# 文本分割方法

在处理大规模文本数据时，LangChain提供了多种文本分割方法，以满足各种类型的应用需求。本文将详细介绍这些方法的特性和工作原理。

#### 1. 按字符分割

这是最简单的方法。它基于字符（默认为”“）进行分割，并通过字符数量来衡量块的大小。

* 文本如何分割：按单个字符
* 块大小如何测量：按字符数量

#### 2. 代码分割

CodeTextSplitter允许你对多种语言的代码进行分割。导入枚举Language并指定语言即可。

#### 3. Markdown标题文本分割器

这种分割器的动机源于许多聊天或问答应用需要在嵌入和向量存储之前将输入文档进行分块。Pinecone的这些注释提供了一些有用的提示。

#### 4. 递归按字符分割

这是通用文本推荐的文本分割器。它由一系列字符参数化。直到块足够小，它会尝试按顺序分割它们。默认列表是[“”, “”, ” “,””]。这样做的效果是尽可能地将所有段落（然后是句子，然后是单词）保持在一起，因为这些通常看起来是最强的语义相关的文本部分。

#### 5. 按标记分割

语言模型有一个标记限制。你不应该超过这个标记限制。因此，当你将文本分割成块时，计算标记的数量是个好主意。有许多标记化器。当你在文本中计数标记时，应该使用与语言模型中使用的相同的标记化器。

通过以上内容，我们可以看到LangChain提供的各种文本分割方法能够满足不同类型的文本处理需求，无论是基于字符的分割，还是基于特定语言代码的分割，甚至是针对Markdown格式文本的分割，都具有各自的优点和特性，为处理和转换大规模文本提供了有效的工具。