(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 117581233 A
(43)申请公布日 2024.02.20

(21)申请号 202280044627.X

(22)申请日 2022.02.04

(30)优先权数据
17/366,751 2021.07.02 US

(85)PCT国际申请进入国家阶段日
2023.12.22

(86)PCT国际申请的申请数据
PCT/JP2022/006283 2022.02.04

(87)PCT国际申请的公布数据
WO2023/276251 EN 2023.01.05

(71)申请人 三菱电机株式会社
地址 日本东京都

(72)发明人 N·莫里茨 堀贵明 J·勒鲁克斯

(74)专利代理机构 北京三友知识产权代理有限公司 11127
专利代理师 叶朝君 原宏宇

(51)Int.Cl.
G06N 3/0455(2023.01)
G06N 3/044(2023.01)
G06N 3/0442(2023.01)
G06N 3/047(2023.01)
G06N 3/08(2023.01)
G06N 3/063(2023.01)

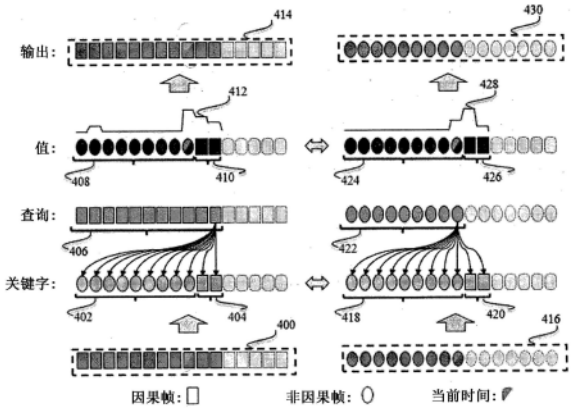
权利要求书3页 说明书14页 附图12页

(54)发明名称

适用于流应用的以双因果和非因果受限自注意力进行序列到序列处理的人工智能系统

(57)摘要

本公开提供了一种适用于流应用的以注意力进行序列到序列建模的人工智能(AI)系统。该AI系统包括:至少一个处理器;以及存储有指令的存储器,所述指令在由处理器执行时使得AI系统通过深度神经网络(DNN)的层处理输入帧序列中的各个输入帧,以生成输出序列。DNN的至少一些层包括具有关注非因果帧和因果帧的双非因果和因果架构的双自注意力模块。此外,该AI系统渲染输出序列。



CN 117581233 A

1.一种适用于数据流应用的人工智能(AI)系统,该AI系统包括:至少一个处理器;以及存储有指令的存储器,所述指令在由所述至少一个处理器执行时使得所述AI系统:

通过深度神经网络(DNN)的层来处理输入帧序列中的各个输入帧以生成输出序列,其中,所述层中的至少一些包括具有关注非因果帧和因果帧的双因果和非因果架构的双自注意力模块;并且

渲染所述输出序列。

2.根据权利要求1所述的AI系统,其中,解释当前帧所关注的所述因果帧是前瞻帧,而所关注的其余帧是所述非因果帧。

3.根据权利要求2所述的AI系统,其中,所述双自注意力模块包括输出在没有前瞻的情况下推导的因果输出帧序列和相对于表示当前时间的输入帧以预定大小的前瞻推导的非因果输出帧序列的双自注意力;并且其中,所述输出序列包括所述因果输出帧和所述非因果输出帧之一或组合。

4.根据权利要求3所述的AI系统,其中,所述输出序列仅包括所述非因果输出帧序列。

5.根据权利要求3所述的AI系统,

其中,所述DNN的当前层的所述双自注意力的因果架构将通过所述DNN的先前层的所述双自注意力的所述因果架构确定的先前因果输出变换为因果关键字帧、因果值帧和因果查询帧的序列,以通过使所述因果查询帧关注所述因果关键字帧以对所述因果值帧进行加权来生成当前因果输出,并且

其中,所述DNN的所述当前层的所述双自注意力的非因果架构将通过所述DNN的所述先前层的所述双自注意力的所述非因果架构确定的先前非因果输出变换为非因果关键字帧、非因果值帧和非因果查询帧的序列,以通过使所述非因果查询帧关注与从所述因果架构复制的因果前瞻关键字帧组合的所述非因果关键字帧以对与从所述因果架构复制的因果前瞻值帧组合的所述非因果值帧进行加权来生成当前非因果输出。

6.根据权利要求3所述的AI系统,

其中,所述DNN的当前层的所述双自注意力的因果架构将通过所述DNN的先前层的所述双自注意力的所述因果架构确定的先前因果输出变换为因果关键字帧、因果值帧和因果查询帧的序列,

其中,所述DNN的所述当前层的所述双自注意力的非因果架构将通过所述DNN的所述先前层的所述双自注意力的所述非因果架构确定的先前非因果输出变换为非因果关键字帧、非因果值帧和非因果查询帧的序列,

其中,所述因果架构通过使所述因果查询帧关注与从所述非因果架构复制的非因果关键字帧组合的因果关键字帧以对与从所述非因果架构复制的非因果值帧组合的所述因果值帧进行加权来生成当前因果输出;并且

其中,所述非因果架构通过使所述非因果查询帧关注与从所述因果架构复制的因果前瞻关键字帧组合的过去和当前非因果关键字帧以对与从所述因果架构复制的因果前瞻值帧组合的非因果过去和当前值帧进行加权来生成当前非因果输出。

7.一种包括根据权利要求3所述的AI系统的语音识别系统,该语音识别系统包括:

编码器,该编码器使用具有所述双自注意力的所述DNN对指示语音输入的数据进行编码,使得所述AI系统的所述输出序列包括所述语音输入的编码序列;以及

解码器,该解码器从所述输出序列对所述语音输入的识别进行解码。

8.根据权利要求7所述的语音识别系统,该语音识别系统还包括:

对齐解码器,该对齐解码器被配置为识别对识别输出进行编码的所述编码序列中的编码,并且响应于识别出对识别输出进行编码的各个编码,触发所述解码器对所识别的编码中的识别输出进行解码。

9.根据权利要求8所述的语音识别系统,其中,所述语音输入的识别是所述语音输入的转录。

10.一种包括根据权利要求3所述的AI系统的声学事件检测系统,该语音识别系统包括:

编码器,该编码器使用具有所述双自注意力的所述DNN对指示声学信号的数据进行编码,使得所述AI系统的所述输出序列包括所述声学信号的编码序列;以及

解码器,该解码器从所述输出序列对声学事件进行解码。

11.一种音频处理方法,该音频处理方法包括以下步骤:

通过深度神经网络(DNN)的层来处理输入帧序列中的各个输入帧以生成输出序列,其中,所述层中的至少一些包括具有关注非因果帧和因果帧的双因果和非因果架构的双自注意力模块;以及

渲染所述输出序列。

12.根据权利要求11所述的音频处理方法,其中,解释当前帧所关注的所述因果帧是前瞻帧,而所关注的其余帧是所述非因果帧。

13.根据权利要求12所述的音频处理方法,其中,所述双自注意力模块包括输出在没有前瞻的情况下推导的因果输出帧序列和相对于表示当前时间的输入帧以预定大小的前瞻推导的非因果输出帧序列的双自注意力;并且其中,所述输出序列包括所述因果输出帧序列和所述非因果输出帧序列之一或组合。

14.根据权利要求13所述的音频处理方法,其中,所述输出序列仅包括所述非因果输出帧序列。

15.根据权利要求13所述的音频处理方法,

其中,所述DNN的当前层的所述双自注意力的因果架构将通过所述DNN的先前层的所述双自注意力的所述因果架构确定的先前因果输出变换为因果关键字帧、因果值帧和因果查询帧的序列,以通过使所述因果查询帧关注所述因果关键字帧以对所述因果值帧进行加权来生成当前因果输出,并且

其中,所述DNN的所述当前层的所述双自注意力的非因果架构将通过所述DNN的所述先前层的所述双自注意力的所述非因果架构确定的先前非因果输出变换为非因果关键字帧、非因果值帧和非因果查询帧的序列,以通过使所述非因果查询帧关注与从所述因果架构复制的因果前瞻关键字帧组合的所述非因果关键字帧以对与从所述因果架构复制的因果前瞻值帧组合的所述非因果值帧进行加权来生成当前非因果输出。

16.根据权利要求13所述的音频处理方法,

其中,所述DNN的当前层的所述双自注意力的因果架构将通过所述DNN的先前层的所述双自注意力的所述因果架构确定的先前因果输出变换为因果关键字帧、因果值帧和因果查询帧的序列,

其中,所述DNN的所述当前层的所述双自注意力的非因果架构将通过所述DNN的所述先前层的所述双自注意力的所述非因果架构确定的先前非因果输出变换为非因果关键字帧、非因果值帧和非因果查询帧的序列,

其中,所述因果架构通过使所述因果查询帧关注与从所述非因果架构复制的非因果关键字帧组合的因果关键字帧以对与从所述非因果架构复制的非因果值帧组合的所述因果值帧进行加权来生成当前因果输出,并且

其中,所述非因果架构通过使所述非因果查询帧关注与从所述因果架构复制的因果前瞻关键字帧组合的过去和当前非因果关键字帧以对与从所述因果架构复制的因果前瞻值帧组合的非因果过去和当前值帧进行加权来生成当前非因果输出。

17.一种具体实现有程序的非暂时性计算机可读存储介质,所述程序能够由处理器执行以用于执行一种方法,该方法包括以下步骤:

通过深度神经网络(DNN)的层来处理输入帧序列中的各个输入帧以生成输出序列,其中,所述层中的至少一些包括具有关注非因果帧和因果帧的双因果和非因果架构的双自注意力模块;以及

渲染所述输出序列。

适用于流应用的以双因果和非因果受限自注意力进行序列到序列处理的人工智能系统

技术领域

[0001] 本公开总体上涉及人工智能(AI),更具体地,涉及一种适用于流应用的用于以注意力进行序列到序列建模的AI系统。

背景技术

[0002] 如今,自注意力机制已成为用于求解序列到序列问题的不同人工智能(AI)应用的许多神经网络(NN)架构中的核心组件。序列到序列问题的示例包括机器翻译、语音处理、语言建模、自动语音识别(ASR)、计算机视觉等。自注意力机制允许输入彼此交互(“自”)并且找出它们应该更注意谁(“注意力”),以针对给定任务以最佳方式计算输出。当整个输入序列可用于处理时,自注意力机制有利于求解序列到序列问题。然而,等待从时间相关信号推导的整个输入序列导致以自注意力模块来处理时间序列的神经网络具有输出延迟,这使得它们不太适用于需要低延迟的在线或流序列到序列处理。

[0003] 例如,在语音处理应用中,为了解释语音话语“she eats a green apple(她吃了一个青苹果)”,自注意力机制对于解释语音话语的不同部分可能投入不同的注意力。例如,对于解释词语“eats”,词语“apple”可能比词语“green”更重要。因此,在解释词语“eats”期间,自注意力模块可学习对词语“apple”而非对词语“green”投入更多注意力。然而,为了对词语“apple”投入更高注意力,自注意力模块应该等待直至词语“apple”变得可用于处理。因此,在此示例中,为了解释句子的开头,自注意力模块需要等待直至句子的结束,从而导致处理延迟。

[0004] 受限自注意力可部分地解决自注意力机制的这一缺点。尽管自注意力机制处理整个输入序列以解释其各个输入,受限自注意力通过查看当前输入前面的有限视域来解释输入序列的当前输入。通常,输入序列包括输入帧序列,因此为了解释当前帧,受限自注意力仅考虑当前帧之后的有限数量的帧。当前帧之后的帧在本文中被称为前瞻帧或仅前瞻,前瞻帧的数量在本文中被称为前瞻大小。

[0005] 受限自注意力可能使自注意力模块的性能下降,但是降低了序列到序列处理的延迟。例如,当前瞻大小为两个帧时,受限自注意力将各个当前帧的解释仅延迟两个帧,这对于流应用而言是可容忍的。

[0006] 然而,在实践中,自注意力模块在深度神经网络的结构内形成复杂架构,其中在彼此层叠的层内使用多个自注意力模块,各个层使用在前层的输出作为输入。深度神经网络的结构在准确性方面改进了序列到序列处理的性能,但这是以增加的处理延迟为代价的。例如,当受限自注意力模块引入两个帧的处理延迟并且深度神经网络包括12个连续的基于自注意力的层以改进处理的准确性时,尽管各个自注意力层仅使用2个前瞻帧,但总延迟变为24帧。结果,深度神经网络的设计者需要平衡处理的准确性与延迟。还需要与DNN的总延迟相比平衡各个自注意力层的前瞻大小,以提供令人满意的序列到序列处理性能而无需大的总延迟。

发明内容

[0007] 技术问题

[0008] 一些实施方式的目的是提供一种适用于流应用的以自注意力进行序列到序列处理的人工智能(AI)系统。另外地或另选地,一些实施方式的目的是提供一种双自注意力模块,其可提供性能保证,而无需序列到序列处理的大处理延迟。另外地或另选地,一些实施方式的目的是提供一种双自注意力模块,其可用于形成各种架构的深度神经网络而不会累积各个层的处理延迟。

[0009] 问题的解决方案

[0010] 一些实施方式基于这样的理解:自注意力以及受限自注意力具有非因果架构。词语因果和非因果是信号处理领域的术语,在本文中以其平常含义使用。例如,在信号处理中,因果滤波器是线性时不变因果系统,其中词语"因果"指示滤波器输出仅取决于过去和现在的输入。输出也取决于未来输入的滤波器是非因果的。因此,如本文所使用的,自注意力模块的非因果架构使用过去、当前以及在当前输入帧之后到来的未来输入来解释当前输入帧。

[0011] 类似地,受限自注意力模块可使用因果架构来实现。这种架构不使用未来信息,因此不引入任何延迟。结果,因果自注意力的任何组合不会导致任何附加延迟。然而,使用因果自注意力的序列到序列处理的性能比非因果自注意力的性能差。为此,一些实施方式的目的是将因果架构与非因果架构组合以用于数据流应用,以利用二者的优点。

[0012] 一些实施方式基于这样的认识:非因果架构使用两个主要原理。第一个原理是关注输入帧的方式,第二个原理是输入帧的本质。在非因果架构中,两个原理均是非因果的。具体地,非因果自注意力关注未来前瞻帧,因此关注的方式是非因果的。另外,第二个原理规定前瞻帧应该考虑未来数据根据非因果原理来生成。因此,生成前瞻帧的方式也可为非因果的。结果,根据两个非因果原理,前瞻大小为两个帧的非因果架构需要关注两个未来前瞻帧,这两个未来前瞻帧继而可利用另两个未来前瞻帧生成,这另两个未来前瞻帧继而可利用另两个前瞻帧生成,依此类推,遵循使用自注意力模块的神经网络的深度。

[0013] 一些实施方式基于这样的认识:可通过提供双因果和非因果(DCN)架构来修改非因果架构的原理。在DCN架构中,关注输入帧的方式仍是非因果的,然而,输入帧本身可以是因果或非因果的。另外地或另选地,在DCN架构中,关注输入帧的方式可以是因果或非因果的,并且输入帧可以是因果或非因果的。如本文所述,因果输入或因果帧是在不考虑一个或多个未来输入的情况下生成的输入或帧,而非因果输入或非因果帧是考虑一个或多个未来输入生成的输入或帧。结果,深度神经网络(DNN)的非因果架构内的因果输入可用于降低总处理延迟,而准确性仅部分降低。

[0014] 在一些实施方式中,非因果自注意力处理的前瞻帧是因果帧,而所关注的其它帧对应于非因果帧。当前瞻帧为因果的时,在前双自注意力模块不需要附加前瞻帧来生成它们。值得注意的是,如果非因果过去帧的前瞻大小不长于其从当前帧过去的距离,则具有非因果过去帧不会引入任何附加延迟。因此,实施方式可使用尽可能多的非因果帧,从而限制性能降低,同时仅使用因果前瞻帧,从而限制处理延迟。

[0015] 一些实施方式基于这样的理解:需要以与非因果帧一致的方式生成因果帧(例如因果前瞻帧),从而允许DCN架构一起使用因果帧和非因果帧。为此,一些实施方式使用双自

注意力,维持因果架构和非因果架构二者彼此并行操作。因果架构以因果方式处理其输入,从而在没有前瞻的情况下生成因果输出。非因果架构以非因果方式处理其输入,相对于输入帧具有预定大小的前瞻,以输出非因果输出。在一些实现方式中,这些前瞻帧是从因果架构复制的因果帧。

[0016] 因此,具有DCN架构的双自注意力模块可输出因果输出和非因果输出。因果输出和非因果输出可通过包括双自注意力的DNN的层传播。以这种方式,双自注意力的非因果架构可访问通过DNN的先前层的双自注意力的因果架构生成的因果帧和通过DNN的先前层的双自注意力的非因果架构生成的非因果帧二者。值得注意的是,DNN的总体输出可包括因果输出和非因果输出之一或组合。然而,由于非因果输出通常比因果输出更准确,所以在一个实施方式中,输出序列仅包括非因果输出。

[0017] 一些实施方式基于这样的认识:在内部,双自注意力将输入帧序列变换为关键字、值和查询帧的序列。查询与关键字以及关键字与值的关系是可微的。以这种方式,随着DNN学习,注意力机制可学习重塑搜索输入与提供上下文的输入之间的关系。

[0018] 为此,查询序列的各个查询帧在关键字序列上搜索,以计算各个关键字帧相对于查询帧的关系。各个关键字帧与对关于各个输入帧的特征进行编码的值帧有关。所计算的各个关键字帧相对于查询帧的关系用于向各个值帧指派加权因子,以用于计算值帧序列和用于查询搜索的输出的加权平均。例如,如果输入帧序列的各个输入帧对应于词语序列(即,句子)中的词语,则所估计的各个关键字帧相对于查询帧的关系将表示与查询帧关联的词语与句子中的所有其它词语的关系。

[0019] 因此,为了计算非因果输出,双自注意力的非因果架构需要复制因果前瞻关键字和值帧,从而允许非因果查询帧关注非因果过去和当前关键字帧以及因果前瞻关键字帧,以为过去和当前非因果值帧和因果前瞻值帧生成权重,以生成非因果输出。

[0020] 一些实施方式基于另一认识:通过只要可能就使因果帧的使用最小化,双自注意力模块的质量可进一步改进。为此,在一些实施方式中,因果架构使用从不需要超出当前查询帧位置的任何信息的非因果架构复制的过去非因果帧和当前非因果帧,而非因果架构使用从因果架构复制的因果前瞻帧。

[0021] 一些实施方式基于这样的理解:具有双自注意力模块的层的DNN可用作序列到序列问题的主要解释器,和/或可形成更复杂神经网络架构的一部分。例如,在一个实施方式中,DNN以联结主义时序分类(CTC)目标训练以输出音频输入的解释。例如,对于自动语音识别(ASR)应用,DNN被训练输出语音话语的转录。另外地或另选地,DNN可作为编码器训练以输出编码器状态,这些编码器状态可稍后变换为期望形式的输出。在一个示例中,这种后期变换可借助于与DNN一起训练的额外投影层来完成。在另一示例中,DNN形成编码器/解码器架构中的双自注意力编码器,其中,解码器也是深度神经网络。

[0022] 例如,在一个实施方式中,编码器神经网络和额外投影层以CTC目标函数来训练。此实施方式是有利的,因为基于CTC的神经网络也可应用于流应用。在另一示例中,编码器之后是基于注意力的解码器神经网络。此实施方式对于流应用不太容易使用,但是可提供比基于CTC的神经网络更准确的解释结果。

[0023] 在另一实施方式中,双自注意力编码器的输出由多个不同的解码器处理,这些解码器一起工作以实现更好的结果。例如,双自注意力编码器的输出由基于CTC和注意力的解

码器处理,以改进总体解码。另外地或另选地,在一个实施方式中,基于CTC的输出触发基于注意力的解码器的执行,以使基于注意力的解码器适应流应用。为此,一些实施方式提供一种语音识别系统,其包括与基于注意力的解码器组合的双自注意力编码器。

[0024] 因此,一个实施方式公开了一种适用于数据流应用的人工智能(AI)系统。该AI系统包括至少一个处理器;以及存储有指令的存储器,所述指令在由至少一个处理器执行时使得AI系统通过深度神经网络(DNN)的层来处理输入帧序列中的各个输入帧以生成输出序列,其中,至少一些层包括具有关注非因果帧和因果帧的双因果和非因果架构的双自注意力模块;并且渲染输出序列。

[0025] 因此,另一实施方式公开了一种音频处理方法。该音频处理方法包括以下步骤:通过深度神经网络(DNN)的层来处理输入帧序列中的各个输入帧以生成输出序列,其中,至少一些层包括具有关注非因果帧和因果帧的双因果和非因果架构的双自注意力模块;以及渲染输出序列。

[0026] 因此,另一实施方式公开了一种非暂时性计算机可读存储介质,其上具体实现有程序,该程序可由处理器执行以用于执行一种方法,该方法包括以下步骤:通过深度神经网络(DNN)的层来处理输入帧序列中的各个输入帧以生成输出序列,其中,至少一些层包括具有关注非因果帧和因果帧的双因果和非因果架构的双自注意力模块;以及渲染输出序列。

[0027] 当前公开的实施方式将参照附图进一步说明。所示的附图未必按比例,而是重点通常放在示出当前公开的实施方式的原理。

附图说明

[0028] [图1]图1是示出用于实现根据本公开的一些实施方式的人工智能(AI)系统的网络环境的框图。

[0029] [图2A]图2A是根据本公开的一些实施方式的图1中示例性示出的AI系统的框图。

[0030] [图2B]图2B示出根据本公开的一些实施方式的深度神经网络的图示表示。

[0031] [图3]图3示出根据本公开的一些实施方式的双因果和非因果架构的示意图。

[0032] [图4]图4示出根据本公开的一些实施方式的对值帧序列的各个值帧进行加权的示意图。

[0033] [图5]图5示出根据一些实施方式的配置用于端到端语音识别的语音识别系统(ASR)的示意图。

[0034] [图6A]图6A示出根据一些实施方式的对齐解码器的示意图。

[0035] [图6B]图6B示出根据一些实施方式的对编码器状态序列进行分区的示例。

[0036] [图6C]图6C示出根据一些实施方式的对编码器状态序列进行分区的示例。

[0037] [图7]图7示出根据一些实施方式的基于注意力的解码器的示例。

[0038] [图8]图8示出基于根据本公开的一些实施方式的AI系统的流系统的示意图。

[0039] [图9]图9示出根据本公开的一些示例实施方式的AI系统的总体框图。

具体实施方式

[0040] 在以下描述中,为了说明,阐述了众多具体细节以便提供本公开的彻底理解。然而,对于本领域技术人员而言将显而易见的是,本公开可在没有这些具体细节的情况下实

践。在其它情况下,设备和方法仅以框图形式示出,以避免使本公开模糊。

[0041] 如本说明书和权利要求中使用的,术语“例如”和“诸如”以及动词“包括”、“具有”、“包含”及其其它动词形式在结合一个或更多个组件或其它项目的列表使用时各自应被解释为开放式,意味着列表不应被视为排除其它附加组件或项目。术语“基于”意指至少部分地基于。此外,将理解,本文所采用的措辞和术语是为了描述目的,不应被视为限制。此描述内利用的任何标题仅是为了方便,不具有法律或限制作用。

[0042] 图1是示出用于实现根据本公开的一些实施方式的人工智能(AI)系统102的网络环境100的框图。网络环境100被描绘为包括与用户装置108关联的用户106。在例示性示例场景中,用户106向用户装置108提供输入,例如音频输入110。用户装置108可接收音频输入110作为声学信号或作为语音话语。用户装置108可包括由服务器104托管的应用,例如自动语音识别(ASR)。音频输入110可经由网络112提供给服务器104。服务器104可被配置为处理音频输入110以执行与ASR有关的不同以用于数据流应用。

[0043] 此外,网络112可包括合适的逻辑、电路和接口,其可被配置为提供多个网络端口和多个通信信道以用于数据的发送和接收。各个网络端口可对应于虚拟地址(或物理机器地址)以用于通信数据的发送和接收。例如,虚拟地址可以是互联网协议版本4(IPv4)(或IPv6地址),物理地址可以是媒体访问控制(MAC)地址。网络112可与应用层关联以用于基于来自一个或更多个通信装置中的至少一个的一个或更多个通信请求来实现通信协议。可经由通信协议发送或接收通信数据。这些有线和无线通信协议的示例可包括(但不限于)传输控制协议和互联网协议(TCP/IP)、用户数据报协议(UDP)、超文本传输协议(HTTP)、文件传输协议(FTP)、ZigBee、EDGE、红外(IR)、IEEE 802.11、802.16、蜂窝通信协议和/或蓝牙(BT)通信协议。

[0044] 网络112的示例可包括(但不限于)无线信道、有线信道、其无线和有线信道的组合。无线或有线信道可与网络标准关联,网络标准可由局域网(LAN)、个域网(PAN)、无线局域网(WLAN)、无线区域网(WAN)、无线广域网(WWAN)、长期演进(LTE)网络、普通老式电话服务(POTS)和城域网(MAN)之一定义。另外,有线信道可基于带宽标准来选择。例如,光纤信道可用于高带宽通信。此外,基于同轴线缆或基于以太网的通信信道可用于中等带宽通信。

[0045] 在一些实施方式中,音频输入110可能很长。在这样的情况下,服务器104的计算复杂度可能很高。因此,服务器104可能无法及时处理音频输入110,这可能导致生成延迟的输出。另外,由于处理很长的音频输入110可能花费时间,所以这导致对用户输入的延迟响应。为此,一些实施方式公开了AI系统102,其可以较少的延迟时间生成准确输出。

[0046] 图2A是根据本公开的一些实施方式的图1中示例性示出的AI系统102的框图。AI系统102适用于数据流应用。AI系统102包括处理器202、存储器204和输入/输出(I/O)接口210。在示例实施方式中,I/O接口210被配置为接收输入序列,其中输入序列可对应于具有时间维度的音频输入(例如音频输入110)。此外,处理器202被配置为执行存储在存储器204中的指令。执行存储的指令使得AI系统102接受输入帧序列。输入帧序列表示描述输入信号(例如音频输入110)的信息的有序特征序列。

[0047] 一些实施方式基于这样的理解:自注意力以及受限自注意力具有非因果架构。词语因果和非因果是信号处理领域的术语,在本文中以其平常含义使用。例如,在信号处理中,因果滤波器是线性时不变因果系统,其中词语“因果”指示滤波器输出仅取决于过去和

现在的输入。输出还取决于未来输入的滤波器是非因果的。因此,如本文所使用的,自注意力模块的非因果架构使用过去、当前以及在当前输入帧之后到来的未来输入来解释当前输入帧。

[0048] 类似地,受限自注意力模块可使用因果架构来实现。这种架构不使用未来信息,因此不引入任何延迟。结果,因果自注意力模块的任何组合不会导致任何附加延迟。然而,使用因果自注意力的序列到序列处理的性能比非因果自注意力的性能差。为此,一些实施方式的目的是将因果架构与非因果架构组合以用于数据流应用,以利用二者的优点。

[0049] 一些实施方式基于这样的认识:非因果架构使用两个主要原理。第一个原理是关注输入帧的方式,第二个原理是输入帧的本质。在非因果架构中,两个原理均是非因果的。具体地,非因果自注意力关注未来前瞻帧,因此关注的方式是非因果的。另外,第二个原理规定前瞻帧应该考虑未来数据根据非因果原理来生成。因此,生成前瞻帧的方式也是非因果的。结果,前瞻大小为两个帧的非因果架构需要关注两个未来前瞻帧,这两个未来前瞻帧继而可利用另两个未来前瞻帧生成,这另两个未来前瞻帧继而可利用另两个前瞻帧生成,依此类推,遵循使用自注意力模块的神经网络的深度。

[0050] 一些实施方式基于这样的认识:可通过提供双因果和非因果(DCN)架构来修改非因果架构的原理。在DCN架构中,关注输入帧的方式仍是非因果的,然而,输入帧本身可以是因果或非因果的。另外地或另选地,在DCN架构中,关注输入帧的方式可以是因果或非因果的,并且输入帧可以是因果或非因果的。如本文所述,因果输入或因果帧是在不考虑一个或多个未来输入的情况下生成的输入或帧,而非因果输入或非因果帧是考虑一个或多个未来输入生成的输入或帧。结果,深度神经网络的非因果架构内的因果输入可用于降低总处理延迟,而准确性仅部分降低。

[0051] 为此,在实施方式中,存储器204包括深度神经网络(DNN)206。DNN 206可包括层,层中的至少一些包括具有DCN架构的双自注意力模块208。

[0052] 图2B示出根据本公开的一些实施方式的DNN 206的图示表示。DNN 206包括层206a-206e。层206a-206e中的一些层包括双自注意力模块208。例如,DNN 206包括输入层206a、输出层206e以及双自注意力层206b、206c和206d。自注意力层206b、206c和206d包括DCN架构。处理器202被配置为通过DNN 206的层来处理输入帧序列中的各个输入帧,以生成输出序列。

[0053] 图3示出根据本公开的一些实施方式的生成输出序列的DCN架构300的示意图。处理器202复制输入特征序列以推导因果输入帧序列302和非因果输入帧序列304。此外,DCN架构300将因果输入帧和非因果输入帧变换为因果和非因果关键字、值和查询帧。DCN架构300包括因果架构和非因果架构。DCN架构300被配置为执行双自注意力,维持因果架构和非因果架构二者彼此并行操作。根据实施方式,双自注意力包括因果自注意力处理306和非因果自注意力处理308。因果自注意力处理306和非因果自注意力处理308并行执行。

[0054] 因果输入帧序列302作为输入被馈送到因果自注意力处理306。因果自注意力处理306包括通过因果架构以因果方式(即,无需前瞻帧)处理因果输入帧序列302的各个因果输入帧,以生成因果输出帧序列310。

[0055] 此外,非因果输入帧序列304作为输入被馈送到非因果自注意力处理308。非因果自注意力处理308包括通过非因果架构以非因果方式相对于表示当前时间的输入帧以预定

大小的前瞻处理非因果输入帧序列304的各个非因果输入帧,以生成非因果输出帧序列312。在非因果自注意力处理308中,前瞻帧是因果帧,而所关注的其它帧对应于非因果帧。例如,为了解释当前输入帧314,关注作为因果帧的前瞻帧318和作为非因果帧的其它帧316。当前瞻帧为因果的时,在前双自注意力模块不需要附加前瞻帧来生成它们。值得注意的是,具有非因果过去帧不引入任何附加延迟。因此,实施方式可使用尽可能多的非因果帧,从而限制性能的下降,同时仅使用因果前瞻帧,从而限制处理延迟。

[0056] 因此,双自注意力模块206的DCN架构300输出因果输出帧序列310和非因果输出帧序列312。此外,因果输出帧序列310和非因果输出帧序列312可通过包括双自注意力的DNN 206的层传播。以这种方式,双自注意力的非因果架构可以访问通过DNN 206的先前层的双自注意力的因果架构生成的因果帧和通过DNN 206的先前层的双自注意力的非因果架构生成的非因果帧二者。

[0057] 根据实施方式,因果输出帧序列310和非因果输出帧序列312对应于输出序列。然而,由于非因果输出帧通常比因果输出帧更准确,所以在一个实施方式中,输出序列仅包括非因果输出帧序列312。

[0058] 一些实施方式基于这样的认识:在内部,双自注意力将输入帧序列变换为关键字、值和查询帧的序列。查询与关键字以及关键字与值的关系是可微的。以这种方式,随着DNN 206学习,注意力机制可学习重塑搜索输入与提供上下文的输入之间的关系。

[0059] 为此,查询序列的各个查询帧在关键字序列上搜索,以计算各个关键字帧相对于查询帧的关系。各个关键字帧与对关于各个输入帧的特征进行编码的值帧有关。所计算的各个关键字帧相对于查询帧的关系用于向各个值帧指派加权因子,以用于计算值帧序列和用于查询搜索的输出的加权平均。例如,如果输入帧序列的各个输入帧对应于词语序列(即,句子)中的词语,则所估计的各个关键字帧相对于查询帧的关系可表示与查询帧关联的词语与句子中的所有其它词语的关系。

[0060] 图4示出根据本公开的一些实施方式的对值帧序列中的各个值帧进行加权的示意图。输入因果帧400被馈送到DNN 206的当前层的双自注意力的因果架构。根据实施方式,输入因果帧400可对应于通过DNN 206的先前层的双自注意力的因果架构确定的因果输出。DNN 206的当前层的双自注意力的因果架构将先前因果输出变换为因果关键字帧、因果值帧和因果查询帧的序列,以通过使因果查询帧关注因果关键字帧以对因果值帧进行加权来生成当前因果输出414。

[0061] 输入非因果帧416被馈送到DNN 206的当前层的双自注意力的非因果架构。根据实施方式,输入非因果帧416可对应于通过DNN 206的先前层的双自注意力的非因果架构确定的非因果输出。DNN 206的当前层的双自注意力的非因果架构将先前非因果输出变换为非因果关键字帧、非因果值帧和非因果查询帧的序列,以生成当前非因果输出430。具体地,在实施方式中,当前非因果输出430通过使非因果查询帧422关注与从因果架构复制的因果前瞻关键字帧420组合的非因果关键字帧418,以对与从因果架构复制的因果前瞻值帧426组合的非因果值帧424进行加权而生成。为与因果前瞻值帧426组合的非因果值帧424估计的权重由428表示。

[0062] 一些实施方式基于另一认识:通过只要可能就使因果帧的使用最小化,双自注意力模块208的质量可进一步改进。为此,在一些实施方式中,因果架构使用从不需要超出当

前查询帧位置的任何信息的非因果架构复制的过去非因果帧,而非因果架构使用从因果架构复制的因果前瞻帧。

[0063] 例如,因果架构通过使因果查询帧406关注与从非因果架构复制的非因果关键字帧402组合的因果关键字帧404,以对与从非因果架构复制的非因果值帧408组合的因果值帧410进行加权来生成当前因果输出414。为与非因果值帧408组合的因果值帧410估计的权重由412表示。另外地或另选地,非因果架构通过使非因果查询帧422关注与从因果架构复制的因果前瞻关键字帧420组合的过去和当前非因果关键字帧418,以对与从因果架构复制的因果前瞻值帧426组合的非因果过去和当前值帧424进行加权来生成当前非因果输出430。

[0064] 一些实施方式基于这样的理解:具有双自注意力模块层的DNN 206可用作序列到序列问题的主要解释器,和/或可形成更复杂神经网络架构的一部分。

[0065] 例如,在一个实施方式中,DNN 206以联结主义时序分类(CTC)目标训练以输出音频输入的解释。例如,对于ASR应用,DNN 206被训练以输出语音话语的转录。另外地或另选地,DNN 206可作为编码器训练以输出编码器状态,这些编码器状态可稍后变换为期望形式的输出。在一个示例中,这种后期变换可借助于与DNN 206一起训练的额外投影层来完成。在另一示例中,DNN 206形成编码器/解码器架构中的双自注意力编码器,其中,解码器也是深度神经网络。

[0066] 例如,在一个实施方式中,编码器神经网络和额外投影层以CTC目标函数来训练。此实施方式是有利的,因为基于CTC的神经网络也可应用于流应用。在另一示例中,编码器之后是基于注意力的解码器神经网络。此实施方式不太容易用于流应用,但是可提供比基于CTC的神经网络更准确的解释结果。

[0067] 在另一实施方式中,双自注意力编码器的输出由多个不同的解码器处理,这些解码器一起工作以实现更好的结果。例如,双自注意力编码器的输出由基于CTC和注意力的解码器处理,以改进总体解码。另外地或另选地,在一个实施方式中,基于CTC的输出触发基于注意力的解码器的执行,以使基于注意力的解码器适用于流应用。为此,一些实施方式提供一种语音识别系统,其包括与基于注意力的解码器组合的双自注意力编码器。

[0068] 图5示出根据一些实施方式的配置用于端到端语音识别的语音识别系统(ASR)500的示意图。语音识别系统500获得输入声学序列并处理输入声学序列以生成转录输出序列。各个转录输出序列是由对应输入声学信号表示的话语或话语的一部分的转录。例如,语音识别系统500可获得输入声学信号502并且生成作为由输入声学信号502表示的话语的转录的对应转录输出510。

[0069] 输入声学信号502可包括作为话语的数字表示的多个音频数据帧的序列,例如连续数据流。多个音频数据帧的序列可对应于时间步序列,例如其中各个音频数据帧与时间上从先前音频数据帧进一步移位10毫秒的25毫秒音频流数据关联。多个音频数据帧的序列中的各个音频数据帧可包括表征对应时间步的话语部分的帧的特征值。例如,多个音频数据帧的序列可包括滤波器组频谱特征向量。

[0070] 转录输出510可包括由输入声学信号502表示的话语的转录片段序列。转录输出可包括一个或更多个字符。例如,转录输出可以是来自Unicode字符集的字符或字符序列。例如,字符集可包括英语、亚洲语、西里尔语以及阿拉伯语的字母表。字符集还可包括阿拉伯

数字、空格字符和标点符号。另外地或另选地,转录输出可包括比特、词语以及其它语言结构。

[0071] 语音识别系统500包括编码器504和基于注意力的解码器508。编码器504可以是双自注意力编码器。编码器504处理输入声学信号502并且生成提供输入声学信号502的替代(例如,更高)表示的编码器状态序列506。编码器状态序列506可包括对应于第二时间步集合的多个音频数据帧的替代序列。基于注意力的解码器508被训练以处理表示输入声学信号502的替代表示的编码器状态序列506,并且从提供给基于注意力的解码器508的编码器状态序列生成转录输出510。

[0072] 一些实施方式基于这样的认识:基于注意力的ASR系统可能需要观察通过语音停顿分割的整个语音话语,以向各个输入帧指派权重,以便识别各个转录输出510。由于缺少关于输入声学信号的哪一部分与识别下一转录输出相关的先验知识并且需要向各个编码器状态指派权重,基于注意力的解码器通常需要处理大的输入序列。这种处理允许利用对话语的不同部分投入注意力,但也增加了输出延迟,因此对于流/在线方式的语音识别不切实际。

[0073] 如本文所使用的,ASR的输出延迟是接收语音话语的声学帧的时间与识别所接收的声学帧的时间之间的差。例如,当基于注意力的ASR系统在整个语音话语上操作时,话语中的词语的识别被延迟直至接收到话语的最后词语。这种识别延迟导致输出延迟增加。

[0074] 一些实施方式基于这样的认识:关于输入序列的不同部分与下一转录输出的识别的相关性的先验知识的示例是与输入序列中要识别的转录输出对应的帧位置的指示。实际上,如果转录输出位置已知,则通过限制输入序列,基于注意力的解码器可被迫对这些位置投入更大的注意力,对其它位置投入较少或不投入注意力。以这种方式,对于各个转录输出,基于注意力的网络可将其注意力集中在其在输入序列中的位置周围。这种引导的注意力减少了处理大的输入序列的需求,这继而减小了输出延迟,使得基于注意力的解码器对于流/在线方式的识别是实用的。

[0075] 为此,ASR系统500包括对齐解码器520,其被训练以确定对诸如字符、比特、词语等的转录输出进行编码的序列506中的编码器状态的位置525。CTC是一种目标函数和关联的神经网络输出,用于训练诸如长短期记忆(LSTM)网络的递归神经网络(RNN),以解决定时可变的序列问题。基于CTC的ASR系统是基于注意力的ASR系统的替代。基于CTC的神经网络为输入序列的各个帧生成输出(即,输入和输出同步),并且波束搜索算法用于在将神经网络输出折叠到输出转录之前寻找最佳输出序列。基于注意力的ASR系统的性能可能优于基于CTC的ASR系统。然而,一些实施方式基于这样的认识:基于注意力的ASR系统可使用基于CTC的ASR系统的中间操作所使用的输入和输出帧对齐来解决其上述输出延迟缺点。

[0076] 为了利用对齐解码器520所提供的对齐信息525,ASR系统500包括分区模块530,其被配置为将编码器状态序列506分区为分区集合535。例如,分区模块530可针对所识别的编码器状态的各个位置525对编码器状态序列进行分割,使得分区535的数量由所识别的对转录输出进行编码的编码器状态525的数量定义(例如,相等)。以这种方式,基于注意力的解码器508不是接受整个序列506,而且该序列的部分535作为输入,而各个部分很可能包括新的转录输出以形成转录输出序列510。在一些实现方式中,对齐解码器、基于注意力的解码器和分区模块的组合被称为触发注意力解码器。实际上,触发注意力解码器可在接收到话

语部分时对其进行处理,使得ASR系统500对于流/在线方式的识别是实用的。

[0077] 图6A示出根据一些实施方式的对齐解码器520的示意图。对齐解码器520的目标之一是对编码器504所生成的编码器状态序列506进行解码。为此,对齐解码器520被训练以对编码器状态序列506进行解码以生成转录输出序列600。然而,至少一些实施方式不使用对齐解码器520的解码的转录输出600。相反,一些实施方式使用由对齐解码器520生成的中间对齐信息来对编码器状态序列506进行解码。换言之,一些实施方式忽略由对齐解码器520解码的转录输出600,而是使用序列506中的编码器状态的位置525来改进基于注意力的解码器508的性能。这种方法背后的理论基础是基于注意力的解码器508的性能可能好于对齐解码器520的性能。为此,由对齐解码器520生成的中间对齐信息用于进一步改进基于注意力的解码器508的性能。

[0078] 然而,在一些实施方式中,由对齐解码器520解码的转录输出600进一步与由基于注意力的解码器508解码的转录输出组合,以进一步改进识别的准确性。在这些实施方式中,对齐解码器520被利用两次:第一次是辅助对编码器状态序列506进行分区以用于基于注意力的解码器508,第二次用于进一步改进由基于注意力的解码器508解码的转录输出的准确性。

[0079] 图6A示出对齐解码器520处理话语带词语“dog”的示例部分的操作的示例。围绕索引序列的元素的框标识对编码器状态序列506中的转录输出进行编码的编码器状态525的位置。例如,编码器504将诸如对数梅尔谱能量的声学特征的输入声学序列X转换为T编码器状态序列H:

[0080] $H=\mathrm{Encoder}(X)$。

[0081] 例如,在一个实现方式中,通过编码器神经网络将特征矩阵X下采样至低四倍的帧频,使得当特征矩阵X的帧频达到100Hz时,编码器输出H具有25Hz的帧频。设$Z=(z_1,\cdots,z_T)$表示从编码器状态H 506计算的长度T的CTC输出的逐帧序列,$z_t \in \mathcal{U} \cup \epsilon$,其中$\mathcal{U}$表示不同字素的集合,其例如可以是单个字符或词语片段,∈表示空白符号。设$C=(c_1,\cdots,c_L)$表示长度L的字素序列,其中$c_l \in \mathcal{U}$,使得在将重复标签折叠成单次发生并去除空白符号时,序列Z减小为C。

[0082] 在一些实施方式中,对齐解码器520对编码器状态序列进行概率解码,其中概率被推导为:

[0083]
$$p_{\mathrm{ctc}}(C|H) = \sum_Z p(C|Z,H)p(Z|H)$$
$$\approx \sum_Z p(C|Z)p(Z|H)$$
$$= \sum_Z p(Z|C)p(Z|H)\frac{p(C)}{p(Z)},$$

[0084] 其中$p(Z|C)$表示转移概率,$p(Z|H)$表示声学模型。

[0085] 在一些实施方式中,对齐解码器520将与Z中的相同字素对应的帧的各个子序列内具有最高概率的帧识别为所识别的CTC输出。例如,如果对于Z中的第1标签c_l的出现的开始和结束使用索引i_l和j_l来书写序列Z,其中$i_l \le j_l < i_{l+1},\ \forall l$,并且对于所有t,$z_t=c_l$,使得对于所有其它索引,$i_l \le t \le j_l$并且$z_t=\epsilon$。对齐解码器520执行从序列Z到包括所识别的对具有最高概率的转录输出进行编码的编码器状态525的子集的相同长度T的序列Z′=

$(\in^*, c_1, \in^*, [0]c_2, \in^*, \cdots, c_L, \in^*)$的映射,其中*表示零或更多次重复,并且其中各个$c_1$在与$c_1$对应的那些帧内具有最大概率的帧处恰好出现一次,如下:

[0086] $$z'_t = \begin{cases} c_l & \text{如果} t = argmax_{i_l \le t' \le j_l} p(z_{t'}|\mathrm{H}),\ \text{对于一些} l \in \{1, \ldots, L\}, \\ \langle b \rangle & \text{否则。} \end{cases}$$

[0087] 另选地,对齐解码器520可将与Z中的相同字素对应的帧的各个子序列内的第一或最后帧识别为所识别的编码器状态。

[0088] 图6B和图6C示出根据一些实施方式的对编码器状态序列506进行分区的示例。在各种实施方式中,由操作上连接到对齐解码器520、基于注意力的解码器508和编码器504的分区模块530执行分区。分区模块530被配置为访问对齐解码器520的对齐信息525,对编码器504所生成的编码器状态序列进行分区,并且将编码器状态序列的部分535依次提交给基于注意力的解码器508。

[0089] 例如,在图6B的一个实施方式中,各个分区602a包括从编码器状态序列的开始直至通过使所识别的编码器状态的位置以固定移位向前移位而确定的前瞻编码器状态的编码器状态。例如,如果固定移位的值为5,并且所识别的状态编码器的位置是编码器状态序列中的第8个,则分区602a包括前13个编码器状态。如果随后识别的编码器状态的位置是11,则分区602a包括前16个编码器状态。实际上,各个分区包括用于新转录输出的编码器状态,同时增加了分区的长度,从而允许基于注意力的解码器508利用其长度。

[0090] 在图6C的替代实施方式中,与所识别的编码器状态的位置对应的分区602b包括以所识别的编码器状态的位置为中心的预定数量的编码器状态。例如,如果编码器状态的预定数量为7并且所识别的编码器状态的位置是编码器状态序列中的第15个,则分区602b包括编码器状态序列506中的第12个和第18个之间的编码器状态。实际上,在具有固定长度的分区的同时,各个分区包括用于新转录输出的编码器状态,以减小基于注意力的解码器508上的计算负担。另外地或另选地,与所识别的编码器状态的位置对应的分区602b包括所识别的编码器状态的位置周围(例如,从中心偏移以提供偏心覆盖)的预定数量的编码器状态。

[0091] 图7示出根据一些实施方式的示例基于注意力的解码器508。基于注意力的解码器508包括上下文向量生成器704和解码器神经网络706。上下文向量生成器704接收来自在前时间步的解码器神经网络706的隐藏解码器状态712、来自在前时间步的上下文向量生成器的注意力权重分布710以及替代表示506(即,上面参照图5描述的声学信号502的替代表示)。上下文向量生成器704处理解码器神经网络706的在前隐藏解码器状态、在前注意力权重分布710和替代表示506以计算替代表示506的时间帧上的注意力权重分布并且生成时间步的上下文向量714作为输出。上下文向量生成器704将时间步的上下文向量714提供给解码器神经网络706。

[0092] 对于不同的迭代,基于注意力的解码器508接收不同的分区731、733和735。例如,分区集合包括第一分区731以及后续分区733和735。基于注意力的解码器508处理第一分区731以生成第一转录输出。在基于注意力的神经网络完成处理将基于注意力的网络置于其内部状态的第一分区之后,基于注意力的解码器508以基于注意力的网络处理后续分区而不重置基于注意力的网络的内部状态,以逐个地为后续分区生成转录输出。

[0093] 实际上,基于注意力的解码器508处理不同的分区而不重置基于注意力的网络的内部状态,以利用先前解码的信息。在确定语音话语结束时,基于注意力的解码器508被配置为重置其内部状态。

[0094] 解码器神经网络706接收该时间步的上下文向量714以及在前时间步的转录输出708和隐藏解码器状态712作为输入。解码器神经网络706在处理该时间步的上下文向量714和来自在前时间步的转录输出708之前利用在前隐藏解码器状态712将其内部隐藏状态初始化,以生成该时间步的转录输出得分集合716作为输出。在一些实现方式中,解码器神经网络706是具有softmax输出层的递归神经网络(RNN)。各个转录输出得分对应于来自转录输出集合的各个转录输出。例如,如上面参照图5描述的,转录输出集合可以是来自用于书写一种或更多种自然语言的Unicode字符集的字符或字符序列,例如英语、亚洲语、西里尔语和阿拉伯语的字母表。转录输出集还可包括阿拉伯数字、空格字符和标点符号。给定转录输出的得分表示对应转录输出是作为话语的转录的输出序列中该时间步的当前转录片段的可能性。

[0095] 语音识别系统处理各个时间步的转录输出得分716,以确定表示话语的转录的转录输出序列。例如,对于各个时间步,语音识别系统可从转录输出得分集合选择具有最高得分的转录输出,以确定转录输出序列。

[0096] 图8示出基于根据本公开的一些实施方式的AI系统102的流系统800的示意图。AI系统102捕获来自现场事件中说话的用户802的语音话语数据(或音频数据)。另外,AI系统102还可捕获说话的用户802的视频数据。语音话语数据可包括语音话语。语音话语包括多个音频数据帧的序列。AI系统102通过DNN 206的层实时处理多个音频数据帧的序列中的各个音频数据帧,以生成用于流和其它现场应用的输出序列804。输出序列804可对应于语音话语的转录。

[0097] 例如,AI系统102可通信上联接到用户装置806。AI系统102可将视频数据和语音话语数据连同语音话语的转录一起发送,以用于事件的现场流传输。换言之,在现场事件的流传输期间,AI系统102提供与现场事件中发生的用户的语音话语对应的字幕。

[0098] 在一些实现方式中,AI系统102或AI系统102的功能可被集成到多个不同的装置和计算机系统中,包括例如通用计算系统、服务器-客户端计算系统、大型机计算系统、电话计算系统、膝上型计算机、蜂窝电话、个人数字助理(PDA)、平板计算机、其它移动装置等。AI系统102也可以是可提供语音识别功能的其它装置或系统的组件,例如自动柜员机(ATM)、售货亭、家用电器(例如冰箱、烤箱等)和车辆(例如汽车、公共汽车、摩托车等),以用于不同的语音识别应用。

[0099] 在一些情况下,AI系统102也可向用户提供语音到文本文档化。例如,用户可将语音话语提供给用户装置。用户装置可与AI系统102通信以提供语音话语的转录。AI系统102可基于转录来提供文本文档。这可帮助用户(例如,用户902)经由语音输入来书写文本或维护文档。

[0100] 图9示出根据本公开的一些实施方式的AI系统900的框图。AI系统900对应于图1的AI系统102。AI系统900可具有将AI系统900与其它系统和装置连接的多个接口。例如,网络接口控制器(NIC)916适于将AI系统900通过总线914连接到网络922,网络922将AI系统900与传感器连接。通过网络922(无论以无线方式还是通过有线),AI系统900接收包括输入帧

序列的输入数据924。另外地或另选地,可经由输入接口902接收包括输入帧序列的输入数据924。

[0101] AI系统900包括被配置为执行所存储的指令的处理器904以及存储可由处理器904执行的指令的存储器912。处理器904可以是单核处理器、多核处理器、计算集群或任何数量的其它配置。存储器912可包括随机存取存储器(RAM)、只读存储器(ROM)、闪存或任何其它合适的存储器系统。处理器904通过总线914连接到一个或更多个输入装置和输出装置。此外,AI系统900包括存储装置906,其适于存储包括可由处理器904执行的指令的不同模块。存储装置906可使用硬盘驱动器、光盘驱动器、拇指驱动器、驱动器阵列或其任何组合来实现。

[0102] 存储装置906被配置为存储深度神经网络908。深度神经网络908包括多个层。在实施方式中,多个层中的一些包括具有关注非因果帧和因果帧的双因果和非因果架构的双自注意力模块910。处理器904被配置为通过深度神经网络908的层来处理输入帧序列中的各个输入帧以生成输出序列,并且渲染输出序列。

[0103] 根据实施方式,双自注意力模块910包括输出在没有前瞻的情况下推导的因果输出帧序列和相对于表示当前时间的输入帧以预定大小的前瞻推导的非因果输出帧序列的双自注意力。为此,在实施方式中,输出序列包括因果输出帧序列和非因果输出帧序列之一或组合。在替代实施方式中,输出序列仅包括非因果输出帧序列。

[0104] 另外,AI系统900包括输出接口918。输出接口918被配置为将输出序列提交给输出装置。

[0105] 以下描述仅提供示例性实施方式,并非旨在限制本公开的范围、适用性或配置。相反,示例性实施方式的以下描述将向本领域技术人员提供用于实现一个或更多个示例性实施方式的可行描述。在不脱离所附权利要求中阐述的公开的主题的精神和范围的情况下,可以想到可对元件的功能和布置进行各种改变。

[0106] 在以下描述中给出具体细节以提供实施方式的彻底理解。然而,本领域普通技术人员可理解,实施方式可在没有这些具体细节的情况下实践。例如,所公开的主题中的系统、过程和其它元件可作为组件以框图形式示出,以免在不必要的细节方面使实施方式模糊。在其它情况下,熟知过程、结构和技术可在没有不必要的细节的情况下示出,以避免使实施方式模糊。此外,各种附图中的相似标号和指代指示相似的元件。

[0107] 另外,各个实施方式可作为过程描述,其被描绘为流程图、数据流程图、结构图或框图。尽管流程图可将操作描述为顺序过程,但许多操作可并行或同时执行。另外,操作次序可重新布置。过程在其操作完成时可终止,但是可具有未讨论或附图中未包括的附加步骤。此外,并非任何具体描述的过程中的所有操作可出现在所有实施方式中。过程可对应于方法、函数、程序、子例程、子程序等。当过程对应于函数时,函数的终止可对应于函数返回到调用函数或主函数。

[0108] 此外,所公开的主题的实施方式可至少部分地手动或自动实现。可通过使用机器、硬件、软件、固件、中间件、微码、硬件描述语言或其任何组合来执行或至少辅助手动或自动实现。当以软件、固件、中间件或微码实现时,执行所需任务的程序代码或代码段可被存储在机器可读介质中。处理器可执行所需任务。

[0109] 另外,本文中概述的各种方法或处理可被编码为可在采用各种操作系统或平台中

的任一种的一个或更多个处理器上执行的软件。另外,这种软件可使用多种合适的编程语言和/或编程或脚本工具中的任一种来编写,并且也可被编译为在框架或虚拟机上执行的可执行机器语言代码或中间代码。通常,在各种实施方式中,程序模块的功能可根据需要组合或分布。

[0110] 本公开的实施方式可被具体实现为一种方法,已提供其示例。作为该方法的一部分执行的动作可按照任何合适的方式排序。因此,可构造以与所示不同的次序执行动作的实施方式,其可包括同时执行一些动作,尽管在例示性实施方式中作为顺序动作示出。

[0111] 尽管参考特定优选实施方式描述了本公开,但是将理解,在本公开的精神和范围内可进行各种其它调整和修改。因此,所附权利要求的方面涵盖落在本公开的真实精神和范围内的所有这些变化和修改。

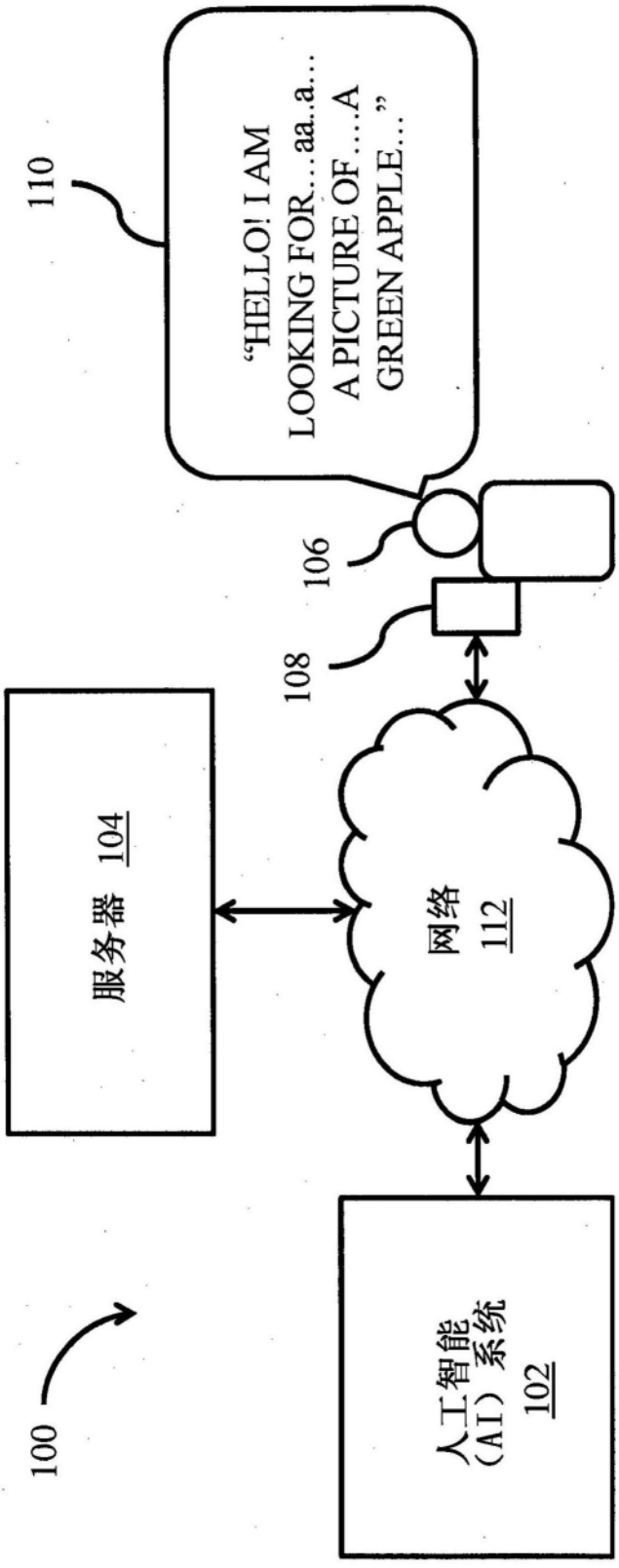

"HELLO! I AM LOOKING FOR...aa..a... A PICTURE OF....A GREEN APPLE..."
110
106
108
服务器 104
网络 112
人工智能 (AI) 系统 102
100

图1

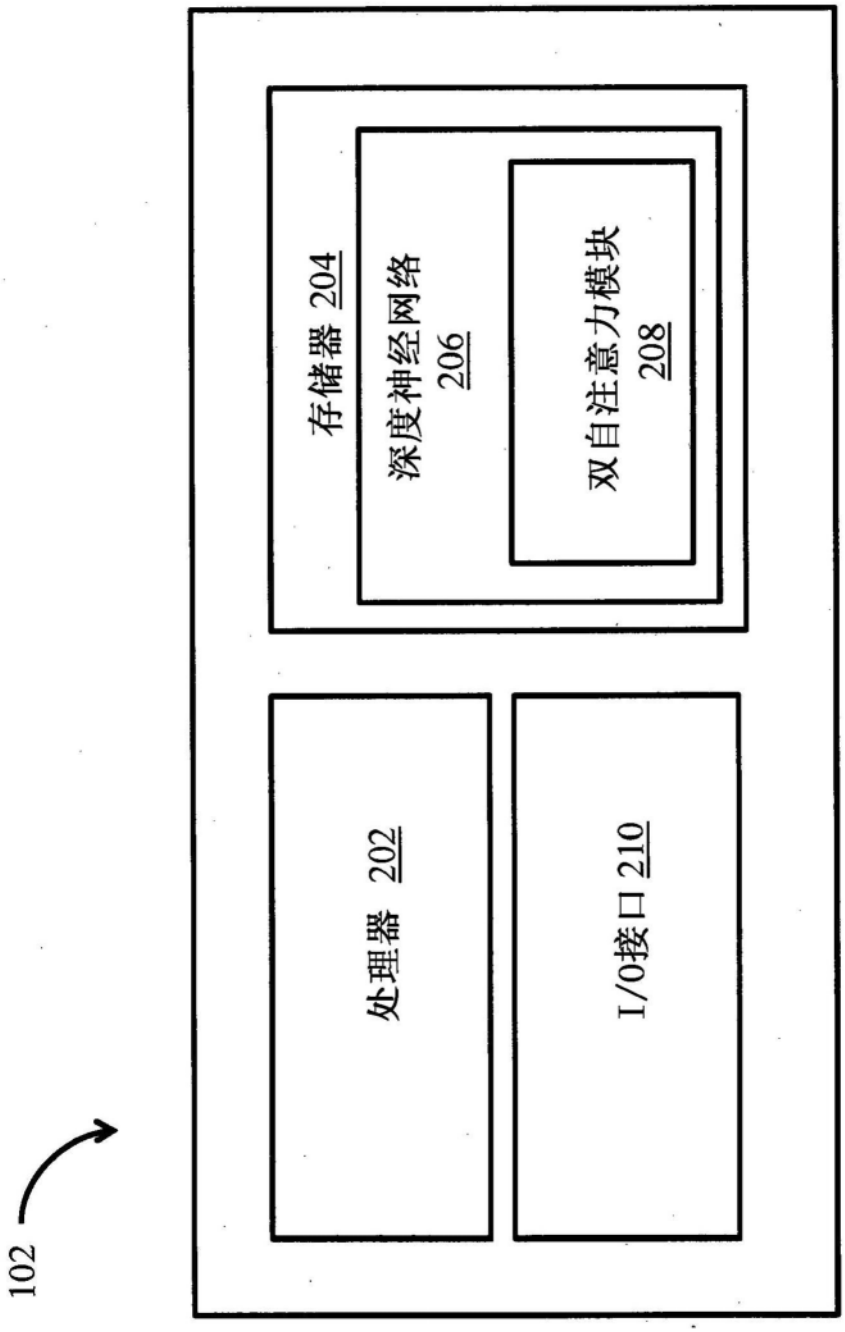


图2A

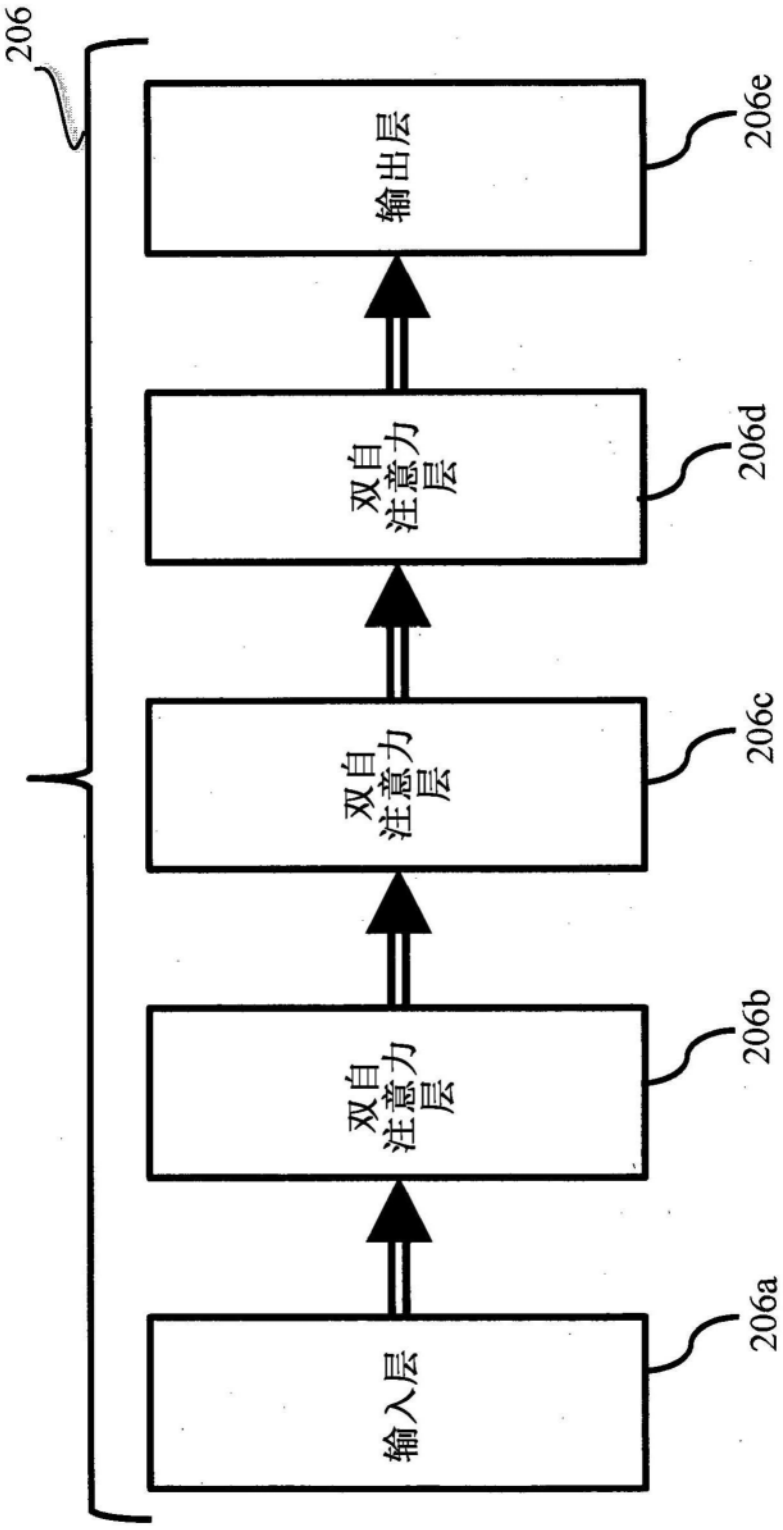


图2B

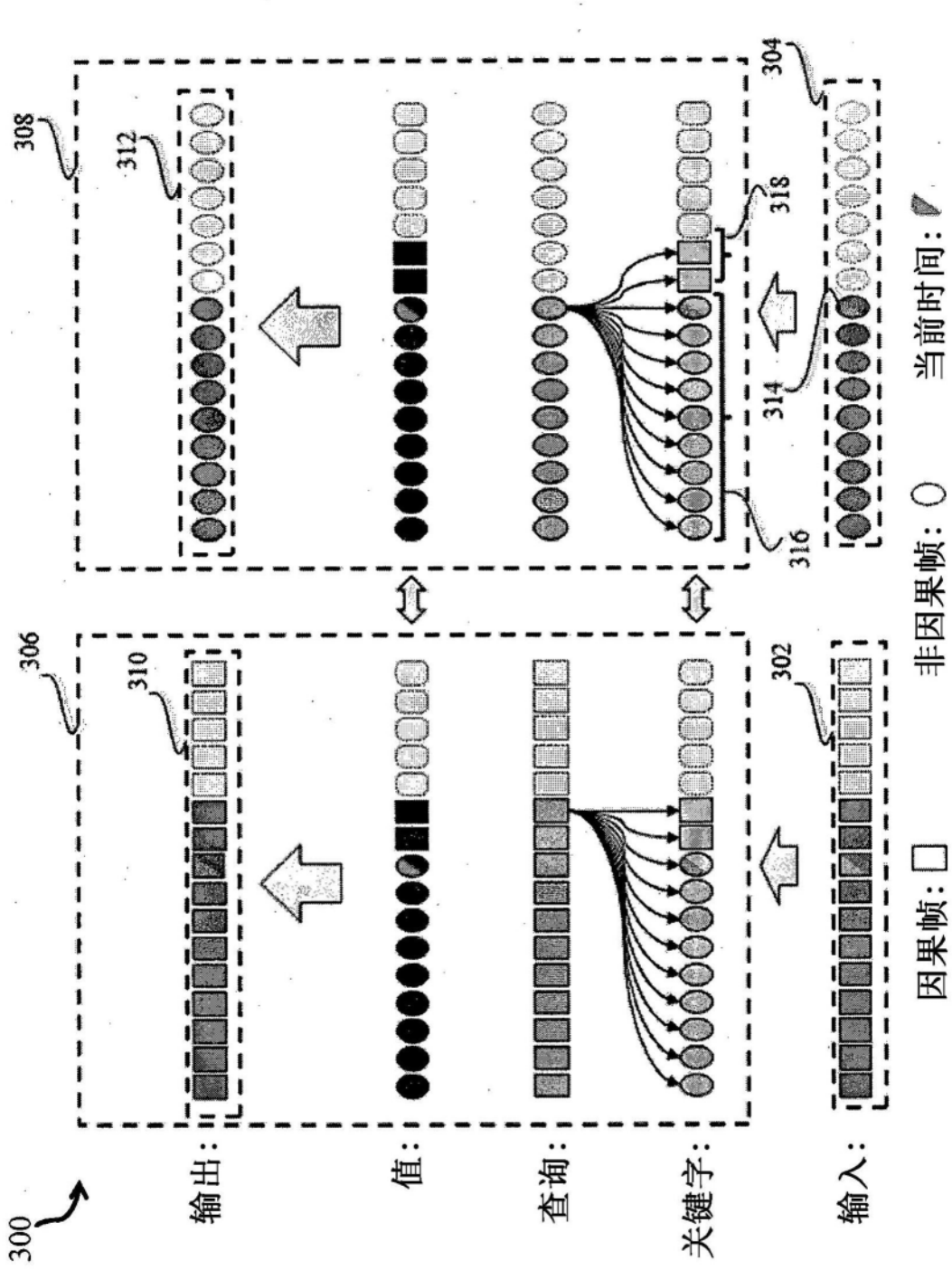

300
306
308
310
312
302
304
316
318
314
输出:
值:
查询:
关键字:
输入:
因果帧:
非因果帧:
当前时间:

图3

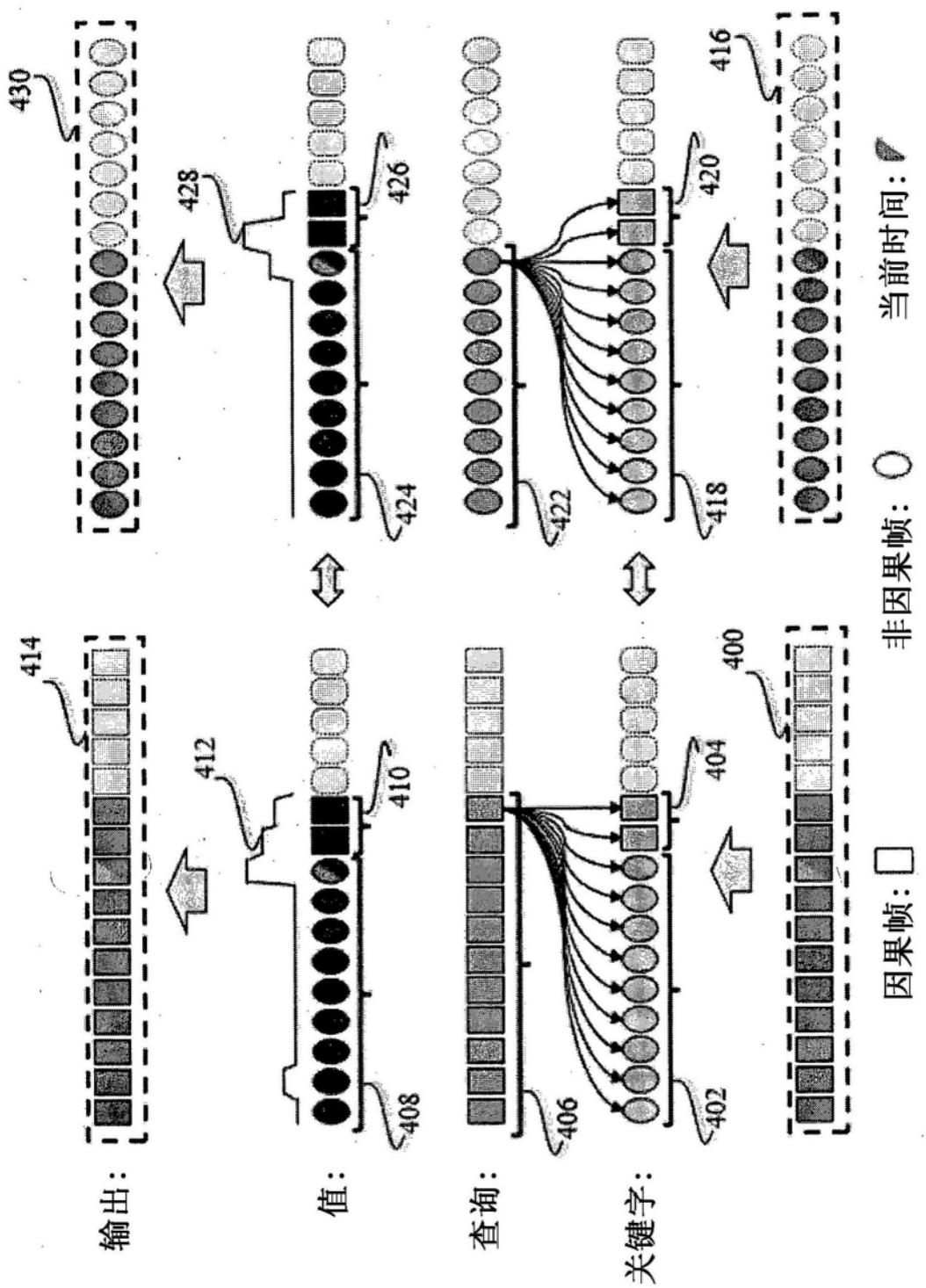

输出：
值：
查询：
关键字：
414
412
410
408
430
428
426
424
400
404
402
406
416
420
418
422
因果帧：
非因果帧：
当前时间：

图4

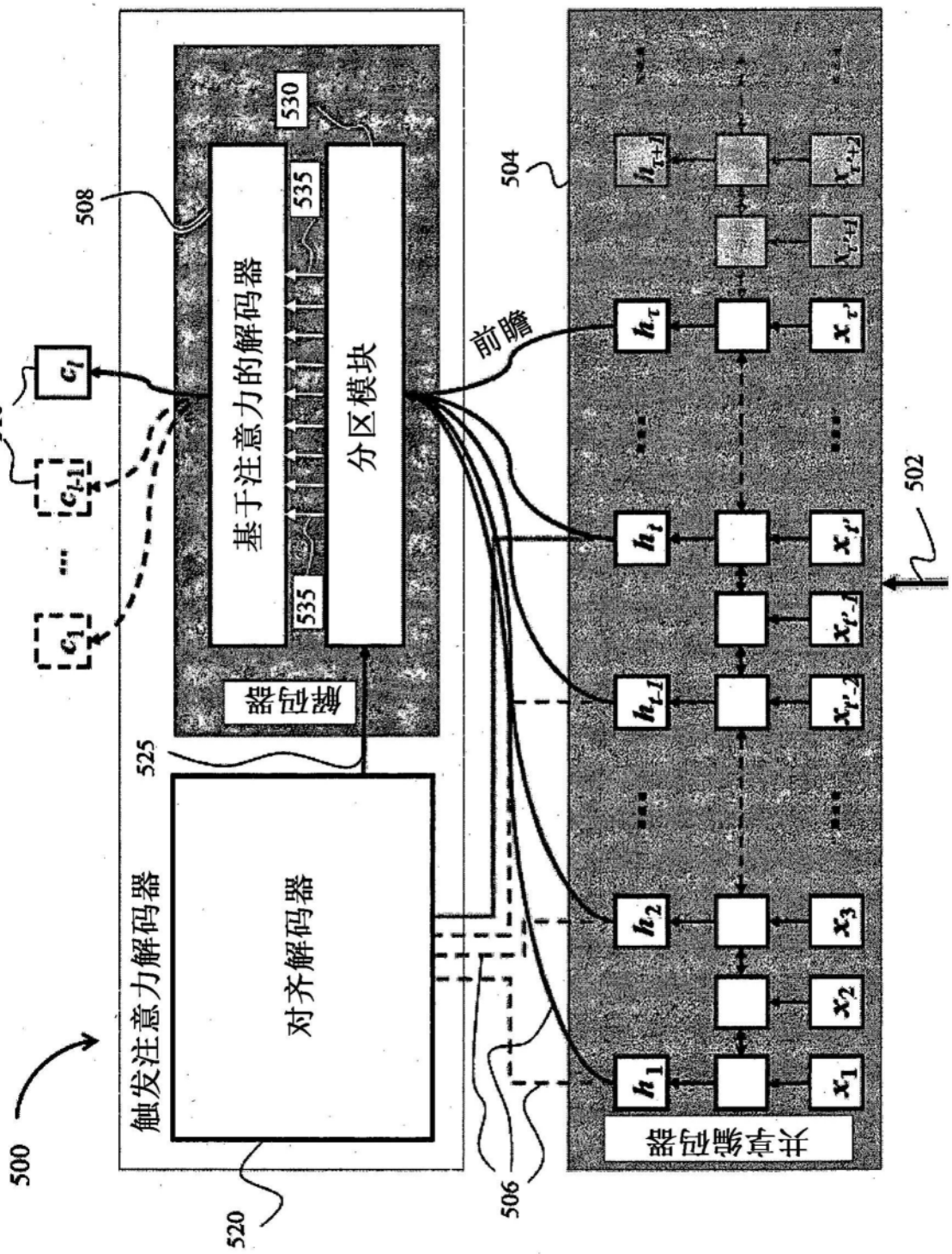

500
触发注意力解码器
对齐解码器
520
525
解码器
基于注意力的解码器
530
535
535
分区模块
508
510
c_t
c_{t-1}
c_1
前瞻
504
506
共享编码器
502
h_1
h_2
h_{t-1}
h_t
h_{t+1}
x_1
x_2
x_3
$x_{t'-2}$
$x_{t'-1}$
$x_{t'}$
$x_{t'+1}$
$x_{t'+2}$

图5

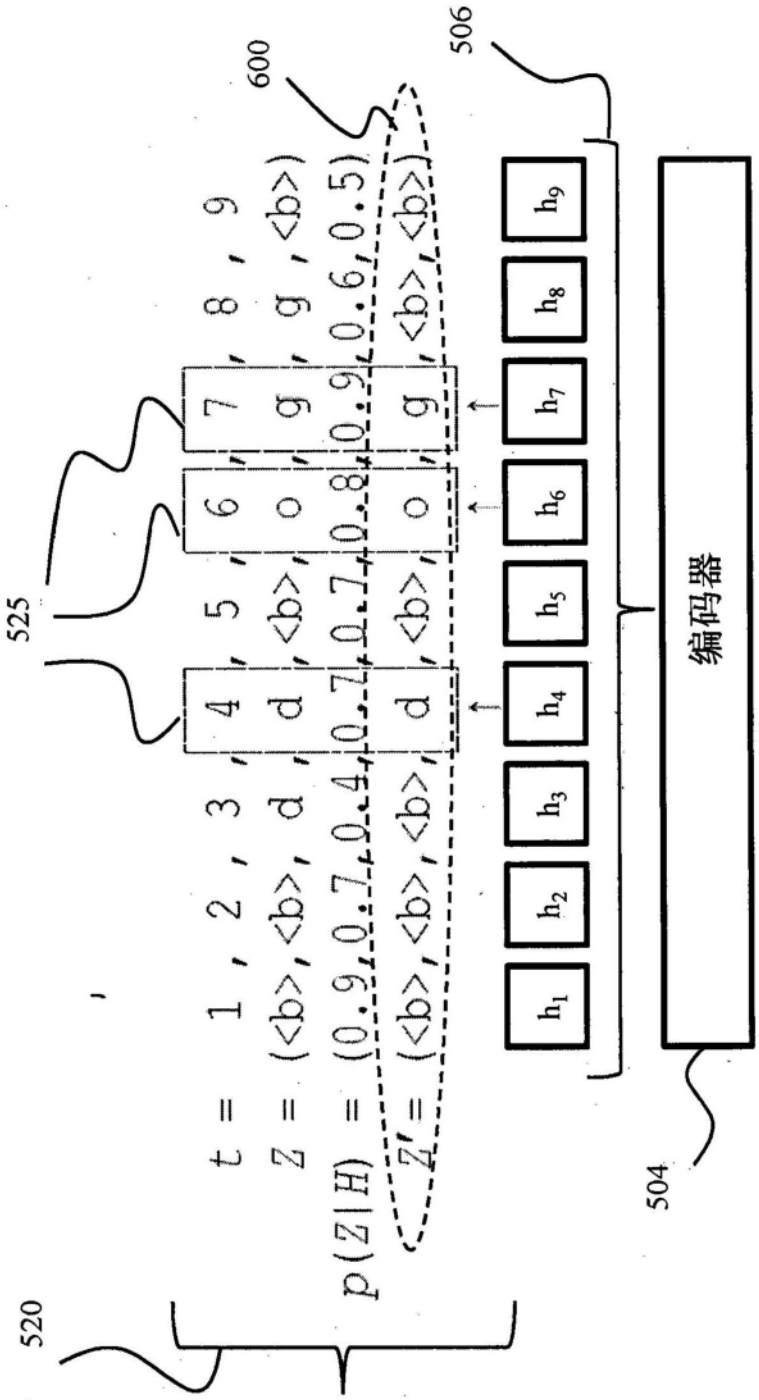

520
525
600
506
504
t = 1 , 2 , 3 , 4 , 5 , 6 , 7 , 8 , 9
Z = (<b>, <b>, d , d , <b>, o , g , g , <b>)
p(Z|H) = (0.9, 0.7, 0.4, 0.7, 0.7, 0.8, 0.9, 0.6, 0.5)
Z' = (<b>, <b>, <b>, d , <b>, o , g , <b>, <b>)
h1
h2
h3
h4
h5
h6
h7
h8
h9
编码器

图6A

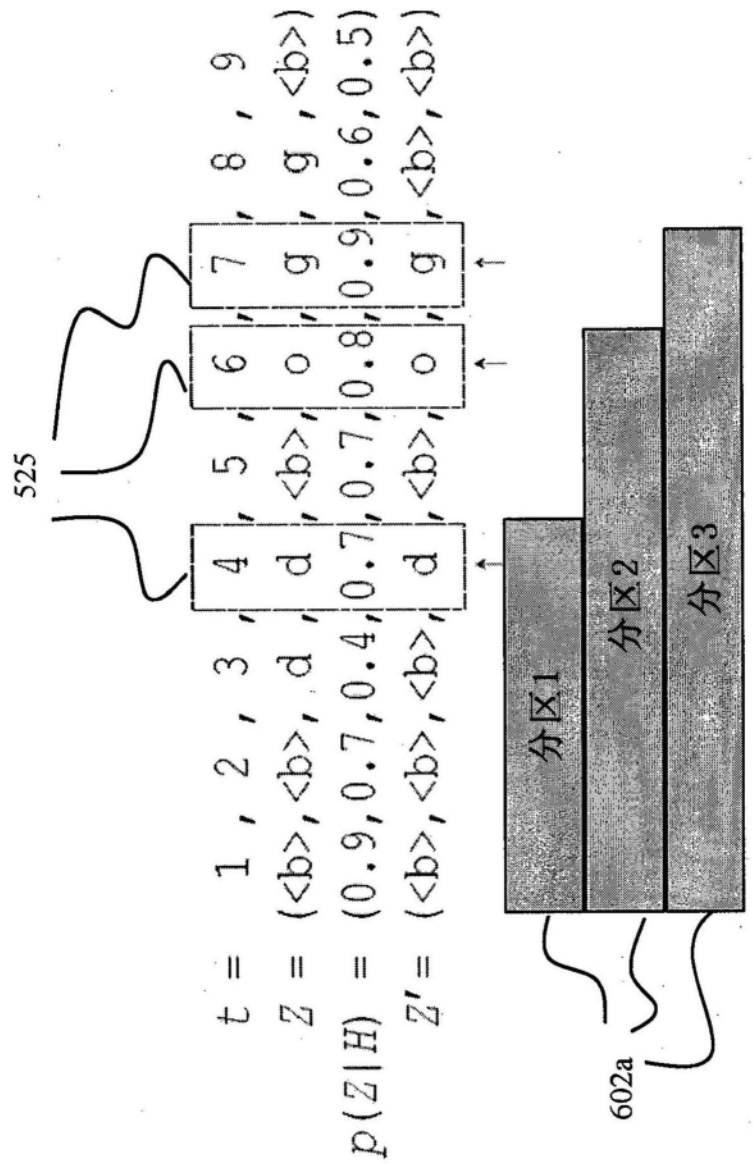

t = 1 , 2 , 3 , 4 , 5 , 6 , 7 , 8 , 9
Z = (<b>, <b>, d , d , <b> , o , g , g , <b>)
p(Z|H) = (0.9, 0.7, 0.4, 0.7, 0.7, 0.8, 0.9, 0.6, 0.5)
Z' = (<b>, <b>, <b>, d , <b> , o , g , <b>, <b>)
525
分区1
分区2
分区3
602a

图6B

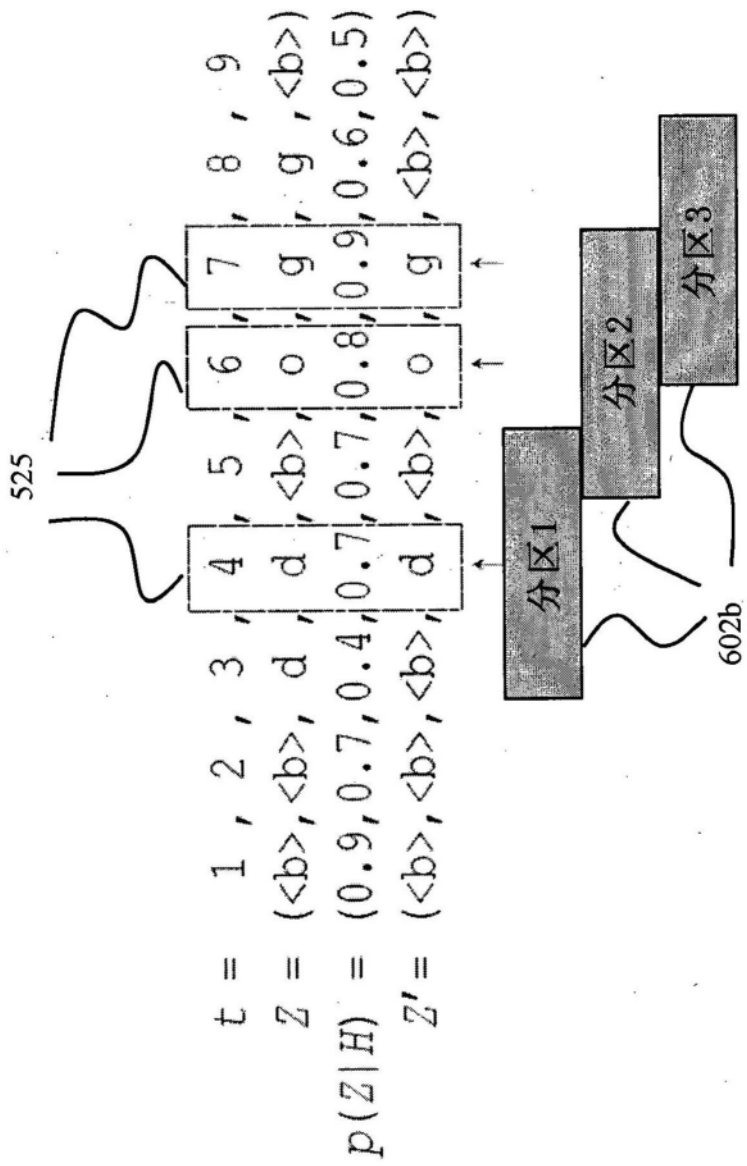

t = 1 , 2 , 3 , 4 , 5 , 6 , 7 , 8 , 9
Z = (<b>, <b>, d , d , <b> , o , g , g , <b>)
p(Z|H) = (0.9, 0.7, 0.4, 0.7, 0.7, 0.8, 0.9, 0.6, 0.5)
Z' = (<b>, <b>, <b>, d , <b> , o , g , <b>, <b>)
525
分区1
分区2
分区3
602b

图6C

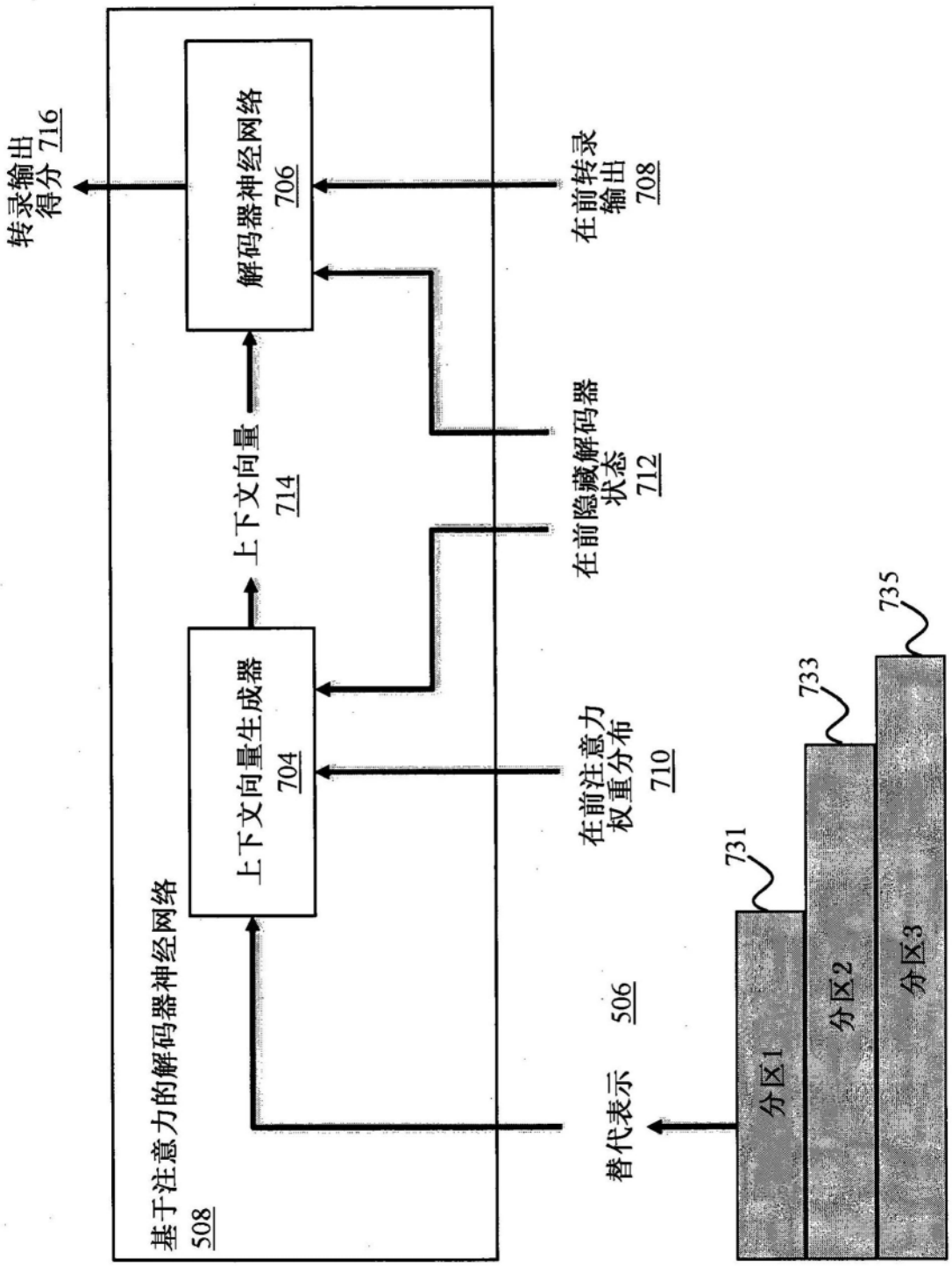

转录输出得分 716
解码器神经网络 706
在前转录输出 708
上下文向量 714
在前隐藏解码器状态 712
上下文向量生成器 704
在前注意力权重分布 710
基于注意力的解码器神经网络 508
替代表示 506
731
733
735
分区1
分区2
分区3

图7

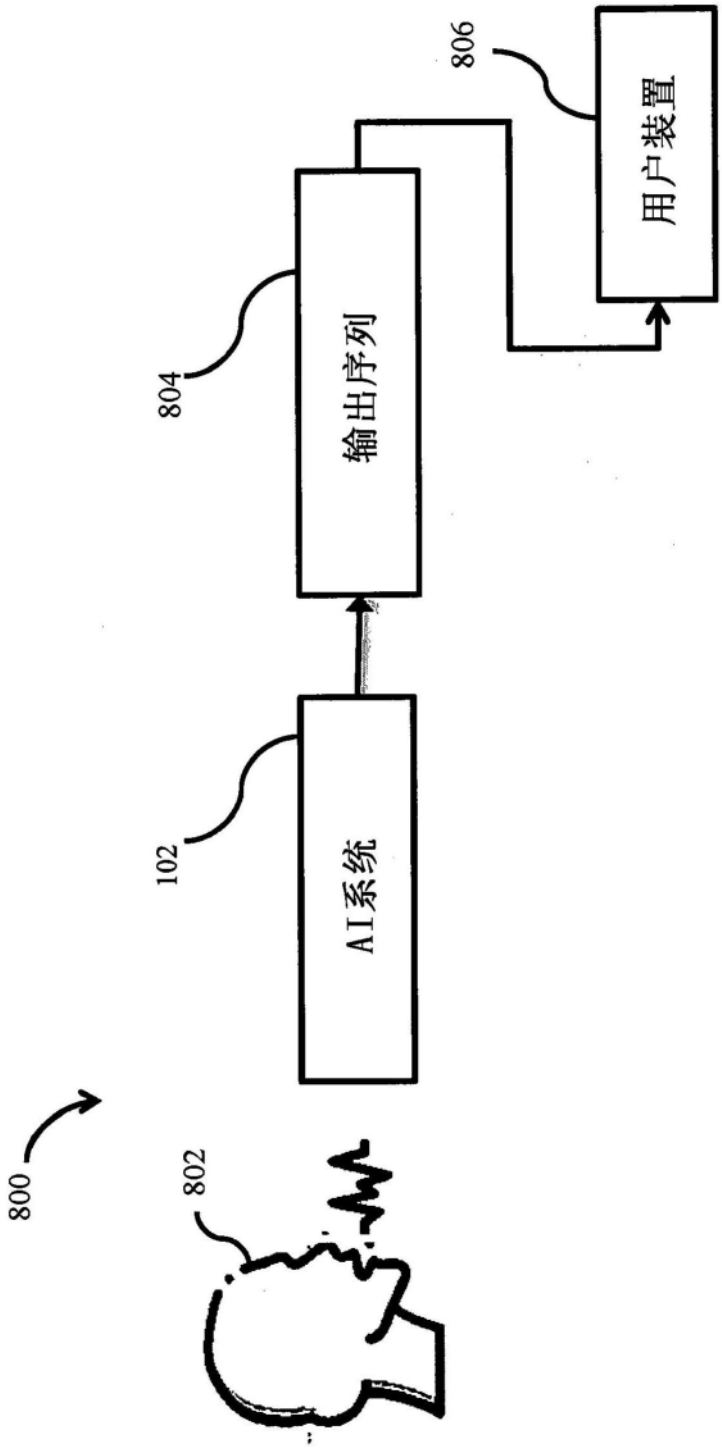


图8

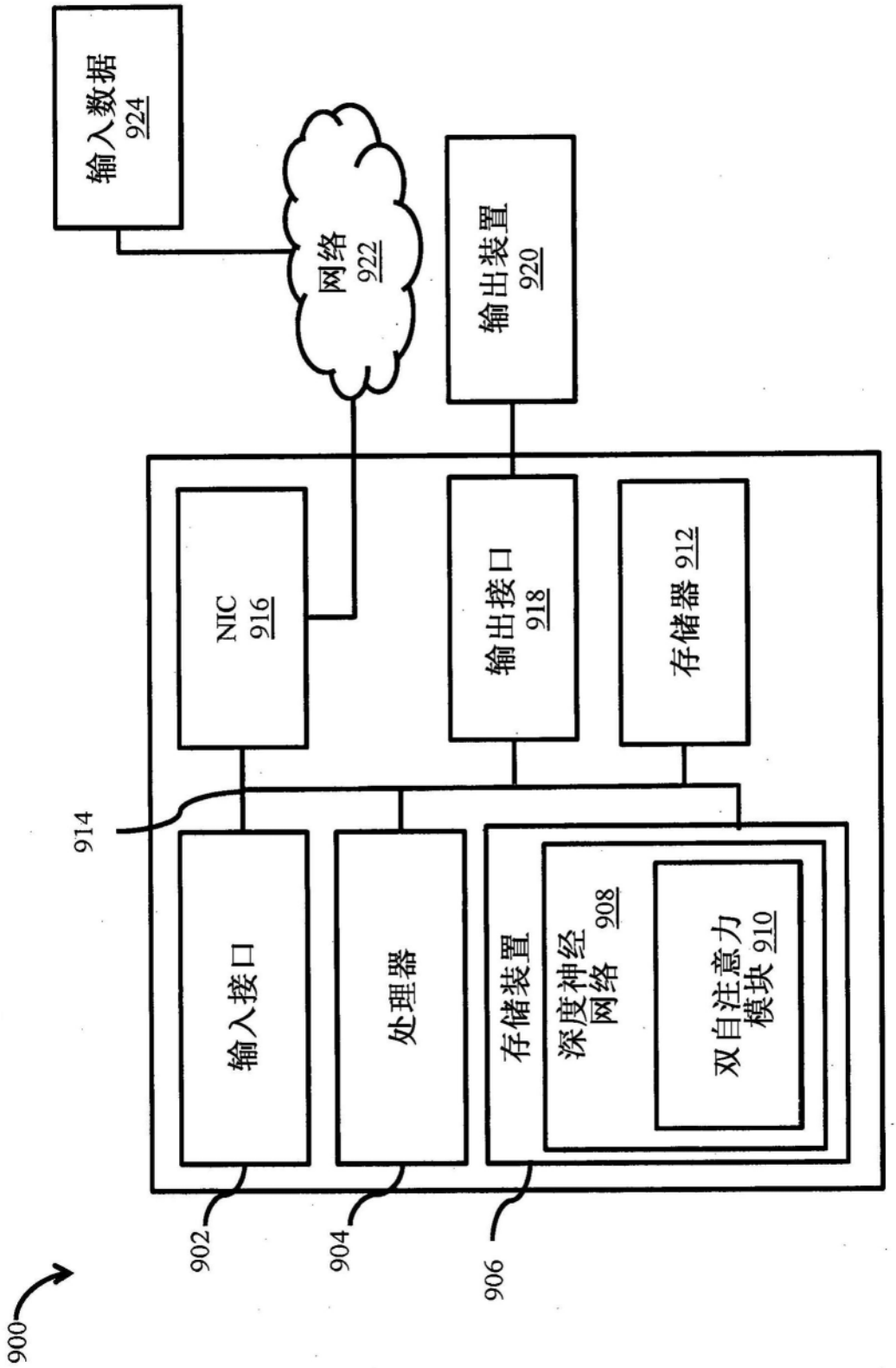

输入数据
924
网络
922
输出装置
920
NIC
916
输出接口
918
存储器 912
914
输入接口
处理器
存储装置
深度神经
网络 908
双自注意力
模块 910
902
904
906
900

图9