



(12) 发明专利申请

(10) 申请公布号 CN 112445933 A

(43) 申请公布日 2021. 03. 05

(21) 申请号 202011417330.2

G06N 20/00 (2019.01)

(22) 申请日 2020.12.07

(71) 申请人 腾讯音乐娱乐科技(深圳)有限公司

地址 518052 广东省深圳市前海深港合作区前湾一路1号A栋201室(入驻深圳市前海商务秘书有限公司)

(72) 发明人 陈肇康 林梅露 吴斌 雷兆恒

(74) 专利代理机构 深圳市深佳知识产权代理事务

所(普通合伙) 44285

代理人 陈彦如

(51) Int. Cl.

G06F 16/65 (2019.01)

G06K 9/62 (2006.01)

G06N 3/04 (2006.01)

G06N 3/08 (2006.01)

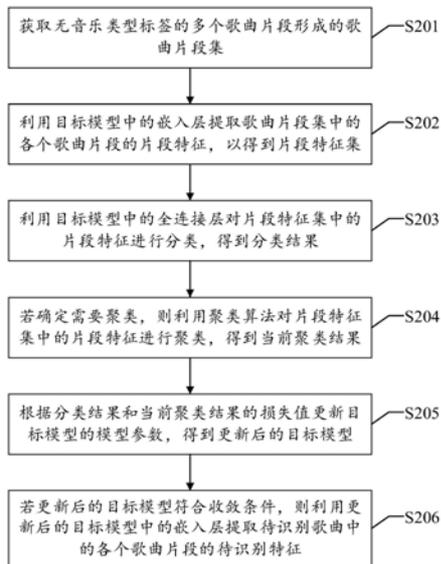
权利要求书3页 说明书13页 附图7页

(54) 发明名称

一种模型训练方法、装置、设备及存储介质

(57) 摘要

本申请公开了一种模型训练方法、装置、设备及存储介质,在该方案中,利用无音乐类型标签的歌曲片段集作为训练数据,因此无需对整首歌曲或歌曲片段打音乐类型标签,简化了模型训练前的准备工作,降低了时间成本。并且,符合收敛条件的模型中的嵌入层可以提取待识别歌曲中的各个歌曲片段的待识别特征,因此可识别各个歌曲片段对应的音乐类型。可见,本申请可以精准识别待识别歌曲中的各个歌曲片段对应的音乐类型,而不是笼统地对整首歌曲进行音乐类型识别,故本申请能够区分一首歌曲中不同音乐类型的片段,提升了歌曲音乐类型的识别精度和准确度,可适用于识别具有多元化音乐元素的歌曲。



1. 一种模型训练方法,其特征在于,包括:
 - 获取无音乐类型标签的多个歌曲片段形成的歌曲片段集;
 - 利用目标模型中的嵌入层提取所述歌曲片段集中的各个歌曲片段的片段特征,以得到片段特征集;
 - 利用所述目标模型中的全连接层对所述片段特征集中的片段特征进行分类,得到分类结果;
 - 若确定需要聚类,则利用聚类算法对所述片段特征集中的片段特征进行聚类,得到当前聚类结果;
 - 根据所述分类结果和所述当前聚类结果的损失值更新所述目标模型的模型参数,得到更新后的目标模型;
 - 若所述更新后的目标模型符合收敛条件,则利用所述更新后的目标模型中的嵌入层提取待识别歌曲中的各个歌曲片段的待识别特征。
2. 根据权利要求1所述的模型训练方法,其特征在于,所述确定需要聚类,包括:
 - 判断聚类算法执行周期是否达到阈值;
 - 若聚类算法执行周期达到阈值,则执行聚类操作;
 - 若聚类算法执行周期未达到阈值,则不进行聚类。
3. 根据权利要求2所述的模型训练方法,其特征在于,所述确定不需要聚类之后,还包括:
 - 若确定存在历史聚类结果,则根据所述分类结果和所述历史聚类结果的损失值更新所述目标模型的模型参数,得到更新后的目标模型。
4. 根据权利要求3所述的模型训练方法,其特征在于,还包括:
 - 若确定不存在所述历史聚类结果,则根据所述分类结果和类别标签的损失值更新所述目标模型的模型参数,得到更新后的目标模型,用所述更新后的目标模型替换所述目标模型,并循环进行模型训练;
 - 其中,所述歌曲片段集中的、属于同一歌曲的、不同歌曲片段标注有不同或相同的类别标签。
5. 根据权利要求1所述的模型训练方法,其特征在于,所述更新后的目标模型符合收敛条件,包括:
 - 若确定所述当前聚类结果与历史聚类结果的差异符合预设要求,则更新符合所述预设要求的次数,若所述次数不小于预设阈值,则确定所述更新后的目标模型符合收敛条件。
6. 根据权利要求5所述的模型训练方法,其特征在于,还包括:
 - 若确定所述差异不符合所述预设要求,或所述次数小于所述预设阈值,则确定所述更新后的目标模型不符合收敛条件,用所述更新后的目标模型替换所述目标模型,用所述当前聚类结果替换所述历史聚类结果,并循环进行模型训练。
7. 根据权利要求1所述的模型训练方法,其特征在于,所述根据所述分类结果和所述当前聚类结果的损失值更新所述目标模型的模型参数,得到更新后的目标模型,包括:
 - 利用目标损失函数计算所述分类结果和所述当前聚类结果的损失值;
 - 根据所述损失值分别优化所述目标模型中的嵌入层参数和全连接层参数,得到所述更新后的目标模型。

8. 根据权利要求1所述的模型训练方法,其特征在于,所述利用聚类算法对所述片段特征集中的片段特征进行聚类,得到当前聚类结果,包括:

根据不同歌曲之间的相似性将所述片段特征集划分为多个相似特征子集;

分别对所述多个相似特征子集中的片段特征进行聚类,以得到每个相似特征子集对应的第一聚类子结果,将所有第一聚类子结果确定为所述当前聚类结果。

9. 根据权利要求8所述的模型训练方法,其特征在于,所述根据不同歌曲之间的相似性将所述片段特征集划分为多个相似特征子集,包括:

从所述片段特征集中提取第一歌曲对应的第一特征组,以及第二歌曲对应的第二特征组;所述第一歌曲和所述第二歌曲为所述片段特征集对应的各个歌曲中的任两首;

确定所述第一特征组和所述第二特征组的组相似性,若所述组相似性大于预设阈值,则将所述第一特征组和所述第二特征组划分至同一相似特征子集;否则,将所述第一特征组和所述第二特征组划分至不同相似特征子集。

10. 根据权利要求9所述的模型训练方法,其特征在于,所述确定所述第一特征组和所述第二特征组的组相似性,包括:

计算所述第一特征组中的所有特征的第一平均特征值,以及所述第二特征组中的所有特征的第二平均特征值;计算所述第一平均特征值和所述第二平均特征值的特征相似性,将所述特征相似性确定为所述组相似性。

11. 根据权利要求1所述的模型训练方法,其特征在于,所述利用聚类算法对所述片段特征集中的片段特征进行聚类,得到当前聚类结果,包括:

按照不同歌曲将所述片段特征集划分为多个歌曲特征子集;

分别对所述多个歌曲特征子集中的片段特征进行聚类,以得到每个歌曲特征子集对应的第二聚类子结果,将所有第二聚类子结果确定为所述当前聚类结果。

12. 根据权利要求1至11任一项所述的模型训练方法,其特征在于,还包括:

利用k-近邻算法、神经网络算法、支持向量机或随机森林算法训练得到识别分类器;

将所述待识别特征输入所述识别分类器,以输出所述待识别歌曲中的各个歌曲片段的音乐类型。

13. 一种模型训练装置,其特征在于,包括:

获取模块,用于获取无音乐类型标签的多个歌曲片段形成的歌曲片段集;

提取模块,用于利用目标模型中的嵌入层提取所述歌曲片段集中的各个歌曲片段的片段特征,以得到片段特征集;

分类模块,用于利用所述目标模型中的全连接层对所述片段特征集中的片段特征进行分类,得到分类结果;

聚类模块,用于若确定需要聚类,则利用聚类算法对所述片段特征集中的片段特征进行聚类,得到当前聚类结果;

更新模块,用于根据所述分类结果和所述当前聚类结果的损失值更新所述目标模型的模型参数,得到更新后的目标模型;

应用模块,用于若所述更新后的目标模型符合收敛条件,则利用所述更新后的目标模型中的嵌入层提取待识别歌曲中的各个歌曲片段的待识别特征。

14. 一种电子设备,其特征在于,所述电子设备包括处理器和存储器;其中,所述存储器

用于存储计算机程序,所述计算机程序由所述处理器加载并执行以实现如权利要求1至12任一项所述的模型训练方法。

15.一种存储介质,其特征在于,所述存储介质中存储有计算机可执行指令,所述计算机可执行指令被处理器加载并执行时,实现如权利要求1至12任一项所述的模型训练方法。

一种模型训练方法、装置、设备及存储介质

技术领域

[0001] 本申请涉及计算机技术领域,特别涉及一种模型训练方法、装置、设备及存储介质。

背景技术

[0002] 目前,在训练模型之前,需要对众多歌曲打标签(即人工标注各个歌曲所属的音乐类型),从而可获得有音乐类型标签的训练数据。其中,打标签的工作量较大,导致模型训练前的准备工作繁杂,需要耗费大量时间成本。并且,由于训练数据中标注的是各个歌曲所属的音乐类型,因此训练完成的模型只能识别整首歌曲所属的音乐类型,无法区分歌曲中不同音乐类型的元素,故当前模型对于歌曲音乐类型的识别精度和准确度有待提高。

发明内容

[0003] 有鉴于此,本申请的目的在于提供一种模型训练方法、装置、设备及存储介质,以提高音乐类型的识别精度和准确度。其具体方案如下:

[0004] 为实现上述目的,一方面,本申请提供了一种模型训练方法,包括:

[0005] 获取无音乐类型标签的多个歌曲片段形成的歌曲片段集;

[0006] 利用目标模型中的嵌入层提取所述歌曲片段集中的各个歌曲片段的片段特征,以得到片段特征集;

[0007] 利用所述目标模型中的全连接层对所述片段特征集中的片段特征进行分类,得到分类结果;

[0008] 若确定需要聚类,则利用聚类算法对所述片段特征集中的片段特征进行聚类,得到当前聚类结果;

[0009] 根据所述分类结果和所述当前聚类结果的损失值更新所述目标模型的模型参数,得到更新后的目标模型;

[0010] 若所述更新后的目标模型符合收敛条件,则利用所述更新后的目标模型中的嵌入层提取待识别歌曲中的各个歌曲片段的待识别特征。

[0011] 另一方面,本申请还提供了一种模型训练装置,包括:

[0012] 获取模块,用于获取无音乐类型标签的多个歌曲片段形成的歌曲片段集;

[0013] 提取模块,用于利用目标模型中的嵌入层提取所述歌曲片段集中的各个歌曲片段的片段特征,以得到片段特征集;

[0014] 分类模块,用于利用所述目标模型中的全连接层对所述片段特征集中的片段特征进行分类,得到分类结果;

[0015] 聚类模块,用于若确定需要聚类,则利用聚类算法对所述片段特征集中的片段特征进行聚类,得到当前聚类结果;

[0016] 更新模块,用于根据所述分类结果和所述当前聚类结果的损失值更新所述目标模型的模型参数,得到更新后的目标模型;

[0017] 应用模块,用于若所述更新后的目标模型符合收敛条件,则利用所述更新后的目标模型中的嵌入层提取待识别歌曲中的各个歌曲片段的待识别特征。

[0018] 又一方面,本申请还提供了一种电子设备,所述电子设备包括处理器和存储器;其中,所述存储器用于存储计算机程序,所述计算机程序由所述处理器加载并执行以实现前述模型训练方法。

[0019] 又一方面,本申请还提供了一种存储介质,所述存储介质中存储有计算机可执行指令,所述计算机可执行指令被处理器加载并执行时,实现前述模型训练方法。

[0020] 本申请利用无音乐类型标签的歌曲片段集作为训练数据,因此无需对整首歌曲或歌曲片段打音乐类型标签,简化了模型训练前的准备工作,降低了时间成本。并且,符合收敛条件的模型中的嵌入层可以提取待识别歌曲中的各个歌曲片段的待识别特征,之后即可识别待识别特征对应的音乐类型。可见,本申请可以精准识别待识别歌曲中的各个歌曲片段对应的音乐类型,而不是笼统地对整首歌曲进行音乐类型识别,故本申请能够区分一首歌曲中不同音乐类型的片段,提升了歌曲音乐类型的识别精度和准确度,可适用于识别具有多元化音乐元素的歌曲。

[0021] 相应地,本申请提供的模型训练装置、设备及存储介质,也同样具有上述技术效果。

附图说明

[0022] 为了更清楚地说明本申请实施例或现有技术中的技术方案,下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图仅仅是本申请的实施例,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据提供的附图获得其他的附图。

[0023] 图1为本申请提供的一种本申请适用的物理架构示意图;

[0024] 图2为本申请提供的第一种模型训练方法流程图;

[0025] 图3为本申请提供的第二种模型训练方法流程图;

[0026] 图4为本申请提供的第三种模型训练方法流程图;

[0027] 图5为本申请提供的一种音乐类似识别方法流程图;

[0028] 图6为本申请提供的一种模型训练示意图;

[0029] 图7为本申请提供的一种模型训练装置示意图;

[0030] 图8为本申请提供的一种服务器结构图;

[0031] 图9为本申请提供的一种终端结构图。

具体实施方式

[0032] 目前,模型训练前的准备工作繁杂,需要耗费大量时间成本。并且,当前模型训练模型只能识别整首歌曲,无法区分歌曲中的不同音乐元素,故对于音乐类型的识别精度和准确度有待提高。

[0033] 鉴于目前所存在的上述问题,本申请提出了模型训练方案,该方案简化了模型训练前的准备工作,降低了时间成本,且能够精准识别到一首歌曲中的歌曲片段的类型,而不是笼统地对整首歌曲进行类型识别,故提升了音乐类型的识别精度和准确度,可适用于识

别具有多元化音乐元素的歌曲。

[0034] 为了便于理解,先对本申请所适用的物理框架进行介绍。

[0035] 应理解,本申请提供的模型训练方法可以应用于具有该识别功能的系统或程序中。具体的,具有该识别功能的系统或程序可以运行于服务器、个人计算机等设备中。

[0036] 如图1所示,图1为本申请适用的物理架构示意图。在图1中,具有该识别功能的系统或程序可以运行于服务器,该服务器通过网络从其他终端设备中获取由无音乐类型标签的多个歌曲片段形成的歌曲片段集,然后利用目标模型中的嵌入层提取歌曲片段集中的各个歌曲片段的片段特征,以得到片段特征集;利用目标模型中的全连接层对片段特征集中的片段特征进行分类,得到分类结果;若确定需要聚类,则利用聚类算法对片段特征集中的片段特征进行聚类,得到当前聚类结果;根据分类结果和当前聚类结果的损失值更新目标模型的模型参数,得到更新后的目标模型;若更新后的目标模型符合收敛条件,则利用更新后的目标模型中的嵌入层提取待识别歌曲中的各个歌曲片段的待识别特征,从而可识别各个歌曲片段对应的音乐类型。

[0037] 如图1可知,该服务器可以与多个设备建立通信连接,服务器从这些设备中获取歌曲片段集对应的任意歌曲、待识别歌曲等。当然,上述歌曲片段集也可以以数据库形式存储。服务器通过收集这些设备上传的数据,对相关模型进行训练,以获得符合收敛条件的模型。图1中示出了多种终端设备,在实际场景中可以有更多或更少种类的终端设备参与到模型训练的过程中,具体数量和种类因实际场景而定,此处不做限定,另外,图1中示出了一个服务器,但在实际场景中,也可以有多个服务器的参与,具体服务器数量因实际场景而定。其中,识别分类器也可以基于该服务器训练获得。

[0038] 应当注意的是,本实施例提供的模型训练方法可以离线进行,即服务器本地存储有歌曲片段集、待识别歌曲,其可以直接按照本申请识别歌曲中各个片段的音乐类型。并且,若获得符合收敛条件的模型后,可利用该模型识别任意歌曲中的片段的音乐类型,无需再次训练该模型。音乐类型如:流行音乐(Pop Music),摇滚音乐(Rock Music),嘻哈音乐(Hip Pop Music),高科技舞曲(Techno),R&B音乐(Rhythm Blues),雷鬼音乐(Reggae Music),古典音乐(Classical)等。

[0039] 可以理解的是,上述具有模型训练的系统或程序也可以运行于个人移动终端,也可以作为云端服务程序的一种,具体运作模式因实际场景而定,此处不做限定。

[0040] 结合以上共性,请参见图2,图2为本申请实施例提供的第一种模型训练方法流程图。如图2所示,该模型训练方法可以包括以下步骤:

[0041] S201、获取无音乐类型标签的多个歌曲片段形成的歌曲片段集。

[0042] 需要说明的是,歌曲片段集中包括不同歌曲的歌曲片段。其中,一首歌曲可以划分获得至少一个歌曲片段。若划分一首歌曲得到多个歌曲片段,不同歌曲片段可以重叠,也可以不重叠(一般设置为不重叠);不同歌曲片段的长度可以相等,也可以不相等(一般设置为相等,便于模型训练)。歌曲片段的长度可以设置为任意,如:10s。

[0043] S202、利用目标模型中的嵌入层提取歌曲片段集中的各个歌曲片段的片段特征,以得到片段特征集。

[0044] 其中,目标模型包括:嵌入层和全连接层。嵌入层能够提取歌曲片段集中的各个歌曲片段的片段特征,即:将歌曲片段集中的各个歌曲片段转化为相应的片段特征,全连接层

能够对片段特征集中的所有片段特征进行分类,从而得到分类结果。全连接层可以看作分类器。

[0045] S203、利用目标模型中的全连接层对片段特征集中的片段特征进行分类,得到分类结果。

[0046] S204、若确定需要聚类,则利用聚类算法对片段特征集中的片段特征进行聚类,得到当前聚类结果。

[0047] 聚类结果是对歌曲片段集中的歌曲片段聚类得到。聚类结果是全连接层输出的分类结果的参考目标,即标准值。全连接层想要自己输出的分类结果不断趋近于聚类结果。

[0048] 聚类的具体实现可以采用任意聚类算法,如:DBSCAN算法、k均值聚类算法或高斯混合模型算法等。但聚类的具体对象可以灵活选择,具体请参见如下介绍。

[0049] 在一种具体实施方式中,可以认为同一首歌曲中的所有歌曲片段属于同一类型(实际上同一首歌曲中的不同歌曲片段可能属于不同类型),因此聚类对象可以为:每首歌曲对应的歌曲特征子集中的各个特征。歌曲特征子集如:歌曲A中的所有歌曲片段对应的特征,一个歌曲片段对应一个特征。在一种具体实施方式中,可以认为同一首歌曲中的不同歌曲片段属于不同类型,因此聚类对象可以为:每首歌曲对应的歌曲特征子集中的各个特征。

[0050] 可见,无论是认为同一首歌曲中的所有歌曲片段属于同一类型,还是认为同一首歌曲中的不同歌曲片段属于不同类型,都可以分别对一首歌中的所有歌曲片段进行聚类。故利用聚类算法对片段特征集中的片段特征进行聚类,得到当前聚类结果,包括:按照不同歌曲将所述片段特征集划分为多个歌曲特征子集;分别对所述多个歌曲特征子集中的片段特征进行聚类,以得到每个歌曲特征子集对应的第二聚类子结果,将所有第二聚类子结果确定为所述聚类结果。假设歌曲片段集由5首歌曲划分片段获得,那么歌曲片段集对应应有5首歌曲,也就有5个歌曲特征子集,分别对这5个歌曲特征子集中的各个特征进行聚类,可得到5个第二聚类子结果。每个第二聚类子结果中包括多个簇。此方式可考虑同一歌曲中的不同音乐类型的片段,提高训练精度和准确性。

[0051] 在一种具体实施方式中,无论认为同一首歌曲中的所有歌曲片段属于同一类型,还是认为同一首歌曲中的不同歌曲片段属于不同类型,都可以同时考虑不同歌曲之间的相似性,因此聚类对象可以为:相似的几首歌曲包括的所有歌曲片段。故利用聚类算法对片段特征集中的片段特征进行聚类,得到当前聚类结果,包括:根据不同歌曲之间的相似性将片段特征集划分为多个相似特征子集;分别对多个相似特征子集中的片段特征进行聚类,以得到每个相似特征子集对应的第一聚类子结果,将所有第一聚类子结果确定为聚类结果。假设歌曲片段集由5首歌曲划分片段获得,那么歌曲片段集对应应有5首歌曲,对这5首歌曲的特征进行相似性分类,假设获得3个相似特征子集,那么进一步分别对这3个相似特征子集中的各个特征进行聚类,可得到3个第一聚类子结果。每个第一聚类子结果中包括多个簇。此方式可考虑同一歌曲中的不同音乐类型的片段,以及不同歌曲的相似性,提高训练精度和准确性。

[0052] 其中,根据不同歌曲之间的相似性将片段特征集划分为多个相似特征子集,包括:从片段特征集中提取第一歌曲对应的第一特征组,以及第二歌曲对应的第二特征组;第一歌曲和第二歌曲为片段特征集对应的各个歌曲中的任两首;确定第一特征组和第二特征组的组相似性,若组相似性大于预设阈值,则将第一特征组和第二特征组划分至同一相似特

征子集;否则,将第一特征组和第二特征组划分至不同相似特征子集。相似性可利用余弦夹角或欧式距离进行计算。

[0053] 其中,确定第一特征组和第二特征组的组相似性,包括:计算第一特征组中的所有特征的第一平均特征值,以及第二特征组中的所有特征的第二平均特征值;计算第一平均特征值和第二平均特征值的特征相似性,将特征相似性确定为组相似性。其中,第一特征组中的所有特征的第一平均特征值为:第一特征组中的所有特征的平均值。相应的,第二特征组中的所有特征的第二平均特征值为:第二特征组中的所有特征的平均值。嵌入层会输出每个歌曲片段对应的特征。

[0054] 若第一歌曲为A,划分A获得3个歌曲片段:A1、A2、A3,那么第一特征组中即包括3个特征,这3个特征分别与A1、A2、A3相对应。若第二歌曲为B,划分B获得5个歌曲片段:B1、B2、B3、B4、B5,那么第二特征组中即包括5个特征,这5个特征分别与B1、B2、B3、B4、B5相对应。第一歌曲和第二歌曲的相似性,即为第一特征组和第二特征组的组相似性。而第一特征组和第二特征组的组相似性可以按照下述过程计算:计算A1、A2、A3的平均值 $X = (A1+A2+A3) / 3$,计算B1、B2、B3、B4、B5的平均值 $Y = (B1+B2+B3+B4+B5) / 5$,那么X与Y的相似性即可作为组相似性。其中,歌曲片段对应的特征可以是目标模型中的嵌入层的输出结果,即将某一歌曲片段输入目标模型中的嵌入层,即可输出对应的特征。

[0055] S205、根据分类结果和当前聚类结果的损失值更新目标模型的模型参数,得到更新后的目标模型。

[0056] 在一种具体实施方式中,根据分类结果和聚类结果的损失值更新目标模型的模型参数,得到更新后的目标模型,包括:利用目标损失函数计算分类结果和聚类结果的损失值;根据损失值分别优化目标模型中的嵌入层参数和全连接层参数,得到更新后的目标模型。

[0057] 其中,目标模型包括:嵌入层和全连接层;模型参数包括:嵌入层参数和全连接层参数。嵌入层参数即为嵌入层的相关参数。全连接层参数即为全连接层的相关参数。更新目标模型的模型参数实质为:更新嵌入层参数和全连接层参数,使得嵌入层不断学习将歌曲片段转化为相应特征的能力,全连接层不断学习分类所有特征的能力。

[0058] 其中,目标损失函数可以为多分类损失函数、交叉熵损失函数、噪声对比估计(Noise-Contrastive Estimation)或对比损失函数(Contrastive Loss)。各个函数的具体计算过程可参照现有相关技术,本说明书在此不再赘述。

[0059] S206、若更新后的目标模型符合收敛条件,则利用更新后的目标模型中的嵌入层提取待识别歌曲中的各个歌曲片段的待识别特征。

[0060] 在一种具体实施方式中,利用更新后的目标模型中的嵌入层提取待识别歌曲中的各个歌曲片段的待识别特征之后,还包括:利用k-近邻算法、神经网络算法、支持向量机或随机森林算法,结合和少量已知流派的歌曲片段训练得到识别分类器;将待识别特征输入识别分类器,以输出待识别歌曲中的各个歌曲片段的音乐类型。

[0061] 其中,利用识别分类器识别待识别特征的具体过程包括:将待识别特征输入识别分类器,该识别分类器可以确定与待识别特征最相近的已知音乐类型的歌曲片段,此时认为最相近的已知音乐类型的歌曲片段与待识别特征属于同一音乐类型,因此可确定待识别特征的音乐类型。

[0062] 本实施例将无音乐类型标签的歌曲片段作为训练数据,实现了对未被标注的数据的无监督机器学习,在学习过程中自适应学习识别能力。更新目标模型的模型参数,即是对目标模型进行一次迭代训练,具体可参照现有的模型训练过程。

[0063] 可见,本实施例利用无音乐类型标签的歌曲片段集作为训练数据,因此无需对整首歌曲或歌曲片段打音乐类型标签,简化了模型训练前的准备工作,降低了时间成本。并且,符合收敛条件的模型中的嵌入层可以将待识别歌曲中的各个歌曲片段转化为相应待识别特征,然后利用识别分类器即可识别待识别特征对应的音乐类型。可见,本申请可以精准识别待识别歌曲中的各个歌曲片段对应的音乐类型,而不是笼统地对整首歌曲进行音乐类型识别,故本申请能够区分一首歌曲中不同音乐类型的片段,提升了歌曲音乐类型的识别精度和准确度,可适用于识别具有多元化音乐元素的歌曲。

[0064] 请参见图3,图3为本申请实施例提供的第二种模型训练方法流程图。如图3所示,该模型训练方法可以包括以下步骤:

[0065] S301、获取无音乐类型标签的多个歌曲片段形成的歌曲片段集;

[0066] S302、利用目标模型中的嵌入层提取歌曲片段集中的各个歌曲片段的片段特征,以得到片段特征集;

[0067] S303、利用目标模型中的全连接层对片段特征集中的片段特征进行分类,得到分类结果;

[0068] S304、判断聚类算法执行周期是否达到阈值;若是,则执行聚类操作,即执行S305;若否,则不进行聚类,即执行S308;

[0069] S305、利用聚类算法对片段特征集中的片段特征进行聚类,得到当前聚类结果;

[0070] S306、根据分类结果和当前聚类结果的损失值更新目标模型的模型参数,得到更新后的目标模型;

[0071] S307、若更新后的目标模型符合收敛条件,则利用更新后的目标模型中的嵌入层提取待识别歌曲中的各个歌曲片段的待识别特征;

[0072] S308、判断是否存在历史聚类结果;若是,则执行S309;若否,则执行S310;

[0073] S309、根据分类结果和历史聚类结果的损失值更新目标模型的模型参数,得到更新后的目标模型,并执行S307;

[0074] S310、根据分类结果和类别标签的损失值更新目标模型的模型参数,得到更新后的目标模型,用更新后的目标模型替换目标模型,并执行S302。

[0075] 在本实施例中,歌曲片段集中的、属于同一歌曲的、不同歌曲片段标注有不同或相同的类别标签。该类别标签仅用于标注哪些歌曲片段属于同一音乐类型,哪些歌曲片段不属于同一音乐类型,各个类别标签的具体音乐类型仍未知。

[0076] 在本实施例中,聚类结果适时变化,模型训练的参考目标也在适时变化。

[0077] 具体的,聚类结果根据模型能力的提升而改变。首先让模型的嵌入层处理歌曲片段集,以将所有歌曲片段转化为相应特征,然后利用聚类算法对这些特征进行聚类。其中,由于模型中的嵌入层的转化能力在更新过程中会不断提升,因此聚类算法要处理的这些特征的精准度会随之不断提升,因此聚类结果的精准度也会相应随之提升。如此一来,模型训练的参考目标的精准度就会根据模型能力的提升而提升。

[0078] 其中,聚类算法执行周期可以按照迭代周期或时间周期进行预设。例如:聚类算法

执行周期为5,那么每迭代训练五次,则认为聚类算法执行周期达到阈值。也可以设置聚类算法执行周期为20秒,那么每迭代训练20秒,则认为聚类算法执行周期达到阈值。

[0079] 当然,也可以每迭代一次,聚类就执行一次。但需要注意,由于初始化的目标模型的嵌入层提取特征的能不足,因此第一次迭代训练时,不进行聚类操作,第二次迭代时,再开始进行聚类操作。那么第一次迭代时,基于分类结果和类别标签的损失值更新目标模型的模型参数。基于类别标签能够将歌曲片段集中的各个歌曲片段划分为不同类别,因此可认为歌曲片段集中初始化有相应类别,但各个类别的具体音乐类型未知。虽然类别标签不够准确,但其准确度一般大于初始化的目标模型的嵌入层,故第一次迭代时,基于类别标签更新模型参数。

[0080] 当然,也可以忽略初始化的目标模型的嵌入层的不足,在第一次迭代训练时,就进行聚类操作。因为模型参数的更新会逐渐提升模型的能力,因此第一次迭代训练的误差可以暂时忽略。

[0081] 在模型训练过程中,收敛条件可以参照现有模型训练过程中的相关条件,收敛条件具体可以为:分类结果和聚类结果的损失值小于预设阈值,或者该损失值的变化小于某一阈值。

[0082] 本实施例中的其他步骤的具体实现过程可以参考其他实施例中公开的相应内容,在此不再进行赘述。

[0083] 可见,本实施例可以精准识别待识别歌曲中的各个歌曲片段对应的音乐类型,而不是笼统地对整首歌曲进行音乐类型识别,故本申请能够区分一首歌曲中不同音乐类型的片段,提升了歌曲音乐类型的识别精度和准确度,可适用于识别具有多元化音乐元素的歌曲。

[0084] 请参见图4,图4为本申请实施例提供的第三种模型训练方法流程图。如图4所示,该模型训练方法可以包括以下步骤:

[0085] S401、获取无音乐类型标签的多个歌曲片段形成的歌曲片段集;

[0086] S402、利用目标模型中的嵌入层提取歌曲片段集中的各个歌曲片段的片段特征,以得到片段特征集;

[0087] S403、利用目标模型中的全连接层对片段特征集中的片段特征进行分类,得到分类结果;

[0088] S404、若确定需要聚类,则利用聚类算法对片段特征集中的片段特征进行聚类,得到当前聚类结果;

[0089] S405、根据分类结果和当前聚类结果的损失值更新目标模型的模型参数,得到更新后的目标模型;

[0090] S406、判断当前聚类结果与历史聚类结果的差异是否符合预设要求;若是,则执行S407;若否,则执行S409;

[0091] S407、更新符合预设要求的次数;

[0092] S408、判断符合预设要求的次数是否小于预设阈值;若是,则执行S409;若否,则执行S410;

[0093] S409、确定更新后的目标模型不符合收敛条件,用更新后的目标模型替换目标模型,用当前聚类结果替换历史聚类结果,并执行S402;

[0094] S410、确定更新后的目标模型符合收敛条件,利用更新后的目标模型中的嵌入层提取待识别歌曲中的各个歌曲片段的待识别特征。

[0095] 在本实施例中,虽然模型参数的更新是基于分类结果和当前聚类结果的损失值进行,但并不用此损失值来确定模型是否收敛。其中,模型收敛的条件为:当前聚类结果与历史聚类结果的差异符合预设要求(差异越小越好),且符合预设要求的次数不小于预设阈值。例如:连续三次的聚类结果都相同,那么认为模型符合收敛条件。

[0096] 本实施例中的其他步骤的具体实现过程可以参考其他实施例中公开的相应内容,在此不再进行赘述。

[0097] 可见,本实施例可以精准识别待识别歌曲中的各个歌曲片段对应的音乐类型,而不是笼统地对整首歌曲进行音乐类型识别,故本申请能够区分一首歌曲中不同音乐类型的片段,提升了歌曲音乐类型的识别精度和准确度,可适用于识别具有多元化音乐元素的歌曲。

[0098] 下面通过具体的应用场景实例描述,来介绍本申请提供的模型训练方案。

[0099] 请参见图5,图5为本申请实施例提供的一种音乐类型识别方法流程图。如图5所示,该音乐类型识别方法可以包括以下步骤:

[0100] S501、终端上传歌曲至服务器;

[0101] S502、服务器获取训练完成的模型中的嵌入层;

[0102] S503、服务器利用该嵌入层将该歌曲中的各个歌曲片段转化为相应特征;

[0103] S504、服务器利用识别分类器识别各个特征对应的音乐类型;

[0104] S505、服务器发送各个特征对应的音乐类型至终端;

[0105] S506、终端展示各个特征对应的音乐类型。

[0106] 具体的,服务器上可以部署歌曲推荐平台,该平台可基于歌曲识别结果推荐或分类相关歌曲。

[0107] 其中,模型可按照如下三种方式中的任一种训练获得。

[0108] 方式一:采用自上而下的自适应无监督训练方案。

[0109] 具体来说,假设训练数据包含N首歌曲,其中歌曲i包含 S_i 个片段。在训练开始阶段,每一首不同的歌曲都被设为不同的类别,而同一首歌曲的不同片段都被设为同一类别(实际上同一首歌曲中的不同歌曲片段可能属于不同类别,此设定是聚类前的初始化假设),即每一首歌曲包含 $L_i=1$ 个类别,构造了类别数为N的多分类问题。

[0110] 将训练数据依次输入模型(该模型包括深度神经网络和全连接层两个部分),从而输出训练结果。其中,深度神经网络(即上文提及的嵌入层)的输出结果不仅作为全连接层的输入,还作为聚类算法的输入,从而可获得训练数据和聚类结果。聚类后,聚类结果中的每个簇对应一个类别,但由于是无监督训练过程,因此每个簇对应的音乐类型未知,那么歌曲i中的各个歌曲片段分别对应 L_i 个类别($1 \leq L_i \leq S_i$)。之后,可利用损失函数计算训练结果和聚类结果的损失值,从而按照损失值更新模型的模型参数,如此可再次迭代,以迭代更新模型。此后,模型自适应地调整为类别数为 $\sum_i L_i$ 的多类别训练。

[0111] 其中,并非是每次迭代过程都需要进行聚类操作。可以设置重复聚类条件(如每10epoch聚类一次),那么聚类过程可在训练过程中的特定时期重复进行。其中,由于聚类算法的输入是深度神经网络的输出结果,因此每次聚类的输入会变化(因为深度神经网络的

模型参数随模型更新在不断优化, 神经网络对于输入的处理能力会不断提升, 故对于同一输入, 其可能会输出不同准确度的结果), 故聚类结果也会相应变化, 也就实现了自适应调整训练策略。

[0112] 由于此训练过程中, 每首歌曲*i*包含的类别数是由一个类别拆分为多个类别, 因此称为自上而下的训练方案。请参见图6, 该方案可按照如下步骤实施:

[0113] 步骤一: 选定嵌入层和分类层, 并以此构建需要训练的模型;

[0114] 步骤二: 选定大量歌曲里的不同片段作为训练数据, 记为 $(x_{i,j}, y_{i,j})$ 。其中, *i*表示歌曲索引, *j*表示片段索引, $x_{i,j}$ 表示歌曲片段, $y_{i,j}$ 表示歌曲片段对应的类别, $y_{i,j}$ 在训练过程中会自适应的更改。这里, 一首歌曲的不同片段, 数量可以为一个, 也可以为多个; 相互可以重叠, 也可以不重叠; 可以是任意固定长度, 也可以在训练过程中改变。为了实现对短歌曲片段的识别, 歌曲片段可以设置为10s不重叠的片段。

[0115] 步骤三: 设定每一首歌曲为一个独立的类别, 且同一首歌曲的不同片段属于相同类别, 即 $y_{i,j}=y_{i,k}$, $\forall j,k$ 。 $y_{i,j}$ 和 $y_{i,k}$ 分别表示同一首歌曲中的不同片段。

[0116] 步骤四: 将训练数据输入需要训练的模型, 同时对训练数据聚类得到聚类结果, 并利用多分类损失函数计算损失值。

[0117] 步骤五: 训练一段时间后, 利用模型中嵌入层的输出结果再次对训练数据聚类, 得到新的聚类结果。每一首歌曲有可能存在不少于一个簇, 即不少于一个类别。换言之, 对于歌曲*i*, $\exists j,k$, 使得 $y_{i,j} \neq y_{i,k}$, 即: 歌曲*i*中的 $y_{i,j}$ 和 $y_{i,k}$ 属于同一类别; 或者, 对于歌曲*i*, $\exists j,k$, 使得 $y_{i,j}=y_{i,k}$, 即: 歌曲*i*中的 $y_{i,j}$ 和 $y_{i,k}$ 属于不同类别。

[0118] 步骤六: 若模型收敛, 则将待识别歌曲输入当前模型的嵌入层, 以输出该歌曲中的各个歌曲片段对应的待识别特征; 同时将一些已知音乐类型的歌曲片段也输入当前模型的嵌入层, 以输出已知音乐类型的特征; 然后利用k-近邻算法、神经网络算法、支持向量机或随机深林算法处理待识别特征和这些已知音乐类型的特征, 以确定待识别特征对应的音乐类型。

[0119] 其中, 可以利用k-近邻算法、神经网络算法、支持向量机或随机深林算法预先训练分类器, 后续将待识别特征和已知音乐类型的特征输入该分类器, 即可输出待识别特征对应的音乐类型。

[0120] 方式二: 采用自下而上的自适应无监督训练方案。

[0121] 具体来说, 假设训练数据包含*N*首歌曲, 其中歌曲*i*包含 S_i 个片段。在训练开始阶段, 每一首歌曲的不同片段被设为不同类别(当然, 一首歌曲的不同片段也可能属于同一类别, 此设定是聚类前的初始化假设), 即歌曲*i*包含 $L_i=S_i$ 个类别, 构造了类别数为 $\sum_i S_i=P$ 的多分类问题。

[0122] 将训练数据依次输入模型(该模型包括深度神经网络和全连接层两个部分), 从而输出第一次训练结果。其中, 深度神经网络(即上文提及的嵌入层)的输出结果不仅作为全连接层的输入, 还作为聚类算法的输入, 从而可获得训练数据的第一次聚类结果。聚类后, 聚类结果中的每个簇对应一个类别, 但由于是无监督训练过程, 因此每个簇对应的音乐类型未知, 那么歌曲*i*中的各个歌曲片段分别对应 L_i 个类别($1 \leq L_i \leq S_i$)。之后, 可利用损失函数计算第一次训练结果和第一次聚类结果的损失值, 从而按照损失值更新模型的模型参

数,如此可再次迭代,以迭代更新模型。此后,模型自适应地调整为类别数为 $\sum_i L_i$ 的多类别训练。

[0123] 由于此训练过程中,歌曲*i*包含的类别数由 $L_i = S_i$ 开始减小,即由多个类别聚合成少量类别,因此称为自下而上的训练方案,该方案的具体实现步骤可参照自上而下的训练方案,本说明书在此不再赘述。

[0124] 方式三:对训练数据先按照相似性分类,在对分类结果中的每个类进行聚类。模型训练过程以及用模型识别音乐类型的过程可参照上述两种方式,在此不再赘述。

[0125] 具体来说,在聚类前,先计算每一首歌里所有歌曲片段的特征平均值,作为该歌曲的歌曲特征值。计算不同歌曲特征值的相似性,将相似性大于预先设定的阈值的歌曲相连接。最后,对将相连接的歌曲里不同歌曲片段进行聚类,每个簇对应一个类别。此方式可以降低聚类的复杂度。

[0126] 其中,相似性可以利用余弦夹角、欧式距离等计算。

[0127] 训练好的模型可以识别任意歌曲中的不同片段所属的音乐类型,以便向用户推荐用户喜好的歌曲。本方案可基于中央处理器(cpu)和图形处理器(gpu)实现。本实施例可以发现属于不同音乐类型的两个歌曲中的相似片段,以及属于同一音乐类型的两个歌曲中的不同片段。同时,省去了人工打音乐类型标签过程,节省了收集训练数据的代价。

[0128] 请参见图7,图7为本申请实施例提供的一种模型训练装置示意图,包括:

[0129] 获取模块701,用于获取无音乐类型标签的多个歌曲片段形成的歌曲片段集;

[0130] 提取模块702,用于利用目标模型中的嵌入层提取所述歌曲片段集中的各个歌曲片段的片段特征,以得到片段特征集;

[0131] 分类模块703,用于利用所述目标模型中的全连接层对所述片段特征集中的片段特征进行分类,得到分类结果;

[0132] 聚类模块704,用于若确定需要聚类,则利用聚类算法对所述片段特征集中的片段特征进行聚类,得到当前聚类结果;

[0133] 更新模块705,用于根据所述分类结果和所述当前聚类结果的损失值更新所述目标模型的模型参数,得到更新后的目标模型;

[0134] 应用模块706,用于若所述更新后的目标模型符合收敛条件,则利用所述更新后的目标模型中的嵌入层提取待识别歌曲中的各个歌曲片段的待识别特征。

[0135] 在一种具体实施方式中,还包括:

[0136] 聚类算法执行周期判断模块,用于判断聚类算法执行周期是否达到阈值;若聚类算法执行周期达到阈值,则执行聚类操作;若聚类算法执行周期未达到阈值,则不进行聚类。

[0137] 在一种具体实施方式中,还包括:

[0138] 聚类结果判断模块,用于若确定存在历史聚类结果,则根据分类结果和历史聚类结果的损失值更新目标模型的模型参数,得到更新后的目标模型。

[0139] 在一种具体实施方式中,聚类结果判断模块还用于:

[0140] 若确定不存在历史聚类结果,则根据分类结果和类别标签的损失值更新目标模型的模型参数,得到更新后的目标模型,用更新后的目标模型替换目标模型,并循环进行模型训练;其中,歌曲片段集中的、属于同一歌曲的、不同歌曲片段标注有不同或相同的类别标

签。

[0141] 在一种具体实施方式中,还包括:

[0142] 模型收敛判断模块,用于若确定当前聚类结果与历史聚类结果的差异符合预设要求,则更新符合预设要求的次数,若次数不小于预设阈值,则确定更新后的目标模型符合收敛条件。

[0143] 在一种具体实施方式中,模型收敛判断模块还用于:

[0144] 若确定差异不符合预设要求,或次数小于预设阈值,则确定更新后的目标模型不符合收敛条件,用更新后的目标模型替换目标模型,用当前聚类结果替换历史聚类结果,并循环进行模型训练。

[0145] 在一种具体实施方式中,更新模块包括:

[0146] 计算单元,用于利用目标损失函数计算分类结果和聚类结果的损失值;

[0147] 优化单元,用于根据损失值分别优化目标模型中的嵌入层参数和全连接层参数,得到更新后的目标模型。

[0148] 在一种具体实施方式中,所述聚类模块包括:

[0149] 第一划分单元,用于根据不同歌曲之间的相似性将片段特征集划分为多个相似特征子集;

[0150] 第一聚类单元,用于分别对多个相似特征子集中的片段特征进行聚类,以得到每个相似特征子集对应的第一聚类子结果,将所有第一聚类子结果确定为当前聚类结果。

[0151] 在一种具体实施方式中,第一划分单元包括:

[0152] 提取子单元,用于从片段特征集中提取第一歌曲对应的第一特征组,以及第二歌曲对应的第二特征组;第一歌曲和第二歌曲为片段特征集对应的各个歌曲中的任两首;

[0153] 划分子单元,用于确定第一特征组和第二特征组的组相似性,若组相似性大于预设阈值,则将第一特征组和第二特征组划分至同一相似特征子集;否则,将第一特征组和第二特征组划分至不同相似特征子集。

[0154] 在一种具体实施方式中,划分子单元具体用于:

[0155] 计算第一特征组中的所有特征的第一平均特征值,以及第二特征组中的所有特征的第二平均特征值;计算第一平均特征值和第二平均特征值的特征相似性,将特征相似性确定为组相似性。

[0156] 在一种具体实施方式中,所述聚类模块包括:

[0157] 第二划分单元,用于按照不同歌曲将片段特征集划分为多个歌曲特征子集;

[0158] 第二聚类单元,用于分别对多个歌曲特征子集中的片段特征进行聚类,以得到每个歌曲特征子集对应的第二聚类子结果,将所有第二聚类子结果确定为当前聚类结果。

[0159] 在一种具体实施方式中,还包括:

[0160] 识别模块,用于利用k-近邻算法、神经网络算法、支持向量机或随机森林算法训练得到识别分类器;将所述待识别特征输入所述识别分类器,以输出所述待识别歌曲中的各个歌曲片段的音乐类型。

[0161] 其中,关于本实施例中各个模块、单元更加具体的工作过程可以参考前述实施例中公开的相应内容,在此不再进行赘述。

[0162] 可见,本实施例提供了一种模型训练装置,该装置可以精准识别待识别歌曲中的

各个歌曲片段对应的音乐类型,而不是笼统地对整首歌曲进行音乐类型识别,故本申请能够区分一首歌曲中不同音乐类型的片段,提升了歌曲音乐类型的识别精度和准确度,可适用于识别具有多元化音乐元素的歌曲。

[0163] 进一步的,本申请实施例还提供了一种电子设备。其中,上述电子设备既可以是如图8所示的服务器50,也可以是如图9所示的终端60。图8和图9均是根据一示例性实施例示出的电子设备结构图,图中的内容不能被认为是对本申请的使用范围的任何限制。

[0164] 图8为本申请实施例提供的一种服务器的结构示意图。该服务器50,具体可以包括:至少一个处理器51、至少一个存储器52、电源53、通信接口54、输入输出接口55和通信总线56。其中,所述存储器52用于存储计算机程序,所述计算机程序由所述处理器51加载并执行,以实现前述任一实施例公开的模型训练中的相关步骤。

[0165] 本实施例中,电源53用于为服务器50上的各硬件设备提供工作电压;通信接口54能够为服务器50创建与外界设备之间的数据传输通道,其所遵循的通信协议是能够适用于本申请技术方案的任意通信协议,在此不对其进行具体限定;输入输出接口55,用于获取外界输入数据或向外界输出数据,其具体的接口类型可以根据具体应用需要进行选取,在此不进行具体限定。

[0166] 另外,存储器52作为资源存储的载体,可以是只读存储器、随机存储器、磁盘或者光盘等,其上所存储的资源包括操作系统521、计算机程序522及数据523等,存储方式可以是短暂存储或者永久存储。

[0167] 其中,操作系统521用于管理与控制服务器50上的各硬件设备以及计算机程序522,以实现处理器51对存储器52中数据523的运算与处理,其可以是Windows Server、Netware、Unix、Linux等。计算机程序522除了包括能够用于完成前述任一实施例公开的模型训练方法的计算机程序之外,还可以进一步包括能够用于完成其他特定工作的计算机程序。数据523除了可以包括歌曲片段集等数据外,还可以包括应用程序的开发商信息等数据。

[0168] 图9为本申请实施例提供的一种终端的结构示意图,该终端60具体可以包括但不限于智能手机、平板电脑、笔记本电脑或台式电脑等。

[0169] 通常,本实施例中的终端60包括有:处理器61和存储器62。

[0170] 其中,处理器61可以包括一个或多个处理核心,比如4核心处理器、8核心处理器等。处理器61可以采用DSP(Digital Signal Processing,数字信号处理)、FPGA(Field-Programmable Gate Array,现场可编程门阵列)、PLA(Programmable Logic Array,可编程逻辑阵列)中的至少一种硬件形式来实现。处理器61也可以包括主处理器和协处理器,主处理器是用于对在唤醒状态下的数据进行处理的处理单元,也称CPU(Central Processing Unit,中央处理器);协处理器是用于对在待机状态下的数据进行处理的低功耗处理器。在一些实施例中,处理器61可以在集成有GPU(Graphics Processing Unit,图像处理器),GPU用于负责显示屏所需要显示的内容的渲染和绘制。一些实施例中,处理器61还可以包括AI(Artificial Intelligence,人工智能)处理器,该AI处理器用于处理有关机器学习的计算操作。

[0171] 存储器62可以包括一个或多个计算机可读存储介质,该计算机可读存储介质可以是非暂态的。存储器62还可包括高速随机存取存储器,以及非易失性存储器,比如一个或多

个磁盘存储设备、闪存存储设备。本实施例中,存储器62至少用于存储以下计算机程序621,其中,该计算机程序被处理器61加载并执行之后,能够实现前述任一实施例公开的由终端侧执行的模型训练方法中的相关步骤。另外,存储器62所存储的资源还可以包括操作系统622和数据623等,存储方式可以是短暂存储或者永久存储。其中,操作系统622可以包括Windows、Unix、Linux等。数据623可以包括但不限于歌曲信息。

[0172] 在一些实施例中,终端60还可包括有显示屏63、输入输出接口64、通信接口65、传感器66、电源67以及通信总线68。

[0173] 本领域技术人员可以理解,图9中示出的结构并不构成对终端60的限定,可以包括比图示更多或更少的组件。

[0174] 进一步的,本申请实施例还公开了一种存储介质,所述存储介质中存储有计算机可执行指令,所述计算机可执行指令被处理器加载并执行时,实现前述任一实施例公开的模型训练方法。关于该方法的具体步骤可以参考前述实施例中公开的相应内容,在此不再进行赘述。

[0175] 需要指出的是,上述仅为本申请的较佳实施例,并不用以限制本申请,凡在本申请的精神和原则之内,所作的任何修改、等同替换、改进等,均应包含在本申请的保护范围之内。

[0176] 本说明书中各个实施例采用递进的方式描述,每个实施例重点说明的都是与其它实施例的不同之处,各个实施例之间相同或相似部分互相参见即可。对于实施例公开的装置而言,由于其与实施例公开的方法相对应,所以描述的比较简单,相关之处参见方法部分说明即可。

[0177] 本文中应用了具体个例对本申请的原理及实施方式进行了阐述,以上实施例的说明只是用于帮助理解本申请的方法及其核心思想;同时,对于本领域的一般技术人员,依据本申请的思想,在具体实施方式及应用范围上均会有改变之处,综上所述,本说明书内容不应理解为对本申请的限制。

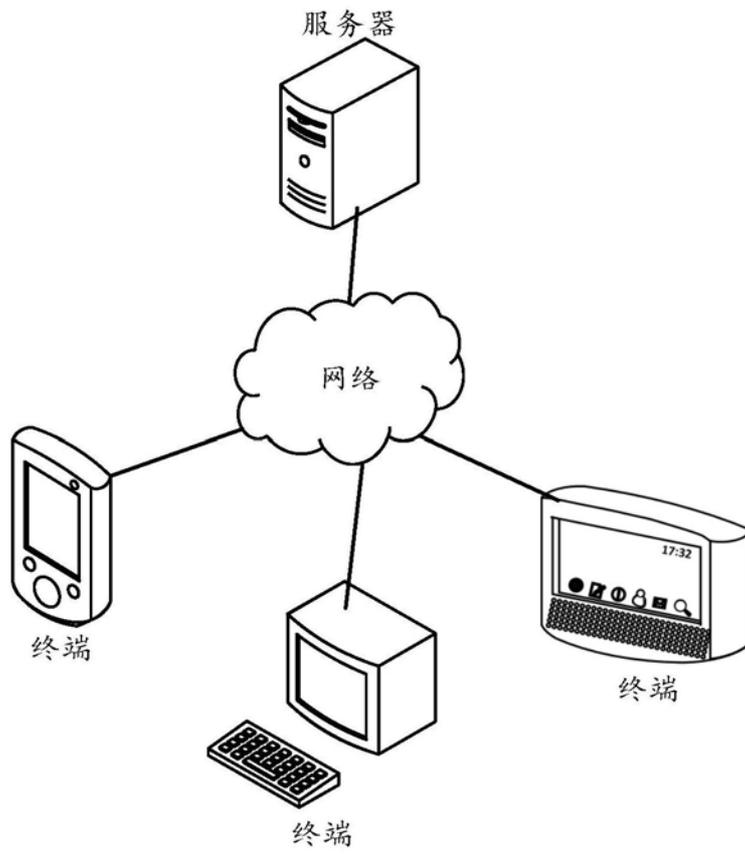


图1

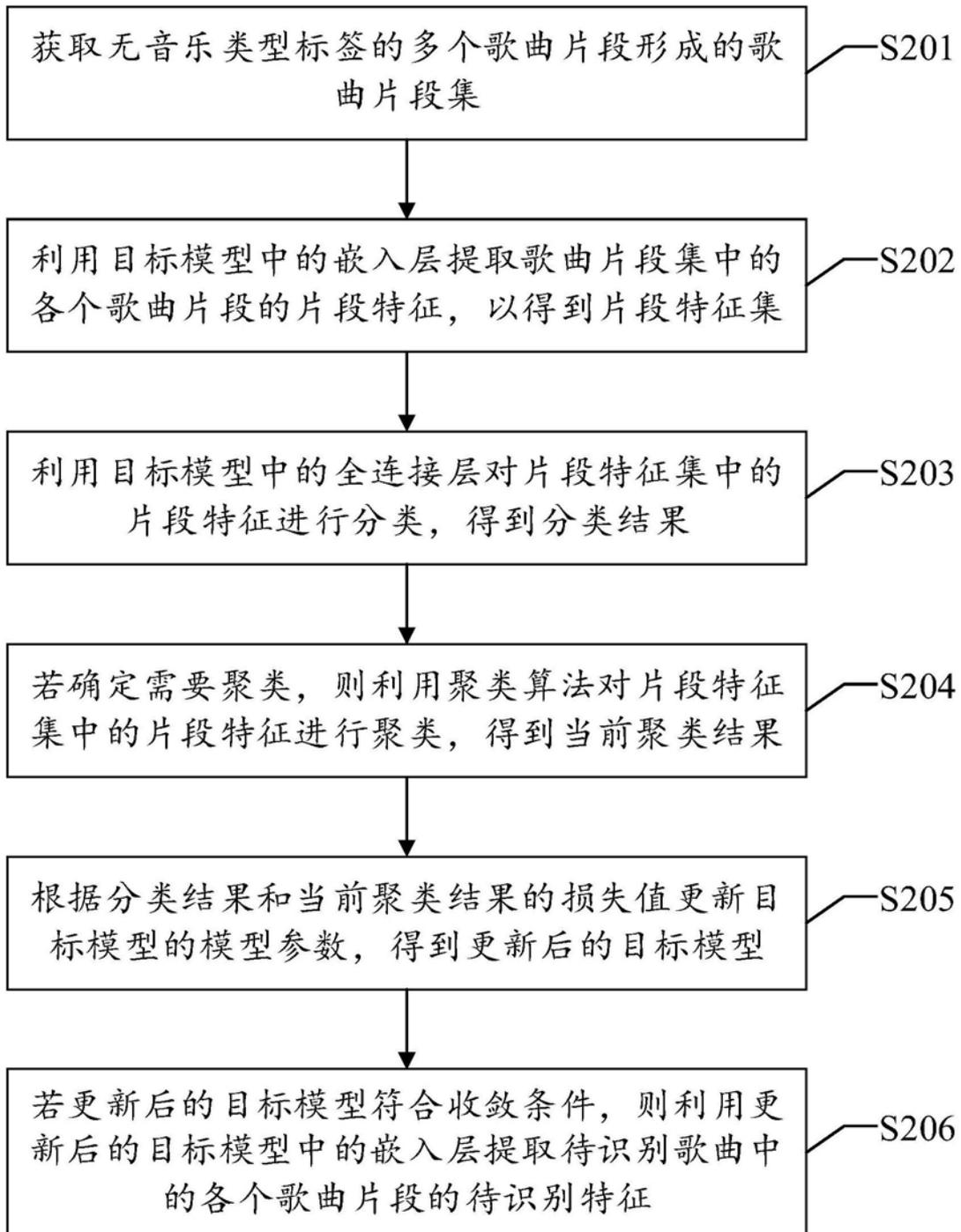


图2

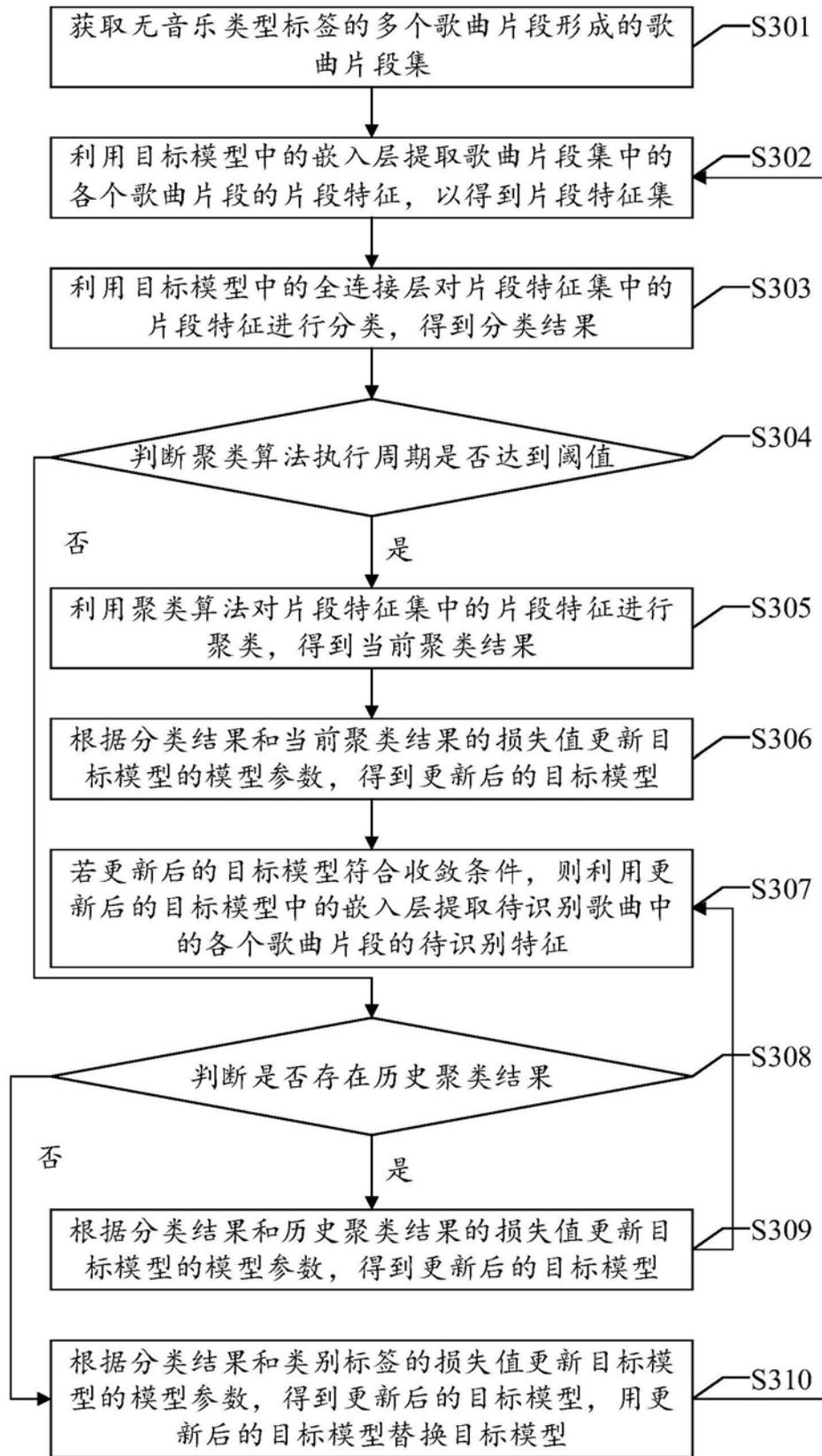


图3

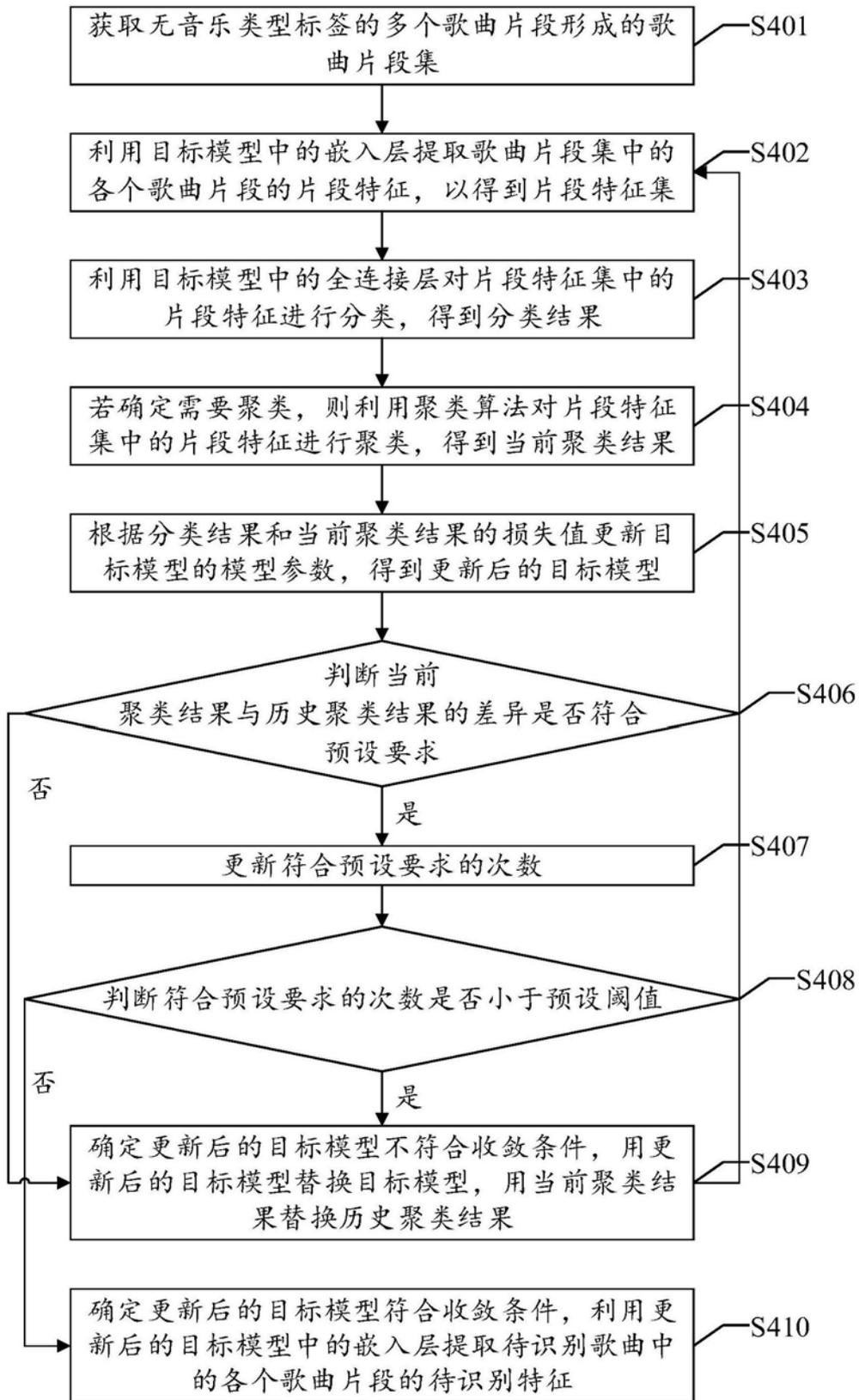


图4

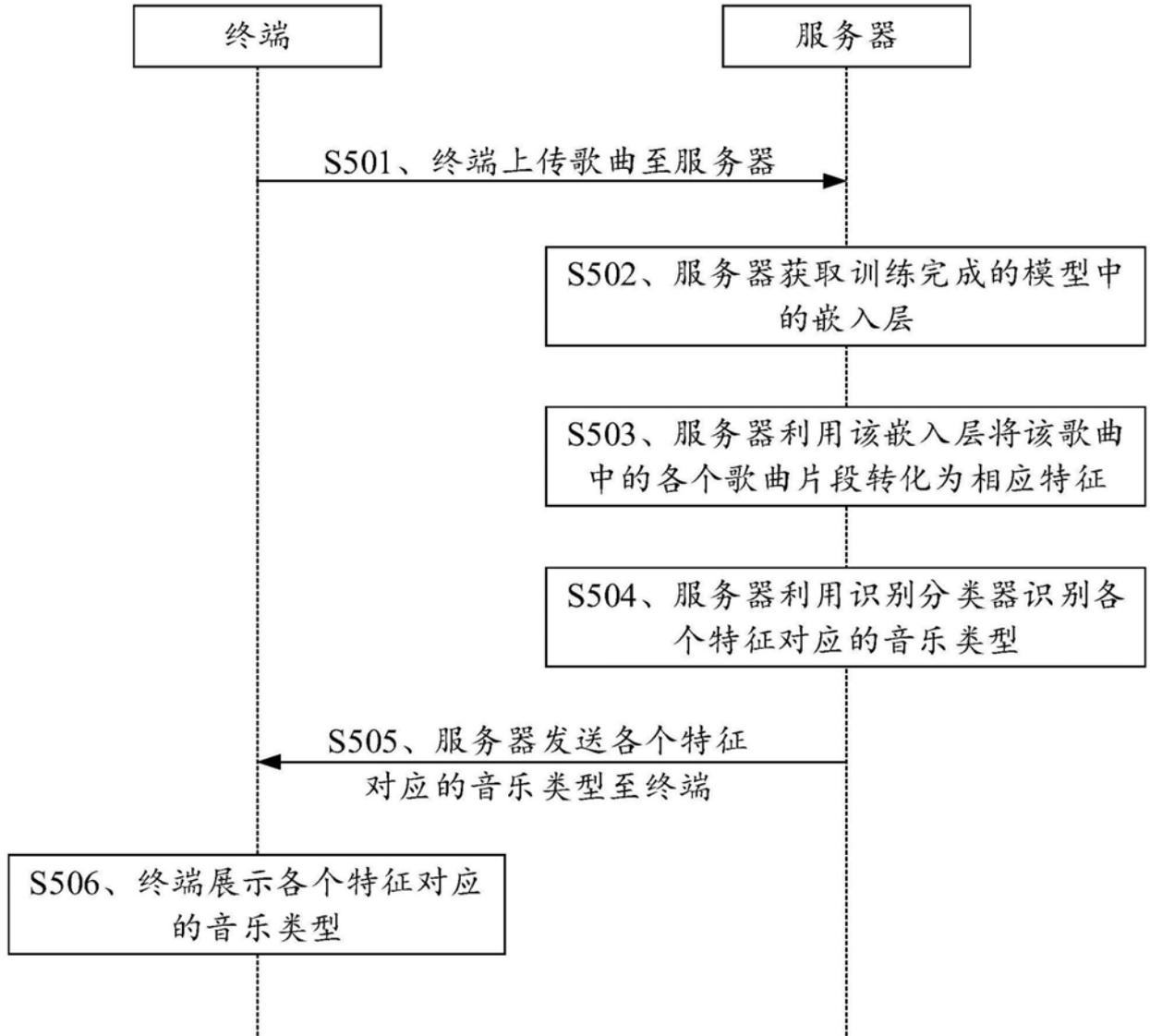


图5

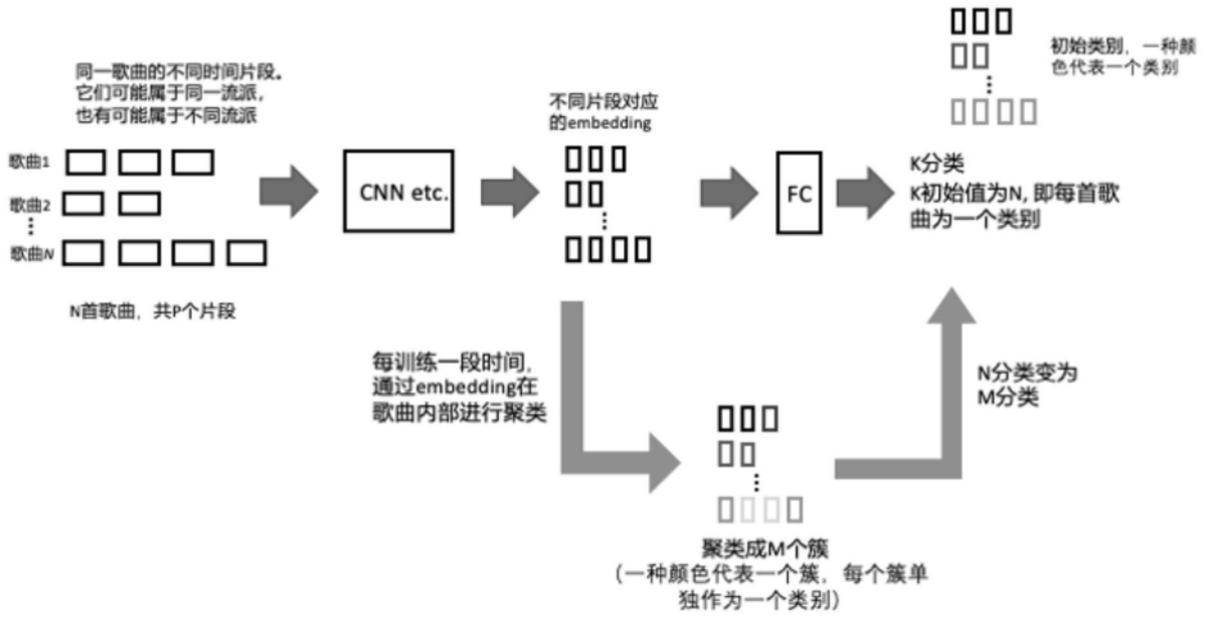


图6

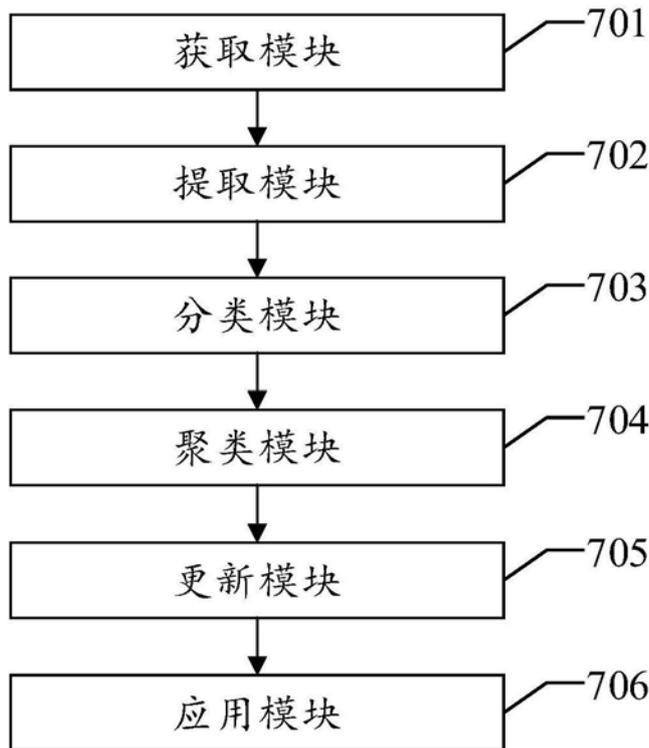


图7

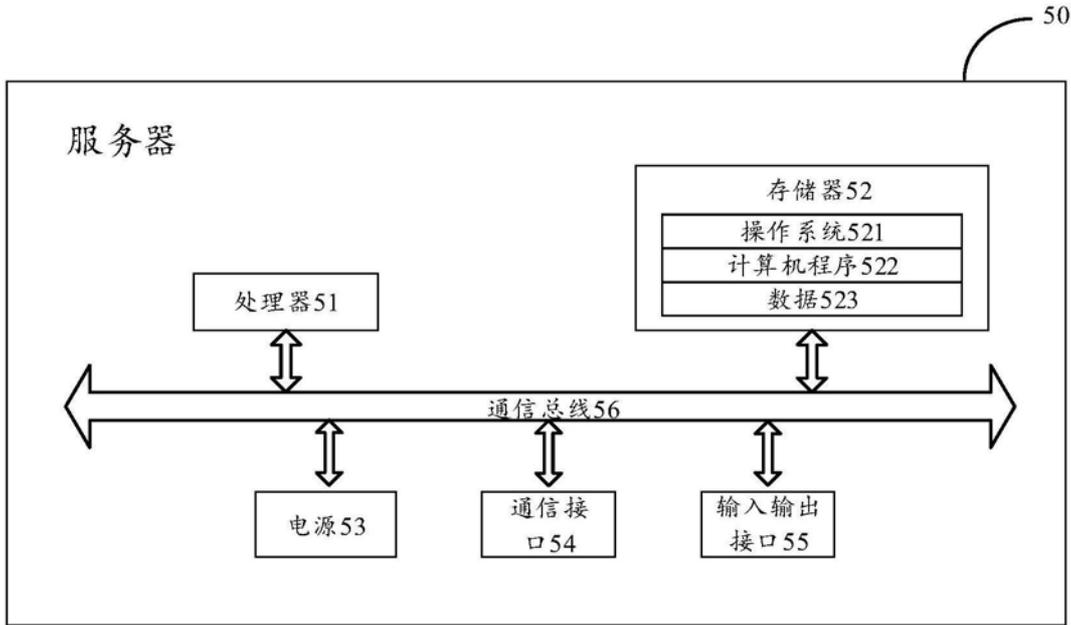


图8

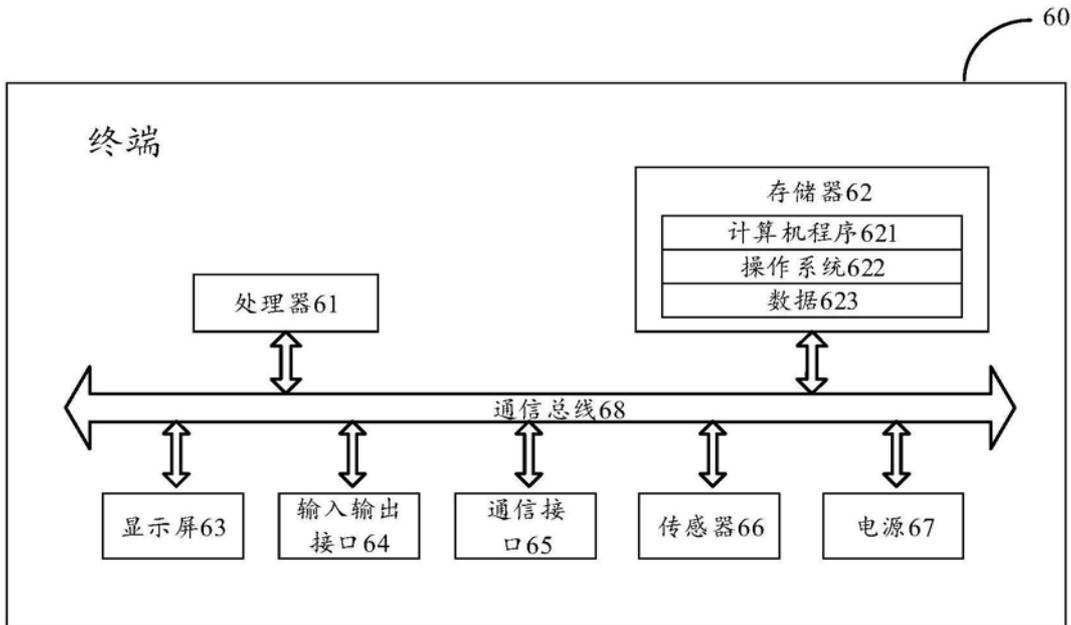


图9