Finite Model Theory: Models

Models are mathematical representations of objects. A class C of objects is modeled in the same way. The way C is modeled is specified by the **model signature**, which identifies the following elements:

1. **A domain.** This is a finite set of elements.
2. **Some relations** over the domain elements.
3. **Some functions** over the domain elements.

For now, we will ignore functions and focus on **relational models**.

To be a valid model for a class of objects, we want to make sure that

1. Every object in C has some model, and
2. Different objects in C have different models.

If we accomplish this, then our model is valid as a model for the class C.

Example 1. We are going to look at a model of **words**. We use the word 'word' synonymously with 'string', so we should understand the class of objects as *sequences* of symbols. Let Σ denote a finite set of symbols and Σ^* the set of all logically possible words of finite length . For example, if $\Sigma = \{a, b, c\}$ then *baba* and *abba* belong to Σ^*. Formally, we want to model the class of objects Σ^*.

Below we define the **successor model** of words. $\mathbb{W}^{\triangleleft}$ is the model signature.

$$\mathbb{W}^{\triangleleft} = \langle \mathcal{D}; \triangleleft, \mathtt{a}, \mathtt{b}, \mathtt{c} \rangle$$

The model signature indicates the domain $\mathcal{D}$, one binary relation $\triangleleft$, and three unary relations $\mathtt{a}, \mathtt{b}$, and $\mathtt{c}$.

$\mathcal{D}$	—	Finite set of elements (positions)
$\triangleleft$	—	A binary relation encoding immediate linear precedence on $\mathcal{D}$
$\mathtt{a}, \mathtt{b}, \mathtt{c}$	—	Unary relations (so subsets of $\mathcal{D}$) encoding positions at which $\mathtt{a}$,$\mathtt{b}$,$\mathtt{c}$ occurs

The model of *abbab* under the signature $\mathbb{W}^{\triangleleft}$ (denoted $\mathcal{M}^{\triangleleft}_{abbab}$) is as follows.

$$\mathcal{M}^{\triangleleft}_{abbab} = \left\langle \begin{array}{l} \{0,1,2,3,4\}; \\ \{(0,1),(1,2),(2,3),(3,4)\}, \\ \{0,3\},\{1,2,4\},\varnothing \end{array} \right\rangle$$

Figure 1 is a diagram illustrating $\mathcal{M}^{\triangleleft}_{abbab}$. Informally, we can think of the domain elements as *events*

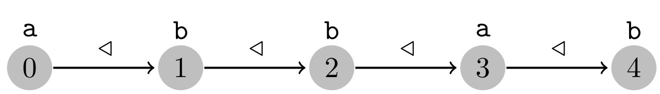

Figure 1: A picture of $\mathcal{M}^{\triangleleft}_{abbab}$

in time, the binary successor relation relates *one event to the next event*, and the unary relations are *properties of the events*.

The model of an object represents the *information* that is *immediately accessible* about the object. Don't let the whole numbers in the domain fool you. They don't have any meaning themselves as numbers; they are just symbols to distinguish one domain element from another.

Exercise 1.

1. Give models for these strings.
   (a) *abc*
   (b) *cacaca*
2. Suppose we removed the unary relations from the signature so the it looks like this: $\mathbb{W}^{\dagger} = \langle \mathcal{D}, \triangleleft \rangle$. Can models with such a signature distinguish all strings in Σ^*?
3. Suppose we removed the successor relation from the signature so it looks like this: $\mathbb{W}^{\ddagger} = \langle \mathcal{D}, \mathtt{a}, \mathtt{b}, \mathtt{c} \rangle$. Can models with such a signature distinguish all strings in Σ^*?
4. For an arbitrary string $x_1 x_2 \ldots x_n \in \Sigma^*$, what is a model for it?
5. Phonological theories often uses properties of speech sounds as representational elements, not segments. How could you define a signature for a model that refers to features? What would the model of *can* [kæn] look like?

Exercise 2. True or False?

1. $1 \triangleleft 2$
2. $1 \triangleleft 3$
3. $\triangleleft\,(1, 2)$
4. $\triangleleft\,(1, 3)$
5. $\mathtt{a}(1)$
6. $\mathtt{a}(2)$
7. $\mathtt{b}(3)$
8. $\mathtt{b}(4)$

Exercise 3. Suppose we want to know the relative position of 1 and 4. For example, which one occurs earlier in the word? Is that information immediately accessible from the model of *abbab*?

Summary Models are representations of objects. We can make models for any objects, not just strings. We can make models of tree structures, graph structures, and other objects such as molecules or faces. We start with strings because they are simple. When making a model you have to determine the model signature. This is the information you want to represent about the object. The relational models we looked at are sometimes referred to as *relational structures*.

Also, for any class of objects, we can actually make many *different models* of it. For example, for words, we can make different models, depending on the kind of information we think it is important to represent. This is where the scientist can *choose*, to some degree anyway, the level of abstraction. Which model is 'right'? There is no intrinsically correct answer to that question.

Next, we are going to learn how to use logic to define properties of objects. Later we will see how to use logic to map one object to another, possibly of a different class.

References

[1] Herbert B. Enderton. *A Mathematical Introduction to Logic.* Academic Press, 2nd edition, 2001.
[2] Shawn Hedman. *A First Course in Logic.* Oxford University Press, 2004.
[3] Wilfrid Hodges. *Model Theory.* Cambridge University Press, 1993.
[4] Leonid Libkin. *Elements of Finite Model Theory.* Springer, 2004.