# 2.3 输出解析器

在使用 GPT-4 或者类似的大型模型时，一个常见的挑战是如何将模型生成的输出转化为我们可以在代码中直接使用的格式。这里，我们会使用 LangChain 的 输出解析器（OutputParsers） 工具来解决这个问题。

虽然语言模型输出的文本信息可能非常有用，但开发的应用与真实的软件数据世界连接的时候，我们更希望得到的不仅仅是文本，而是更加结构化的数据。为了在应用程序中展示这些信息，我们需要将这些输出转换为某种常见的数据格式。我们可以编写一个函数来提取这个输出，但这并不理想。比如在提示结尾加上“请输出答案为JSON格式”，模型会返回字符串形式的JSON，我们还需要通过函数将其转化为JSON对象。但是在实践中，我们常常会遇到异常问题，例如返回的字符串JSON无法被正确解析。

处理生产环境中的数据时，我们更可能会遇到千奇百怪的输入，导致模型的响应无法解析，增加额外的补丁来进行处理异常。这就使得整个处理流程变得更为复杂。

结构化数据,如数组或JSON对象, 在软件开发中起着至关重要的作用, 它提高了数据处理的效率，便利了数据的存储和检索，支持了数据分析，并且有助于提高数据质量。

还有,大语言模型目前确实存在一些问题，例如机器幻觉，这是指模型在理解或生成文本时产生的错误或误解。另一个问题是为了显得“聪明”，模型有时候会加入不必要的冗长和华丽的语句，这可能会导致模型过度详细，显得“话痨”了。比如你提示的结尾是“你的答案是：”，模型就不会“话痨”了。

在真实的开发环境中，我们不仅希望获取模型的输出结果，而且还希望能够进行后处理，比如解析输出的结构化数据。

这就是为什么在大语言模型的开发中，结构化数据，如数组或JSON对象，显得尤为重要。他们可以帮助我们更好地理解和处理模型的输出结果，比如通过解析输出的JSON对象，我们可以得到模型的预测结果，而不仅仅是一个长长的文本字符串。我们也可以根据需要对这些结果进行进一步的处理，例如提取关键信息，进行数据分析等。这样，我们不仅可以得到模型的“直接回答”，而且可以根据自己的需求进行定制化的**后处理**， 比如传递给下一个任务函数，从而更好地利用大语言模型。

这就是输出解析器的用武之地。

输出解析器是一组工具，它们的主要功能是处理和格式化模型的输出。这个工具组包括了几个部分，对于我们的需求来说，最关键的部分是 Structured Output Parser。这个工具可以把我们之前作为字符串返回的模型输出，转化为可以在代码中直接使用的数据结构。

输出解析器是帮助结构化语言模型响应的类。一个输出解析器主要需要实现两种方法：

“get\_format\_instructions”：这是一种返回字符串的方法，该字符串包含如何格式化语言模型输出的指示。 “parse”：这是一种接受字符串（模型的响应）并将其解析为某种结构的方法。

本书中,我们选取最常用的列表和JSON格式的输出解析器,通过代码来解释输出解析器的必要性。Langchain 还提供了 Datetime parser, Enum parser , Structured output parser 等类型的输出解析器。