are many more than our discussion above would suggest — but when all these fail, to live with the ambiguity.

A third approach is to translate sentences, not into the full first-order logic, but into "rationalized natural language", a representation that is part-way between natural language and first-order logic. It may not be necessary to resolve the ambiguity in getting to this level of representation. Depending on the ultimate application, it may never be necessary to resolve the ambiguity.