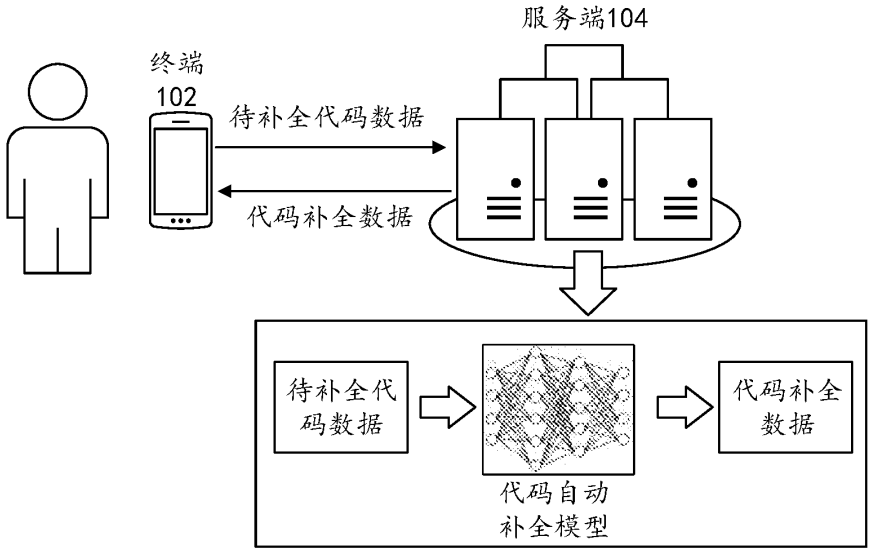

服务端104
终端
102
待补全代码数据
代码补全数据
待补全代码数据
代码自动
补全模型
代码补全数据

图 1

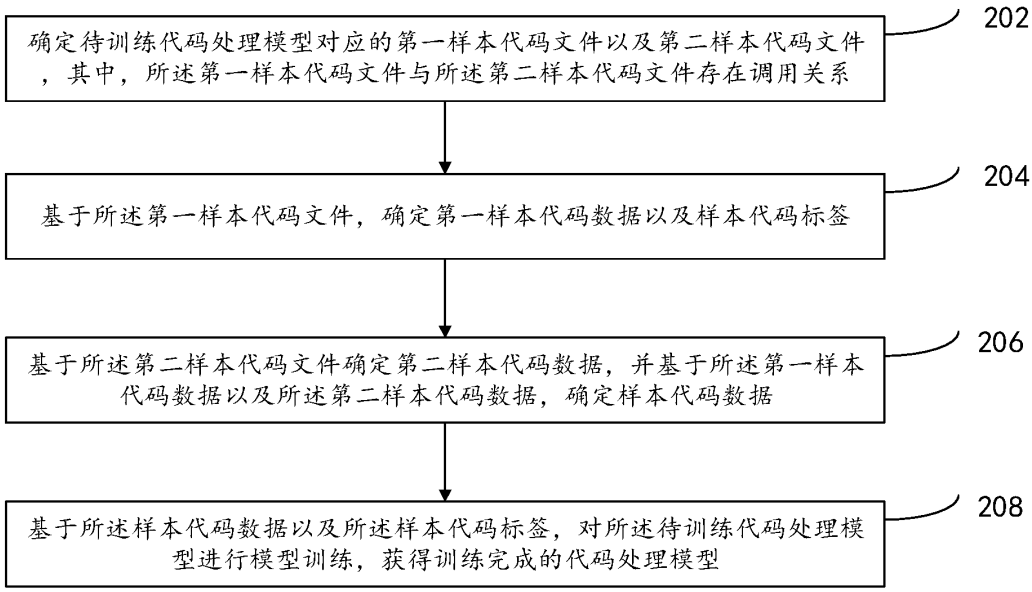

确定待训练代码处理模型对应的第一样本代码文件以及第二样本代码文件，其中，所述第一样本代码文件与所述第二样本代码文件存在调用关系
202
基于所述第一样本代码文件，确定第一样本代码数据以及样本代码标签
204
基于所述第二样本代码文件确定第二样本代码数据，并基于所述第一样本代码数据以及所述第二样本代码数据，确定样本代码数据
206
基于所述样本代码数据以及所述样本代码标签，对所述待训练代码处理模型进行模型训练，获得训练完成的代码处理模型
208

图 2

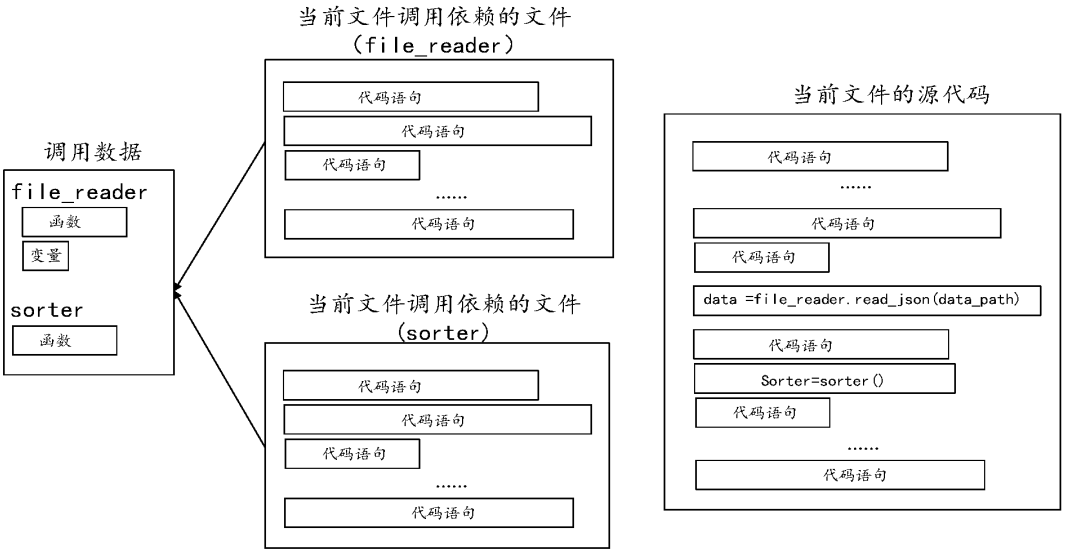

当前文件调用依赖的文件
(file_reader)
代码语句
代码语句
代码语句
.....
代码语句
当前文件的源代码
代码语句
.....
代码语句
代码语句
data =file_reader.read_json(data_path)
代码语句
Sorter=sorter()
代码语句
.....
代码语句
调用数据
file_reader
函数
变量
sorter
函数
当前文件调用依赖的文件
(sorter)
代码语句
代码语句
代码语句
.....
代码语句

图 3

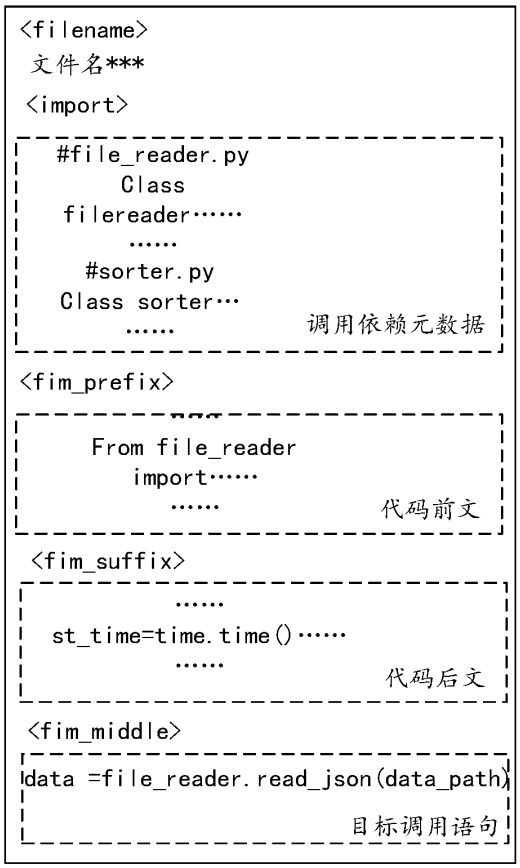

<filename>
文件名***
<import>
#file_reader.py
Class
filereader……
……
#sorter.py
Class sorter…
……
调用依赖元数据
<fim_prefix>
From file_reader
import……
……
代码前文
<fim_suffix>
……
st_time=time.time()……
……
代码后文
<fim_middle>
data =file_reader.read_json(data_path)
目标调用语句

图 4

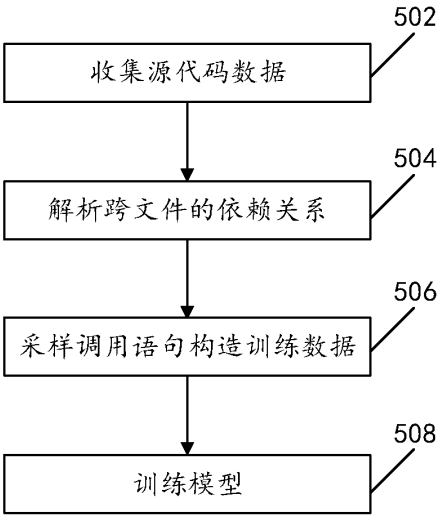

502
收集源代码数据
504
解析跨文件的依赖关系
506
采样调用语句构造训练数据
508
训练模型

图 5

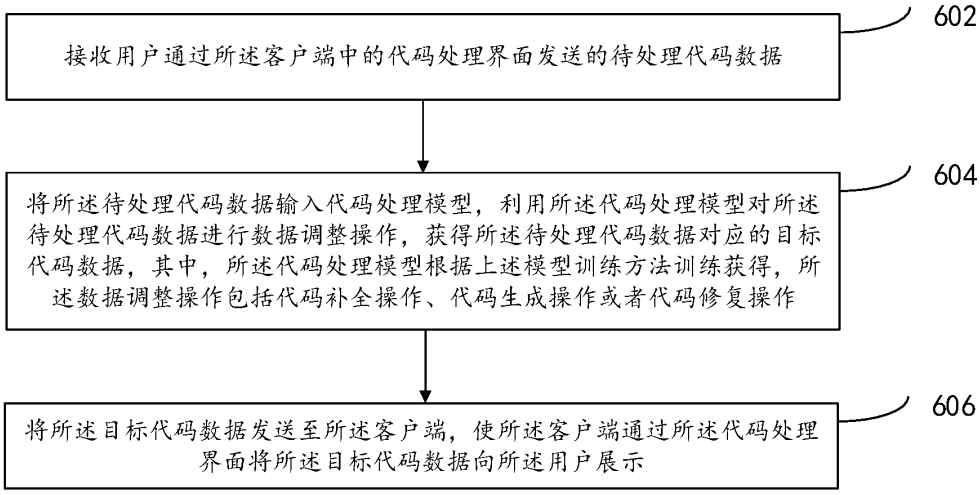

602
接收用户通过所述客户端中的代码处理界面发送的待处理代码数据
604
将所述待处理代码数据输入代码处理模型，利用所述代码处理模型对所述待处理代码数据进行数据调整操作，获得所述待处理代码数据对应的目标代码数据，其中，所述代码处理模型根据上述模型训练方法训练获得，所述数据调整操作包括代码补全操作、代码生成操作或者代码修复操作
606
将所述目标代码数据发送至所述客户端，使所述客户端通过所述代码处理界面将所述目标代码数据向所述用户展示

图 6

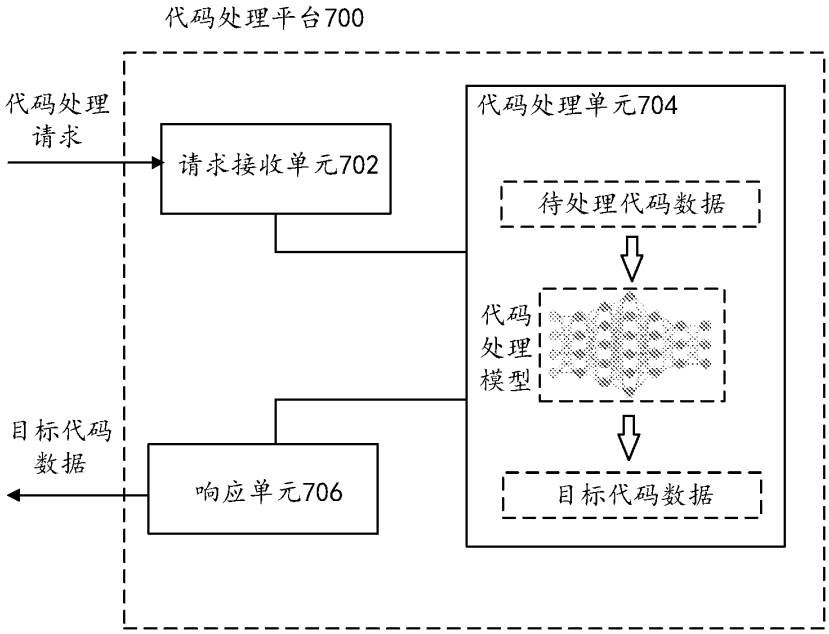

代码处理平台700
代码处理请求
请求接收单元702
代码处理单元704
待处理代码数据
代码处理模型
目标代码数据
目标代码数据
响应单元706

图 7

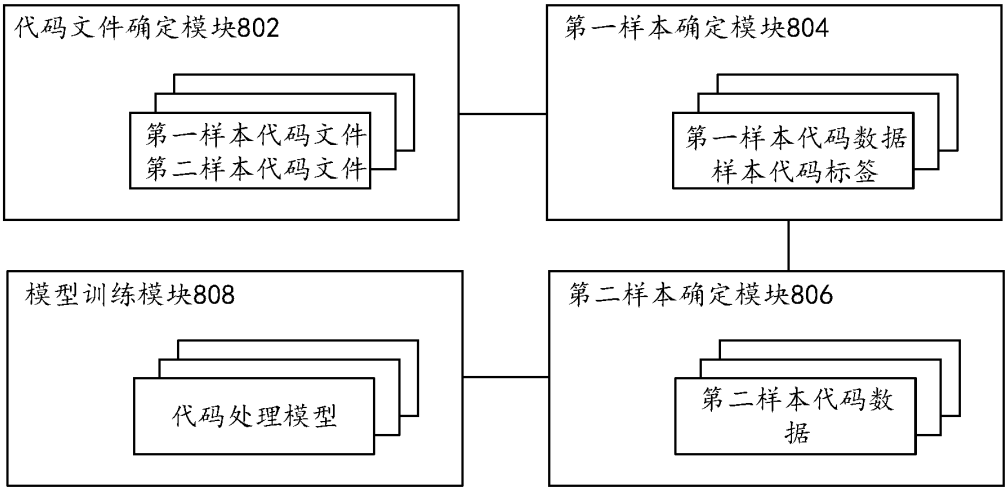

代码文件确定模块802
第一样本代码文件
第二样本代码文件
第一样本确定模块804
第一样本代码数据
样本代码标签
模型训练模块808
代码处理模型
第二样本确定模块806
第二样本代码数据

图 8

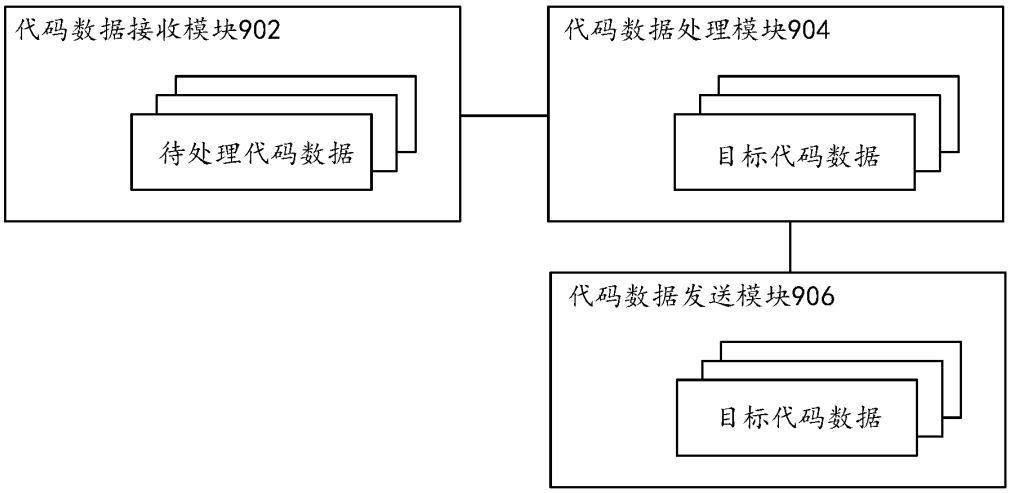

代码数据接收模块902
待处理代码数据
代码数据处理模块904
目标代码数据
代码数据发送模块906
目标代码数据

图 9

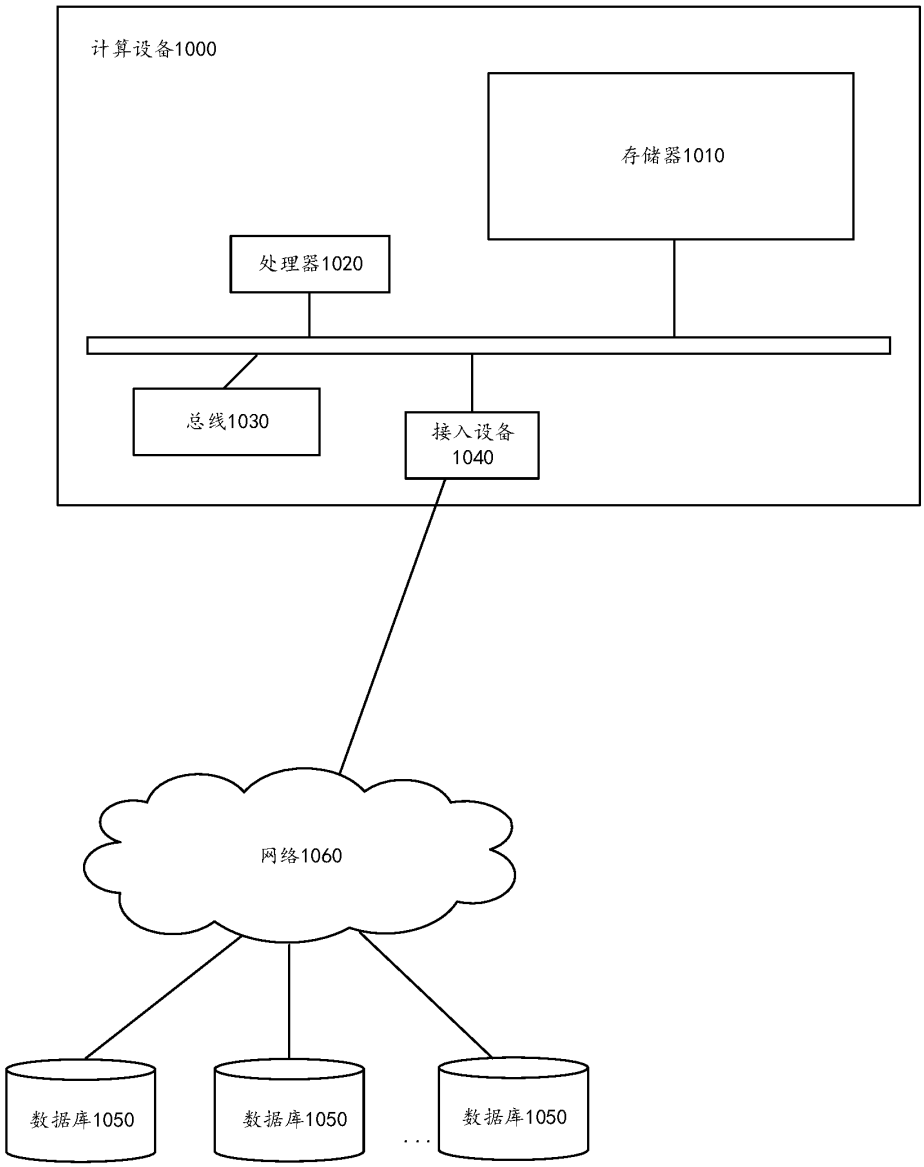

计算设备1000
存储器1010
处理器1020
总线1030
接入设备
1040
网络1060
数据库1050
数据库1050
数据库1050

图 10