系统建模部分中文版

我们认为这是一个系统设计问题，从系统设计问题到数学模型，需要抽象。最终选择的抽象方法是基于UML统一建模语言的系统建模方法。对于温度控制系统这样的问题域，我们使用用例图描述用户和温控系统的可能使用场景，然后设计了MVC体系结构框架解决温控系统与用户的交互问题，使用类图描述如何控制温度这个问题域中出现的对象，建立了一个最基本的问题域模型。最终确定要研究的核心问题是在中央处理器上如何综合传感器发来的和用户发来的数据。这个核心问题就是所要研究的数学模型。下面详细阐述从智能温度系统中抽象数学问题的方案。

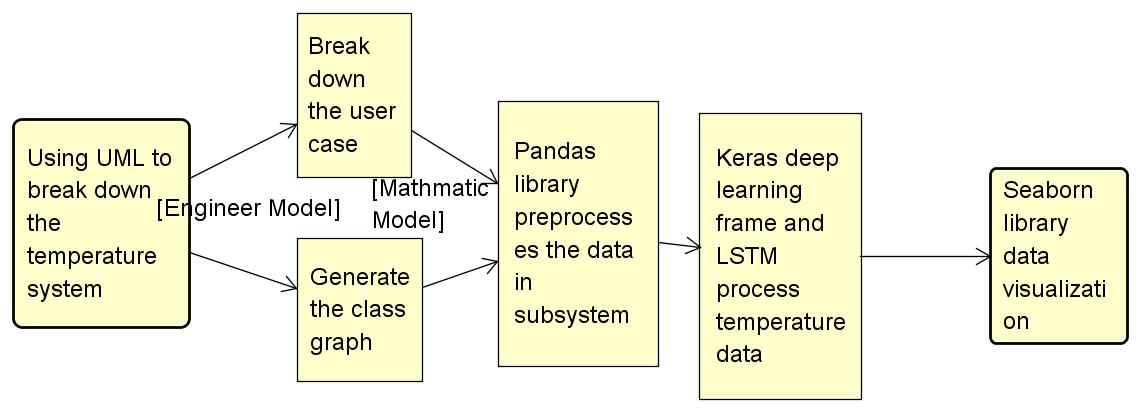


Figure 1工程化建模过程流

1、面向对象分析——用例建模

**用例建模是面向对象分析的第一步。用例包括智能温度系统中的用例以及外部对象，在这个用例图中的外部对象，有用户和中央处理系统两个对象。对象的交互过程是通过用例完成的，用例也就是所谓的场景。用户面对的场景是如何去设定温度程序，换句话说用户去编程控制温度。用户编制的程序应该是能够表征温度的函数，比如说可能是输入温度的函数，然后通过程序，把它翻译成MVC框架中的控制器所能识别的数据流，就可以控制屋内的温度和湿度等信息。对于节省能量、适应用户的喜好这两个用例，在某种程度上存在矛盾，一个系统在运行的过程中必须要先保证用户的满意才能节省能量。用户的喜好判断主要是由系统完成的。有关用户喜好的用例是包含设置温度程序的子用例。用户编程设定的温度场景通过因特网实现，因特网是用户通过自己的计算机，或者是用户的嵌入式终端，如手机设定温度。但是用户编程设定温度非常麻烦，不能总是通过编写程序设计，这样效率极低。因此，应当设计一个用户可图形编程的界面与用户交互，从而算出温度的变化。这也就是刚才体系结构框架里所述的界面设计。**

**下面将介绍与系统有关的几个用例，这几个用例都能和系统作用感知外部的温度变化，温度变化主要是从传感器中感知，其中有一个扩展用例感知的是湿度的变化，需要湿度传感器。还有一个要感知空气污染水平的传感器。系统除了要从这几个传感器中接收数据，并计算出温度应该怎么变化，还应该能够算出在室内有多少个人，根据人数进一步调整室内温度。例如，如果在一个教室里面，有一百个人，此时空气非常污浊。系统这时要控制空气调节器，开的比较大。但是如果在一个30个人的屋子里，这个空气调节器不用开这么大，因为只有30个人的屋子里面的空气，相对来说比较清新，这也是本系统可以节能的一个原因。**

**综上所述，系统要有一个面向用户的图形界面，能够在界面上进行显式的编程，同时系统也能够根据用户的数据，学习出一个模型，根据用户的人数，根据传感器发出来的各种数据去学习固定的模式。也就是说我们需要据此抽象出数学模型。**

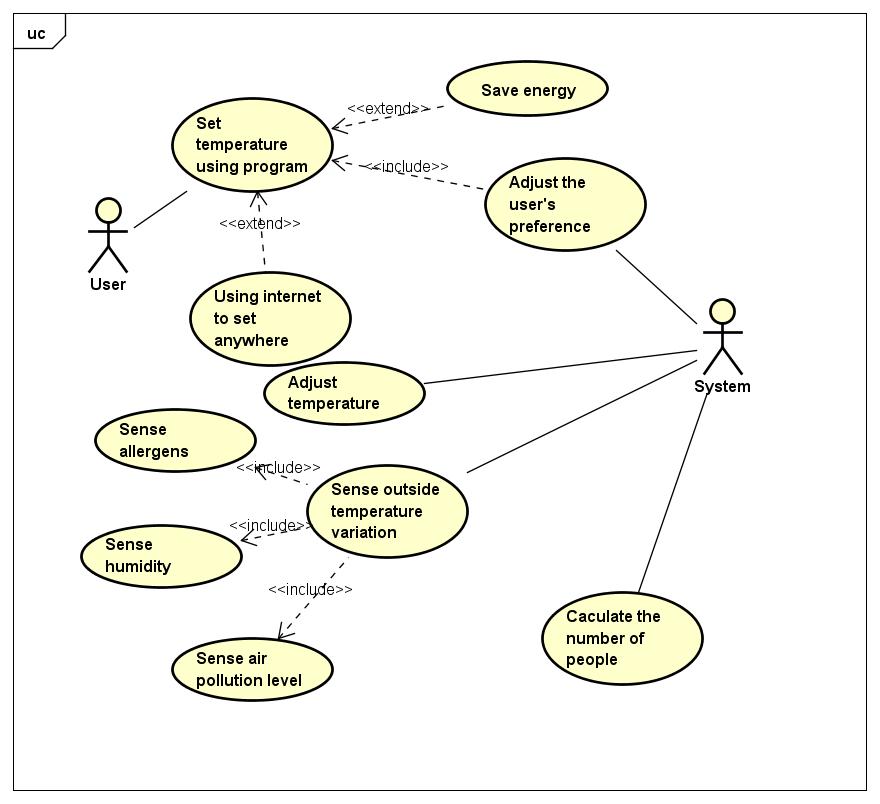


Figure 2温控系统需求分析用例（UML统一建模语言）

2、体系结构框架设计——MVC框架

我们基于MVC设计出一个面向用户和智能温度系统的体系结构框架。MVC框架是一个体系结构框架，由三部分组成：模型对象，视图对象和控制器对象。

其中的控制器对象实例化为温度系统中心处理器。我们选择的是一个核心控制器，它在智能家庭系统中控制所有的任务，并完成核心的计算功能。物联网芯片在智能家居领域中比较广泛，比如树莓派、和英特尔的凌动CPU。他们都可以完成相关的计算工作，并且在神经网络计算棒的支持下奠定了深度学习的神经网络的基础，成为下一步选择模型的依据。

视图对象是人机交互的窗口，当用户需要访问智能控制家居系统时，首先需要通过手机应用或者电脑端应用访问系统界面，在下面解释中我们会看到，界面应该具有可编程的特性，以便让用户去图形化编程，控制这个系统的温度。

模型对象具体指LSTM神经网络模型。我们应用Keras深度学习框架实现了这个模型，具体的模型将在后文中讨论，相关的代码以及注释作为附录。

3、温度系统的抽象映射——基于类的建模

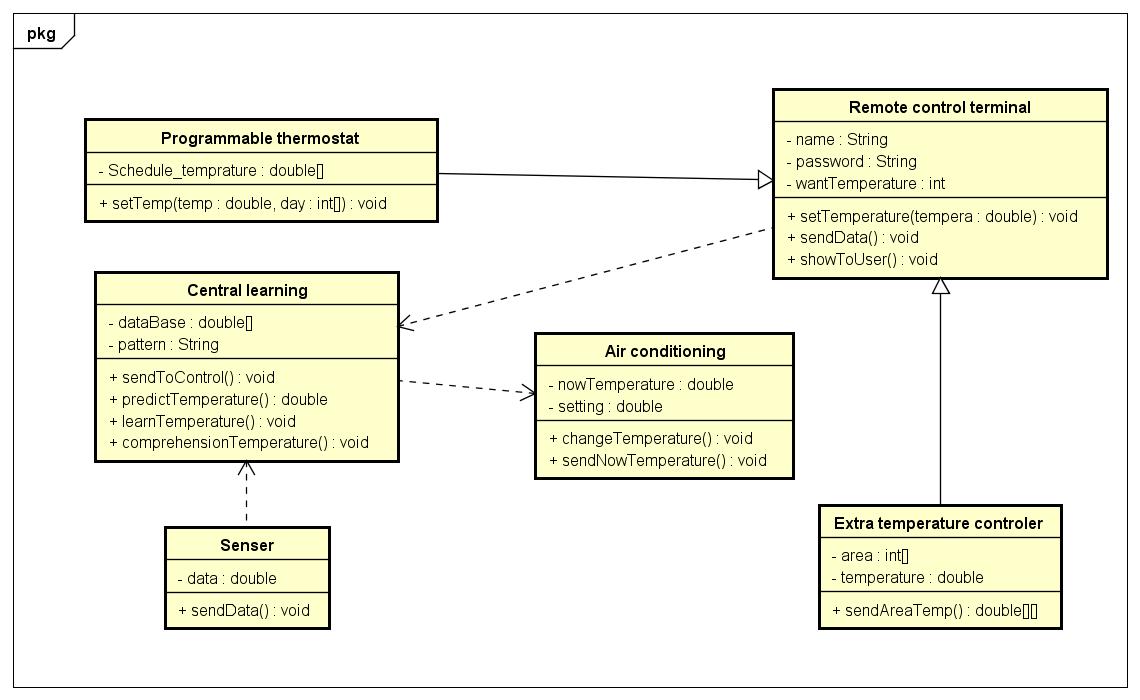
**我们分析出不同的用例，也就是场景。接下来需要根据场景进一步建模，需要进一步需求分析并作系统设计。类图起到了映射的作用，把问题域映射成抽象的对象。绘制出类图，类图中有几个对象呢？程控的温度设定器的实质是上述建立MVC框架中所述的图形温度控制界面。用户还可以在图形界面中输入函数计算，所以也属于程控温度器。这个程控温度器对象中有属性，也有方法。这个方法把这个用户的输入、用户的编程转换成移动数据，发给远程终端。远程终端能够感知到这个消息。远程终端负责接收用户数据，同时负责发送数据。把数据发送给MVC框架中的核心处理器，简而言之是一台服务器。在物联网中设置的服务器不是特别大型的，更多的是嵌入式为主的微服务器。但是考虑到在建模的过程中会用到神经网络，在这个系统设计时，处理器必须有USB神经网络加速器接口。因为我们要处理的不是少量数据，而是大量的用户数据。**

Figure 3基于类的建模

**另一方面，也要考虑到题目中所给的一些其他限定，只有一个温度调节器和额外温度调节器的情况下，怎么分别对系统产生影响？在建模的时候要考虑到，额外的温度控制器对象有继承关系，继承于远程控制终端。在刚才的面向类的建模过程中，考虑了在什么样的场景下使用，在这个场景中，在这个问题域中有哪些类？我们需要做些什么？最终把温度系统这样的实际问题归结为数学问题——处理各个传感器和用户发来的数据。在接下来过程中，要把这个对工程的建模转化成对数学问题的建模，更详细的讨论细节的数学问题。**