

Algorithmic Representation and Use Examples of Interface Intelligence Models

Yebin Lin, Yemin Lin

Hangzhou Yihu Intelligent Technology Co., Ltd., Hangzhou Zhejiang
Email: 358477162@qq.com

Received: Oct. 23rd, 2018; accepted: Nov. 7th, 2018; published: Nov. 14th, 2018

Abstract

Interfaces are common in our lives, such as In, out of data input and output, ports where items are connected to each other. These values are not very good. If they are represented by logic, they are very simple and unconnected logic, but the whole of them can be very diverse. Then the text tries to use an interface-based intelligent model to calculate the diversified whole of this interface. The following is an explanation of the feasibility and practicability of the interface intelligent model from the perspectives of algorithm representation, example description, and future prospects.

Keywords

Artificial Intelligence, Interface Driver, Algorithm Model, Interface Composition

接口智能模型的算法表示和使用例子

林焯斌, 林焯敏

杭州蚁护科技有限公司, 浙江 杭州
Email: 358477162@qq.com

收稿日期: 2018年10月23日; 录用日期: 2018年11月7日; 发布日期: 2018年11月14日

摘 要

接口在我们生活中很常见, 比如数据输入输出的In, out, 物品相互连接的端口等。这些用数值不太好表示, 用逻辑来表示的话, 又属于很简单连接和不连接的逻辑, 但他们形成的整体也可以是很多样化的。那么文本试着用一个以接口为基础的智能模型, 来计算这种接口形成的多样化的整体。下面就从算法表示, 例子说明, 未来展望这几个角度来说明接口智能模型的可行性和实用性。

关键词

人工智能, 接口驱动, 算法模型, 接口构成

Copyright © 2018 by authors and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

目前机器学习领域算法的原理, 大多可以归纳为通过权值的学习来适应应用所需的结果。目前流行的人工智能算法, 都是对数据进行运算处理, 让机器学习到相应的权值, 最终能做出准确的预判。本文中的“选择模式”旨在研究一种以接口为基础的模式选择方式。在《以接口为基础的智能模型》的论文中, 已经大致说明接口的形成也可以作为一种选择模式[1]。个体可以通过接口相互连接形成一个整体, 而这个连接的过程也是可以调整的。通过选择接口的形成, 从而让个体组合成我们想要的结果个体。下面从算法方面和它的用途, 进一步阐述它的原理和使用价值。

2. 数组表示

在接口模型中, 用“正”和“负”来表示两种相异的接口类型[2]。负的代表接口的接收一方, 可以把它想象为积木的凹槽; 正的代表接口给予的一方, 可以把它想象为积木的凸槽。这里用接口的名字及正负符号来定义一个接口实例。例如可以用 $-x$ 表示一个接口实例, 其中 x 表示实例名称, 符号表示接口类型。个体是包含一个或多个接口的实例, 比如个体 I 的接口 $[x,-x,x]$, 表示着个体 I 具备3个 x 轴的接口, 2个凸出来的接口, 一个凹进去的接口。

个体的接口用带正负号来表示之后, 他们之间的连接, 相当于正的对上负的, 一种相互对应的关系。用数组来表示的话, 就是如下所示:

$[x,-x,x]$

$[-x,x,-x]$

这样可以很清晰地看出接口之间的相互连接对应关系。

接口之间的相互连接, 计算上就相当于 $x + (-x) = 0$ 。当然虽然结果等于0了, 并不是说什么都没有, 而是指这两个接口已经相互结合了, 没有多余接口的意思。假设 $I1$ 拥有接口 $[x,-x,x]$, $I2$ 拥有接口 $[x,x,-x]$, 那么他们之间的连接就可以表示为:

$I1: [x,-x,x]$ —个体拥有的接口

$I1: [x,0,0]$ —个体活跃的接口

$I1: [-x,x]$ —个体参与连接的接口

$I2: [x,-x]$ —个体参与连接的接口

$I2: [x,0,0]$ —个体活跃的接口

$I2: [x,x,-x]$ —个体拥有的接口

上面的0也同样表示了接口的状态。原本活跃的接口, 通过接口连接变成了不活跃的接口。如果相互连接的接口在某些条件下断开的话, 就如同图1中, 接口中间连接的线断开, 则所有接口恢复为活跃接口, 等待与其它接口的结合。

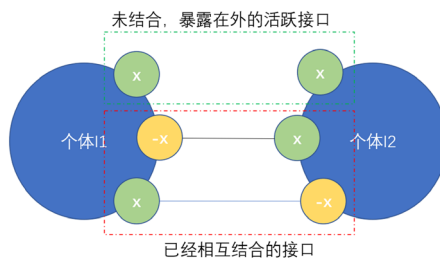


Figure 1. Schematic diagram of two individuals
图 1. 两个个体结合示意图

接口连接的计算, 就可以用上面数组的方式来进行求和运算。比如存在目标接口 $[x, -x, x, x, -x]$, 可以连接的接口单元有 $[-x]$, $[x]$ 。那么目标接口连接的达成, 就可以由这些接口单元与目标接口相互连接, 直到目标活跃接口全为 0。当然目前的接口并不一定都是单一的对应, 也可以是相互连接传递了几层的连接。比如与目标接口相连接的接口单元也是自带接口的, 那么就会一层层地相互连接下去。如下图 2 所示:

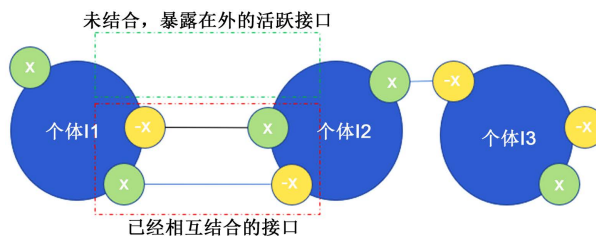


Figure 2. Individual multilayer connection
图 2. 个体多层连接

- I1: $[x, 0, 0]$ —个体活跃的接口
- I1: $[-x, x]$ —个体参与连接的接口
- I2: $[x, -x]$ —个体参与连接的接口
- I2: $[x, 0, 0]$ —个体活跃的接口
- I2: $[x]$ —个体参与连接的接口
- I3: $[-x]$ —个体参与连接的接口
- I3: $[0, x, x, -x]$ —个体活跃的接口
-

通过上面的算法表示, 可以看出只要拥有了目标的接口和接口单元。就可以形成跟目标接口相互匹配的接口整体。这个整体可以是单向的像基因链一样的连接, 也可以是叠加的像身体一样的构造。那么就意味着, 我们只需要提供目标接口, 这个算法就可以形成一个我们需要的整体。比如: 需要采摘远处的一个苹果。只需要把苹果单作一个形状凸的目标接口, 通过相互连接的接口单元。可以形成一个从起始点开始, 到苹果位置的手臂, 并且拥有可以包住苹果的凹形的手或者嘴巴的形状。

当然想要达到上面的效果, 还是需要更加深入地讨论具体怎么计算和选择接口单元形成的整体。接口的选择, 除了是互补的接口之外还存在其它条件的限制。比如我们平时熟悉的, 2 个插在一起的积木, 可以通过加大力气来把它们分开。因为地面的存在, 你也无法把积木往地板里面堆积。更加甚至的, 只是你的一个整体表面整齐的想法, 不让个体超过边缘也可以成为一个整体的选择条件。也就是在接口形成的基础上, 也还存在着各种条件来限制接口的形成。那么就把这些条件作为选择的条件。每个接口的

形成都会附带这些选择条件的参数, 而我们也只需要提供这些选择的条件, 就可以选择我们所要的接口整体。比如上面的摘苹果的例子, 那么形成的整体就需要能够承受这个苹果带来的力。整体的表示也需要加上它所带的值。如下所示:

I1: [-x,x]—个体参与连接的接口

I1: [“power”,20,20]—个体接口最大承受的值

接口拥有相应条件的值之后, 就可以根据实际情况来判断它是否断开和连接。比如采摘苹果需要大于 40 N 的力, 那么这个接口就无法直接承受这个力。但它可以通过跟其它的接口单元相互连接来分担这个力, 从而达到能够承受采摘苹果的力。其它的值比如个体的地点坐标, 我们可以定义不能够超出什么位置来限定形成的形状。也就是接口的形成, 首先需要遵守的是它们之间是否存在相反接口的规则, 然后再根据接口上所带的各种条件来选择形成的整体。

接口模式的计算和表示如上 [图 3](#) 所示。其中第 1 行表示已经连接的接口, 第 2 行表示可以连接的接口, 第 3 行表示全部的接口, 第 4 行表示全部接口对应的属性, 第 5 行表示可以连接接口对应的属性, 第 6 行表示已经连接接口对应的属性。

```

1 ||0,0,0,-y||x,-x,y,-y||x,-x,y,0||0,-x,y,0||0,-x,0,-y||x,0,0,0||x,0,0,0||0,0,0,0||0,0,0,-y||0,0,y,-y||x,-x,y,
2 ||0,0,0,0||0,0,0,0||0,0,0,-y||0,0,0,0||0,0,0,0||0,0,0,0||0,0,0,0||x,-x,0,0||0,-x,0,0||0,0,0,0||0,0,0,0,
3 ||0,0,0,-y||x,-x,y,-y||x,-x,y,-y||0,-x,y,0||0,-x,0,-y||x,0,0,0||x,0,0,0||x,-x,0,0||0,-x,0,-y||0,0,y,-y||x,-x,
4 ||area,4,0/:orientations,0,0,0,0||area,0,0/:orientations,0,0,0,0||area,0,0/:orientations,0,0,0,0||area,0,0/:
5 ||area,4,0/:orientations,0,0,0,0||area,0,0/:orientations,0,0,0,0||area,0,0/:orientations,0,0,0,0||area,0,0/:
6 ||area,4,0/:orientations,0,0,0,y||area,4,1/:orientations,x,-x,-y,y||area,5,1/:orientations,x,-x,y,0||area,4,
7
```

Figure 3. Result representation of the interface model

图 3. 接口模型的结果表示

3. 形状形成的使用

对于接口模型来说, 最直观的就是类似于原子一样, 通过原子键形成的各种物质。甚至是庞大有序的生命体。而现实生活中, 接口的连接最类似的就是带有凹槽和凸出的积木。我们平时生活中, 会用这些积木组装成各种形状, 类似于高达啊, 船啊等的模型。用算法来表示这个过程的话。比如需要形成一个圆形, 我们一般的想法是把积木单纯地堆积上去, 整体看上去像个圆形。但这样是很不稳定, 可能下次调整, 就会使整个模型崩溃。更别提堆积成摩天大楼。而接口智能模型则可以完成这个任务, 它会遵守个体的接口连接的前提下, 慢慢地向圆形靠拢。当它的外形到达类似圆形的时候, 它的内部的接口也都是比较牢固地连接着, 不会轻易崩塌。甚至只要遵守一些重力的原则, 那么相互连接形成摩天大楼, 也会是一个稳定的整体, 而不是一个容易涣散的整体。

接口智能模型可以把这些基础的积木, 搭建形成我们想要的形状。在此之前首先需要定义各种积木的接口, [x], [-x,x], [-x,x,y], [-x,y], [y,x]。那么定义好之后, 就可以根据需求组合成为一个想要的形状。比如想要组合成为一个圆形, 那么把选择关系定义为, 圆形边框内聚集越多的积木, 圆形边框外聚集越少的积木来选择这个形成的整体。那么最终就会得到我们想要的圆形的整体。数组表示的接口如下:

接口单元:

1. -x,y; 2. x,y; 3. -x,-y; 4. x,-y; 5. x,-x,x; 6. -x,x,-x; 7. x,-x; 8. y,-y; 9. y,-y,y; 10. -y,y,-y;

11. x,-x,y,-y

判断条件:

- 1) 以(4, 4)为圆心, 半径 4 的圆形。
- 2) 园内的相互连接的积木个体数量多。

期望结果:

积木相互连接成圆的形状。

运算过程:

- 1) 从点(4.0)开始, 随机选择带积木连接。
- 2) 再选择积木跟之前形成的整体连接。执行 100 次, 直到没有多余的接口可以连接。
- 3) 判断圆形内没有连接的点, 并且以该点为圆心, 向外辐射寻找最近的连接积木。找到后, 打破该积木与其它积木连接接口。
- 4) 在打破的地方开始, 重新随机选择积木连接。
- 5) 重复执行 3, 4 的步骤 100 次, 直到积木覆盖完整的圆。

计算结果形成的结果如图 4, 图里红色三角形代表凸的接口, 绿色三角形代表凹的接口, 黑色小点代表积木(个体)。他们相互连接, 形成圆形的整体。

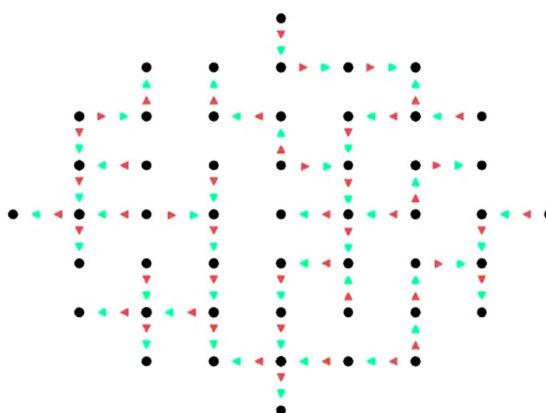


Figure 4. Circular interface overall

图 4. 圆形的接口整体

这里需要说明下为什么这么做。对比神经网络模型, 调整的是权重的大小[3]。但接口模型, 只有接口的相互连接, 想要调整也是调整接口之间的连接。接口之间的连接也只有连接不连接这 2 种选择。但接口可以跟不同的个体连接, 所以调整的其实是跟不同的个体连接, 而不是连接或不连接。那么为了获得想要的连接个体, 就要把之前不好的连接给去掉, 然后在重新匹配新的连接个体。直到匹配到需要的连接个体。所以例子里最主要的运算就是, 判断没有覆盖到的地方, 然后打破最近的积木, 重新形成可以覆盖完全的积木连接。也就是神经网络里面调整的权值, 接口模式里面调整的是接口的不同形成。那么例子里的正向的校验, 就是随机形成整体。然后反向的校验是通过没有覆盖的点来进行附近的积木打破和重新形成。

这个调整是和神经网络较大的不同点。相比较神经网络的训练[4], 接口模型更多的是生成和破坏。即使用数学来表示, 也只是简单的 1, -1。但它形成的整体, 也是一个比较严密的过程, 需要不断的调整。

其它的形状也是一样, 长方形, 三角形, 甚至是各种形状的模式。都可以通过这些积木来进行拼接, 最后选择最佳的。可以制作一个类似于画板的界面, 在这个界面上画各种形状, 然后这些接口单元就可以形成这些形状的个体。当然因为是接口模型, 所以两个整体之间也是可以连接的。比如你画了一个圆形的头, 然后又画了一个身体。那么头和身体, 可以在相接触的地方进行连接, 从而形成一个更大的整体。由此可以想象, 它可以组成各种复杂的事物。

4. 未来期望可做的

4.1. 函数方法的自动构建

接口可以很容易联想到程序中的一个类对象, 这些类对象由很多接口及方法进行暴露。这些类接口和方法遵循输入输出的接口定义。那么把这些函数 in, out 作为接口模型的接口, 就可以进行函数的构建了。比如: 在一个界面上显示一张图片。

接口单元:

界面上的位置 A1: [-picture_url]

输出图片的函数 I1: [picture_url][-banner_picture], I2: [picture_url][-icon_picture]

图片的获取的函数: P1: [banner_picture], P2: [icon_picture]

那么就可以根据函数的接口, 进行函数的匹配和相互的连接。然后每个函数的 in, out 的参数, 还可以从其它接口单元里提取可以进行连接的。直到中间部分的接口都为 0, 那就构建了满足输入输出要求的一个可以运行函数。

4.2. 语言识别的运用

语言识别里如果把不同的语言定义自己的接口, 那么就可以形成意思完整的对话。比如: 把语言分为主, 谓, 宾的接口, 那么获取到信息“木头”、“火”的时候, 就可以形成完整的语句。

接口单元: 主语: [-名词, 主语, -谓语], 谓语[-动词, 谓语, -主语, -宾语], 宾语[-名词, 宾语]

目标接口: 木头[名词]火[名词]

选择的过程:

- 1) 根据目标接口, 提取主语。然后主语根据接口提取谓语, 谓语根据接口提取宾语。
- 2) 主语名词, 提取木头或者火。谓语动词根据逻辑意思提取, 着[动词], 燃烧[动词], 宾语提取名词。
- 3) 输出结果: 木头着火, 火燃烧木头。

期望结果:

木头着火

当然这样只是比较简单的表示, 但是复杂的也是一样的, 只要定义足够多的接口。就可以形成逻辑严密, 整体完善的语句段落。

4.3. 图像识别上的使用

在图像识别上, 如果能够把图像拆分成可以相互组合的块。那么就可以进行一种组合识别的模式。结合胶囊神经网络的把图像分为小的识别块[5]。然后接口模型给这些识别块加上接口, 就可以组合识别了。比如识别猫的图片:

接单单元:

猫的尾巴[尾巴][-身体接口][-毛]

猫的身体[身体][-尾巴][-毛][-头]

猫的头[头][-身体][-眼睛][-鼻子][-耳朵][-嘴巴]

眼睛[眼睛][-头]

鼻子[鼻子][-头]

耳朵[耳朵][-头]

嘴巴[嘴巴][-头]

那么在识别到猫的尾巴的部分之后, 根据尾巴的接口, 就可以推导出身体。然后附近识别出是否有身体。脸也是, 识别出眼睛后, 就会寻找头的形状。找到之后, 就会找鼻子, 耳朵, 嘴巴。这样识别的好处就是, 即使猫只露出身体和尾巴, 但根据接口模型是相互匹配的。那么也可以识别成为猫。而一些误差识别的, 比如拥有眼睛条纹的蝴蝶, 狗, 也可以根据接口模型, 识别出不是猫的部分而判断出不是猫。甚至也可以修改相应的接口, 进行推导学习不同的物种。比如无毛猫, 那么就把猫的其它特征接口都有, 只是没有毛的部分。那么即使没有给它训练无毛猫的图片, 它也可以根据其它猫的模式, 来进行识别。根据是否有毛来区分跟其它猫的条件, 然后是否是猫的部分来区分猫和其它动物。图像识别加入接口模型的话, 就可以进行一种推导识别的模式。

5. 总结

本文通过正负号来表示接口智能模型的接口两种类型, 并且展示他们之间的数组的运算规则。初步通过一个简单的例子来说明接口智能模型的运算过程。展示接口智能模型在通过接口形成整体上有着一一定的优势。而这个整体的形成可以在函数的构建, 语句的编写, 图像推导识别上使用。

参考文献

- [1] 林焯斌. 以接口作为基础的智能模型[J]. 人工智能与机器人研究, 2018, 7(3): 129-134.
- [2] 林焯斌. 如何让筛选一直持续下去[M]. 北京: 中国社会出版社, 2018.
- [3] [德]海德格尔. 存在与时间[M]. 陈嘉映, 王庆节, 译. 北京: 三联书店, 1999.
- [4] Oz, C. and Leu, M.C. (2011) American Sign Language Word Recognition with a Sensory Glove Using Artificial Neural Networks. *Engineering Applications of Artificial Intelligence*, No. 4, 1204-1213.
- [5] 张博宇. 网络背景下人工智能技术的应用[J]. 科技资讯, 2016(36): 18-20.

知网检索的两种方式:

1. 打开知网页面 <http://kns.cnki.net/kns/brief/result.aspx?dbPrefix=WWJD>
下拉列表框选择: [ISSN], 输入期刊 ISSN: 2326-3415, 即可查询
2. 打开知网首页 <http://cnki.net/>
左侧“国际文献总库”进入, 输入文章标题, 即可查询

投稿请点击: <http://www.hanspub.org/Submission.aspx>

期刊邮箱: airr@hanspub.org