(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 107506047 A
(43)申请公布日 2017.12.22

(21)申请号 201710761127.9

(22)申请日 2011.09.29

(30)优先权数据
1016385.5 2010.09.29 GB

(62)分案原申请数据
201180053255.9 2011.09.29

(71)申请人 触摸式有限公司
地址 英国伦敦

(72)发明人 本杰明·麦德洛克
道格拉斯·亚历山大·哈珀·欧

(74)专利代理机构 永新专利商标代理有限公司
72002
代理人 赵腾飞 王英

(51)Int.Cl.
G06F 3/023(2006.01)

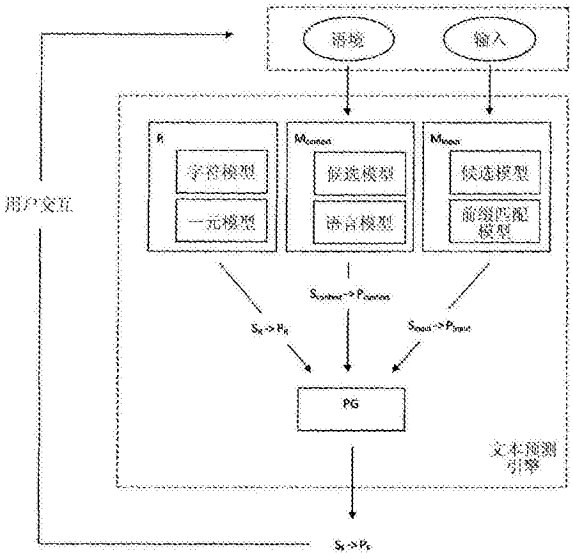
权利要求书2页 说明书20页 附图2页

(54)发明名称

用于向电子设备输入文本的文本预测引擎、系统及方法

(57)摘要

一种文本预测引擎、一种包含文本预测引擎的系统以及用于生成序列预测的方法。所述文本预测引擎、系统和方法生成一组各带有相关概率值的最终的序列预测。



CN 107506047 A

1.一种文本预测系统,包括:

一个或多个处理器;

存储指令的存储器,所述指令当被所述一个或多个处理器执行时,配置所述一个或多个处理器:

基于第一证据源和第一模型生成至少一个第一序列预测,其中,所述至少一个第一序列预测中的每一者皆包括第一序列和第一相关概率估计;

基于第二证据源和第二模型生成至少一个第二序列预测,其中,所述至少一个第二序列预测中的每一者皆包括第二序列和第二相关概率估计;以及

耦合到所述一个或多个处理器或所述存储器中至少一项的显示器,所述显示器被配置为:

在文本输入图形用户界面内输出所述至少一个第一序列预测和所述至少一个第二序列预测中的至少一者。

2.根据权利要求1所述的系统,其中,所述第一证据源或所述第二证据源中的一者是基于呈现关于所述用户当前正在输入的词条的所观察证据的输入的,所述第一证据源或所述第二证据源中的另一者不是基于呈现关于所述用户当前正在输入的所述词条的所观察证据的所述输入的。

3.根据权利要求2所述的系统,其中,所述第一证据源是独立于所述第二证据源而被建模的。

4.根据权利要求3所述的系统,其中,所述第一模型不同于所述第二模型。

5.根据权利要求4所述的系统,其中,所述第一模型包括语境模型,并且所述第二模型包括输入模型。

6.根据权利要求4所述的系统,其中,所述第一证据源是独立于所述第二证据源而被建模的。

7.根据权利要求4所述的系统,其中,所述第一证据源由所述第一模型建模以生成所述至少一个第一序列预测,并且所述第二证据源由所述第二模型建模以生成所述至少一个第二序列预测。

8.根据权利要求5所述的系统,其中,所述输入模型包括候选模型和语言模型。

9.根据权利要求5所述的系统,其中,所述语境模型包括候选模型和前缀匹配模型。

10.根据权利要求1所述的系统,其中,所述第一模型包括语境模型,并且所述第二模型包括输入模型。

11.根据权利要求10所述的系统,其中,所述输入模型包括候选模型和语言模型。

12.根据权利要求10所述的系统,其中,所述语境模型包括候选模型和前缀匹配模型。

13.根据权利要求10所述的系统,还包括:优先模型,其被配置为生成带有相关概率估计的第三组序列。

14.根据权利要求1所述的系统,还包括:优先模型,其被配置为生成带有相关概率估计的第三组序列。

15.根据权利要求14所述的系统,其中,所述优先模型包括一元模型。

16.根据权利要求14所述的系统,其中,所述优先模型包括字符模型。

17.根据权利要求1所述的系统,其中,所述第一证据源是独立于所述第二证据源而被

建模的。

18.根据权利要求17所述的系统，其中，所述第一证据源由所述第一模型建模以生成所述至少一个第一序列预测，并且所述第二证据源由所述第二模型建模以生成所述至少一个第二序列预测。

19.一种计算设备，包括一个或多个处理器和存储指令的存储器，所述指令当被所述一个或多个处理器执行时，配置所述计算设备：

基于第一证据源和第一模型生成至少一个第一序列预测，其中，所述至少一个第一序列预测中的每一者皆包括第一序列和第一相关概率估计；以及

基于第二证据源和第二模型生成至少一个第二序列预测，其中，所述至少一个第二序列预测中的每一者皆包括第二序列和第二相关概率估计；

所述计算设备还包括耦合到所述一个或多个处理器或所述存储器中至少一项的显示器，其中，所述显示器被配置为，在文本输入图形用户界面内输出所述至少一个第一序列预测和所述至少一个第二序列预测中的至少一者。

20.一种用于由计算设备预测文本的方法，所述方法包括：

由所述计算设备基于第一证据源和第一模型生成至少一个第一序列预测，其中，所述至少一个第一序列预测中的每一者皆包括第一序列和第一相关概率估计；

由所述计算设备基于第二证据源和第二模型生成至少一个第二序列预测，其中，所述至少一个第二序列预测中的每一者皆包括第二序列和第二相关概率估计；

在通信性地耦合到所述计算设备的显示设备上，在文本输入图形用户界面内输出所述至少一个第一序列预测和所述至少一个第二序列预测中的至少一者。

用于向电子设备输入文本的文本预测引擎、系统及方法

[0001] 本申请是申请日为2011年9月29日、申请号为201180053255.9的发明专利的分案申请。

技术领域

[0002] 本发明主要涉及一种用于向电子设备输入文本的文本预测引擎、系统及方法。

背景技术

[0003] 一些现有的发明利用多种不同技术提供了电子设备用户文本输入的改善方法。然而，众所周知，已公开的相关系统首先面临着使用稳定且完全整合的概率模型预测用户预期写入文本的问题。

发明内容

[0004] 在本发明的第一方面中，提供了一种文本预测引擎，包括：至少一个模型，其用于从证据源中生成带有相关概率估计的第一组序列；概率生成器，其用于接收带有相关概率估计的所述第一组序列并生成一组带有相关概率值的序列预测；其中，在给定由所述概率生成器接收到所有可能的序列的情况下，在由所述概率生成器生成的所有可能的序列预测上归一化所述概率值。

[0005] 优选地，所述文本预测引擎包括优先模型，其用于生成带有相关概率估计的第二组序列。

[0006] 优选地，所述模型根据所述证据源以及所述证据源中的不确定性生成第一组序列。优选地，所述概率生成器用于接收带有相关概率估计的所述第一、第二组序列。

[0007] 所述概率生成器优选地通过将n个最可能的序列预测同剩余可能的序列预测的概率值代表常量相加，估计所述概率值的归一化因数。所述常量表示由所述模型和所述优先模型生成的剩余的可能序列预测的概率值。

[0008] 所述模型包括用于生成带有相关概率估计的多个第一组序列的多个模型。在一实施例中，所述多个模型根据多个证据源生成多个第一组序列。

[0009] 优选地，所述文本预测引擎是某一系统的一部分，而所述用户输入文本通过一个或多个用户选择、字符输入或语音识别被输入至该系统中。

[0010] 所述文本预测引擎根据相应的模型包括给定的语境序列的概率对所述序列预测的概率值进行加权。在一实施例中，所述多个模型包括与多种不同语言相对应的多个语言模型；而所述文本预测引擎对与涉及到用户输入文本的最可能语言的语言模型相对应的序列预测的概率值进行最高级的加权。

[0011] 各证据源由用于生成带有相关概率估计的序列的对应模型塑造。在给定所述序列预测的情况下，所述概率生成器优选地将各证据源作为其他所有证据源的有条件独立体处理。

[0012] 在所述文本预测引擎的一优选实施例中，所述模型包括语境模型和输入模型，所

述语境模型和所述输入模型用于接收用户输入的文本并生成一组序列和相关的概率估计；而且所述优先模型包括用于生成一组序列和相关概率估计的目标优先模型。所述输入模型优选包括候选模型和语言模型。所述语境模型优选包括候选模型和前缀匹配模型。所述目标优先模型优选包括字符模型和一元模型。

[0013] 在本发明的第二方面中，提供了一种系统，包括：用户界面，其用于接收由用户输入的文本；文本预测引擎，其用于接收从所述用户界面输入的所述文本并生成一组带有相关概率值的序列预测，其中，在的所有可能的序列预测上归一化所述概率值；其中，所述文本预测引擎还用于向所述用户界面提供所述序列预测。

[0014] 优选地，所述输入模型包括候选模型和语言模型。优选地，所述语境模型包括候选模型和前缀匹配模型。优选地，所述目标优先模型包括字符模型和一元模型。

[0015] 在本发明的第三方面中，提供了一种处理用户文本输入的方法，包括：接收输入至用户界面的文本；使用文本预测引擎生成一组序列预测和相关的概率值，其中，在所有可能的序列预测上归一化所述概率值；将所述序列预测提供给所述用户界面。

[0016] 生成归一化概率值的步骤优选包括：通过将n个最可能的序列预测的概率值同剩余的可能序列预测的概率值代表常量相加，估计所述概率值的归一化因数。

[0017] 该方法还包括：将所述序列预测显示在所述用户界面上以供用户选择。优选地，通过所述文本预测引擎对所述序列预测进行排序，以供所述用户界面进行有序显示。仅当所述序列预测的对应概率值大于或等于第一阈值时，将所述序列预测提供给所述用户界面。类似地，仅当所述序列预测的对应概率值大于或等于第一阈值时，上述系统将所述序列预测提供给所述用户界面。

[0018] 优选地，所述序列预测中的至少一个相当于由用户输入至所述用户界面的文本的调整或修正版本。

[0019] 所述方法还包括：自动输入具有大于第二阈值或在第二阈值之上的概率值的序列预测。类似地，在一实施例中，上述系统自动输入具有大于第二阈值或在第二阈值之上的概率值的序列预测。

[0020] 本方法中使用的概率生成器优选包括用于生成一组序列预测和相关概率值并根据相应的模型包括给定的语境序列的概率加权所述概率值的多个模型。

[0021] 本发明还提供了一种计算机程序，包括：计算机可读介质，其中存储有用于使处理器执行上述方法的计算机程序。

[0022] 本发明还涉及了一种用于生成序列预测的文本预测引擎以及用于生成序列预测以供显示和用户选择的系统及方法。在一实施例中，本发明涉及自动修正错误输入序列的系统及实现这一修正的方法。在一优选实施例中，本发明提供了一个文本预测引擎和通过结合任意个序列预期的不同概率估计生成带有相关概率值的一组最终序列预测。本发明的文本预测引擎、系统及方法由此可以提供基于任意独立证据源的预测。可通过向各预期序列分配正确的概率而非为序列排名，实现这一目的。通过分配正确的概率值，可分析分配给不同词条的概率的演化，并可以比较两个不同时间点上的给定词条或一组词条的概率。这意味着在特定预测中给定预设阈值“信任”(confidence)的情况下，可使用预设阈值来调节系统的行为。举例来说，仅显示预测到的序列，或者如果系统估计概率准确度超过0.75，或换句话说预测出的序列至少有75％的可能是准确的，此时进行自动修正。如果使用某种专

门的评价来为元素排名，诸如那些无法在时间点上的序列之间进行可靠比较的值，则这种推理是不能实现的。

[0023] 为了生成正确的概率值，本发明优选提供了一种有效估计所有序列归一化加和的方法。

[0024] 下面参考下列附图，详细介绍本发明。

附图说明

[0025] 图1为本发明高级预测结构的示意图；

[0026] 图2为本发明优选预测结构实例的示意图。

具体实施方式

[0027] 定义：

[0028] ●字符-表示基本的规范单元符号；

[0029] ●字符集-字符的有限集合；

[0030] ●序列-排好序的有限长度字符串；

[0031] ●前缀-如果起始于各序列中的首字符相同，并且存在连续的一一映射且length(s)≤length(s′)，则序列s为另一序列s'的前缀；

[0032] ●真前缀-如果序列s为序列s'的前缀，且length(s)<length(s′)，则序列s为另一序列s'的真前缀；

[0033] ●语言-具有特定书写或口头表述特性的一组序列(通常为无限的)；

[0034] ●文本-从一种或多种语言中提取的写入数据；

[0035] ●系统-本发明主题；

[0036] ●用户-与上述系统交互的人员。

[0037] 一般但不唯一，可参见图1实施本发明系统。图1是本发明的高级文本预测结构的方块图。本系统包括文本预测引擎，该文本预测引擎生成一组由用户预期输入的最可能的序列预测S_F。各序列预测具有一与其相关的概率值。

[0038] 如图1所示，文本预测引擎优选包括：多个训练过的将会被用于从多个证据源e_1、e_2、e_3等中作出概率推理的模型M_1、M_2、M_3……，以及概率生成器(PG，probability generator)。然而，在其他实施例中，可有单个训练过的模型和单个证据源。

[0039] 现存一些任意类型的潜在证据源e_1、e_2等。这些证据源的实例包括：

[0040] ●用户已输入的序列；

[0041] ●用户当前正在输入的词条(term)/短语(phrase)；

[0042] ●存储的由用户输入的历史序列；

[0043] ●用户的母语；

[0044] ●正输入的语言的特殊类型(style)；

[0045] ●正输入有当前序列的应用；

[0046] ●在消息发送环境中的目标消息接收者；

[0047] ●时间/日期；

[0048] ●作为本系统主机的设备的位置。

[0049] 通用模型

[0050] 本系统的目标在于，根据用户预期输入某一序列的可能性为给定的语言子集中的序列排名。在概率学中，这相当于为集合S中的序列排名，该集合受下列表达式支配：

[0051] $P(s \in S|e,M)$ (1)

[0052] 其中，e为观测到的证据，而M为将会被用于作出概率推理的经过训练的模型集合。换言之，本系统将给定证据e的情况下，在能够提取出预测的所有序列的集合上估计条件概率。目标序列表示为s。

[0053] 为了简化对来自于不同数据源的预测进行组合的过程，在一优选实施例中，将目标序列s定义为来自于特定数据源的预测。

[0054] M中的各模型于特定数据源上训练。因此，可由M中的模型表示特定数据源，而表达式(1)中的集合S涉及由M中的模型生成的所有不同词条(或序列)。通过查询模型来提供预测词条。该词条与其出处模型相关，并且因为该词条与其出处模型相关，因此其不同于出自其他模型但词法一致的词条。这种相关可隐含于上述数据中。然而，该词条可标记有与该词条出处模型相关的标识符。

[0055] 在这一优选的对预测进行组合的过程中，将出自不同数据源的另外两个相同预测视为不同。为了组合出自不同模型的序列以获得预测列表，通过移除重复的预测，将该序列进行简单排序。在这一优选操作中，对给定的词法词条/序列保存最可能估计(estimate)，并丢弃任何(可能性低的)词法重复。

[0056] 以一非限制性实例为例，如果M包括两个语境语言模型，法语(LM_{French})和英语($LM_{English}$)，词条“pain”可能出现在这两个模型中，并在S中出现两次，一次与法语模型有联系，而另一次与英语模型有联系。这样在给定一组特定证据(其中，在这种情况下，该证据为预测词条“pain”之前的语境)的情况下，对词条“pain”进行两次分开的估计。

[0057] 这些估计涉及两个不同的序列(一个出自法语模型，一个出自英语模型)，但因为其词法相同，无需全部呈现给用户。因此，根据本优选实施例，对给定的词法序列保存最可能估计，并丢弃掉任何词法重复。

[0058] 为了根据用户预期输入某一序列的可能性为给定的语言子集中的序列排名，需要计算表达式(1)中的条件概率$P(s \in S|e,M)$。为了确定这一概率，使用贝叶斯公式(Bayes' rule)重新排列该表达式如下：

[0059] $$\frac{P(e|s,M)P(s|M)}{P(e|M)} \qquad (2)$$

[0060] 并边缘化(marginalise)该表达式分母中的目标序列，由此得到：

[0061] $$\frac{P(e|s,M)P(s|M)}{\sum_{j=1}^{|S|}P(e|s_j,M)P(s_j|M)} \qquad (3)$$

[0062] 在优选实施例中，为了计算$P(e|s,M)$，假设：在给定上述目标序列的情况下，可将证据拆分成根据关联模型$[M_1 \cdots M_N]$下的某一分布分别求出的非重叠集合$[e_1 \cdots e_N]$。该独立性假设可写成：

[0063] $$P(e|s,M) = \prod_{i=1}^{N}[P(e_i|s,M_i \in M)] \qquad (4)$$

[0064] 并表述如下：

[0065] 假设1：在给定目标序列的情况下，能够将证据拆分为不同的集合，从而使各集合中的证据彼此间有条件地相互独立。

[0066] 其中，各e_i具有与其相关的模型M_i。这样，可以构建一框架，在该框架中能够以计算效率高的方式任意地组合多个证据源。在一优选实施例中，模型$R \in M$与目标序列优先(prior)相关。考虑到这一假设，我们可以重新表述表达式(3)如下：

[0067] $$\frac{P(s|R)\prod_{i=1}^{N}P(e_i|s,M_i)}{\sum_{j=1}^{|S|}P(s_j|R)\prod_{i=1}^{N}P(e_i|s_j,M_i)} \qquad (5)$$

[0068] 因此，在一优选实施例中，可通过计算目标序列优先$P(s|R)$以及各证据概率$P(e_i|s,M_i)$，计算出表达式(1)的条件概率。

[0069] 表达式(5)中的分母相对于s为恒定值，因此不会影响到排名，更确切地说其是计算出的概率值的归一化因数。在一优选实施例中，这一恒定值被估计为最可能序列的子集与一常量之和，以克服不得不计算S中所有序列的条件概率的问题(参见下文中的表达式13-15)。由于一些自然语象的Zipfian(齐普夫分布)特性，使这一方法合理化，在该特性中少数概率事件带有多数概率质量。齐普夫分布是幂次定律分布的一个实例，其中，给定事件的频率与其排名大致成反比。

[0070] 表达式(5)提供了将关于文本输入意向的不同证据源组合起来的原则方法，并在本发明的优选系统中，在给定有证据源e_1，e_2，…的情况下通过一组训练过的模型R，M_1，M_2，…生成一组序列S_R，S_1，S_2，…和一组相关的条件概率值P_R，P_1，P_2，…来实施表达式(5)。模型R用于计算优先目标序列概率$P(s|R)$，同时各模型M_1，M_2，…计算各证据概率$P(e_i|s,M_i)$。每个模型输出一组序列S_i和一组相关的条件概率P_i。每个模型M_1，M_2，…包括一个或多个子模型。概率生成器PG将序列和相关的条件概率作为输入，并输出一组与概率值P_F相关的最终序列S_F。概率生成器PG可以如上述优选处理那样将预测组合起来，也就是说，将预测按照概率次序排名，并仅删除掉任意的重复预测。与最终概率值P_F相关的该组序列S_F能够以诸如列表的形式呈现在本系统的用户界面上，以供用户浏览和选择。用户可通过做出预测选择或以其他方式操作包含本系统的装置，与本系统进行交互，因此可以更新证据。当文本输入至本系统时，可更新各模型R，$M_1 \cdots M_N$。

[0071] 本发明提供了的通过边缘化呈现在图形框架中的证据候选释义计算概率框架中的证据概率的两种优选方法，尽管还可以使用其他方法。下面，将介绍这两种优选方法。

[0072] 候选模型1

[0073] 在形成来自于单个证据源的证据的概率估计$P(e_i|s,M_i)$时，通常有帮助的是：以术语“候选”来表示模型，“候选”是介于‘用户预期(user-intended)’序列和观测到的证据之间的中间级。如果以候选来表示模型，则概率$P(e_i|s,M_i)$可重新表示为：

[0074] $$P(e|s,M)=\sum_{j=1}^{K}P(e|c_j,s,M_{candidate})P(c_j|s,M_{sequence}) \qquad (6)$$

[0075] 其中，c_j为单个候选，并且现在对于一给定证据源存在两个子模型M：候选模型$M_{candidate}$和序列模型$M_{sequence}$。此处的关键假设如下：

[0076] 假设2：在给定候选的情况下，上述模型下的概率可表示为候选的边缘化，其中，证

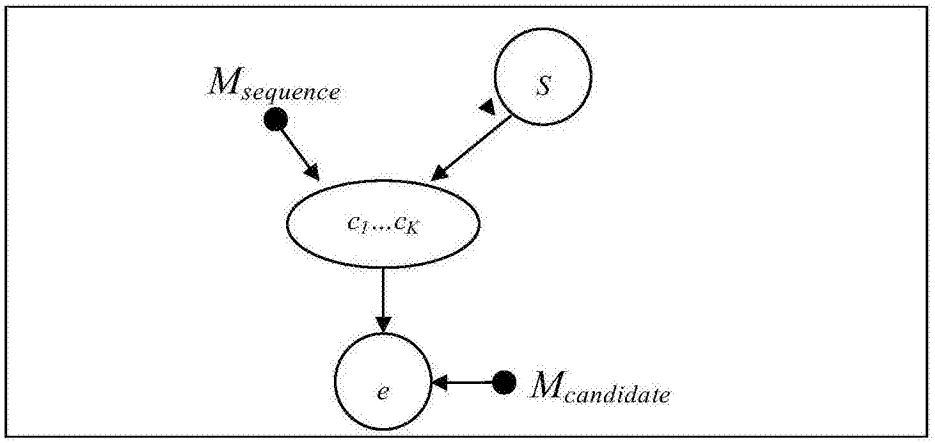
据有条件地独立于目标序列。

[0077] 应用这一假设，可从证据词条中删除对于s的依赖性：

[0078] $$P(e|s,M)=\sum_{j=1}^{K}P(e|c_j,s,M_{candidate})P(c_j|s,M_{sequence}) \quad (7)$$

[0079] 候选模型的属性还能够被编码成图形模型的形式，该图形模型描述了变量和模型之间的如下所示的关系：

[0080]



[0081] 候选模型2

[0082] 候选模型的另一变体首先使用贝叶斯公式转换证据概率：

[0083] $$P(e|s,M)=\frac{P(s|e,M)P(e|M)}{P(s|M)} \quad (8)$$

[0084] 在一实施例中，证据条件序列概率可重新表示为：

[0085] $$P(s|e,M)=\sum_{j=1}^{K}P(s|c_j,e,M_{sequence})P(c_j|e,M_{candidate}) \quad (9)$$

[0086] 其中，c_j为单个候选，而且与上述相同，对于一给定证据源存在两个子模型M：候选模型$M_{candidate}$和序列模型$M_{sequence}$。在该情况中，关键假设如下：

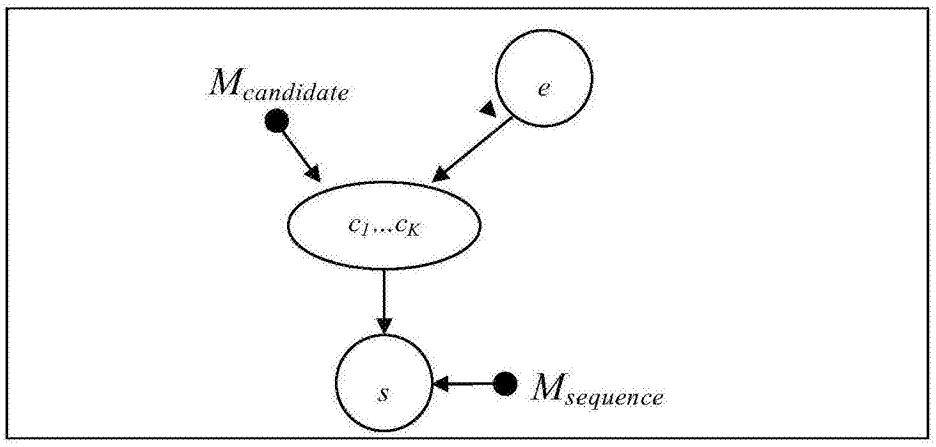
[0087] 假设3：在给定候选的情况下，上述模型下的概率可表示为候选的边缘化，其中，目标序列有条件地独立于证据。

[0088] 应用这种假设，可从证据词条中删除对于s的依赖性：

[0089]

$$P(s|e,M)=\sum_{j=1}^{K}P(s|c_j,M_{sequence})\,P(c_j|e,M_{candidate}) \quad (10)$$

[0090] 这一版本的候选模型的图形模型如下：

[0091] 

[0092] 而完整的证据概率为：

[0093] $$P(e|s,M)=\frac{\sum_{j=1}^{K}P(s|c_j,M_{sequence})P(c_j|e,M_{candidate})P(e|M)}{P(s|M)} \quad (11)$$

[0094] 特殊模型

[0095] 参照示出了预测引擎从两个不同源：语境(context)和输入(input)中提取证据的本系统优选实例的图2，我们利用一般的候选模型，提出了本系统的一个特殊实例。然而，如上所述，本系统不限于以语境和输入作为证据。如果使用了其他或额外的证据源，本系统将相应地根据此类证据源生成预测。

[0096] 通俗地讲，“语境”表示关于用户已输入文本的观测到的证据，而“输入”则表示关于用户当前正在输入文本的观测到的显证据。举例来说，如果用户已输入英文序列“My name is B”，则我们会认为语境证据为序列“My name is”，而输入证据为序列“B”。然而，仅就举例而言，需要注意的是在最普通的形式中，该模型不会对观察到的证据的特殊形式作出任何具体表示。例如，输入证据实际上可能是一系列来自于虚拟键盘的触摸坐标。

[0097] 如图2所示，将证据(输入和语境)作为预测引擎的输入使用，在该预测引擎中优选存在着三个模型R、$M_{context}$、M_{input}，各个模型优选包括至少两个子模型(字符模型、一元(unigram)模型；候选模型、语言模型；候选模型、前缀匹配模型)。如图2所示，该预测引擎优选包括目标序列优先模型R。尽管这一特征是优选的，但本系统不限于包括目标序列优先模型R的实施例。

[0098] 目标序列优先模型R包括：

[0099] ●字符模型-在没有固定词汇概念的情况下，在语言中的序列上实施分布。一般被实施为字符序列上的马尔可夫模型(Markov model)。

[0100] 字符模型是一种根据字符(而不是词条)建立的序列模型。例如，如果训练中的集合为“explaining”，一元字符模型可能是下列样子：

[0101] P(e)＝0.1

[0102] P(x)＝0.1

[0103] P(p)＝0.1

[0104] P(l)＝0.1

[0105] P(a)＝0.1

[0106] P(i)=0.2

[0107] P(n)=0.2

[0108] P(g)=0.1。

[0109] 三元字符模型可能是下列样子：

[0110] P(e)=0.1

[0111] P(x|e)=1.0

[0112] P(p|ex)=1.0

[0113] P(l|xp)=1.0

[0114] P(a|pl)=1.0

[0115] P(i|la)=1.0

[0116] P(n|ai)=1.0

[0117] P(i|in)=1.0

[0118] P(n|ni)=1.0

[0119] P(g|in)=1.0。

[0120] ●一元模型-在不考虑语境的情况下，在语言中的序列上实施分布，将每个序列当做基本实体(atomic entity)进行内部处理。

[0121] 举例来说，如果训练中的集合是“the dog chased the cat”，则对应的一元语言模型可能是：

[0122] P(the)->0.4

[0123] P(dog)->0.2

[0124] P(chased)->0.2

[0125] P(cat)->0.2。

[0126] 语境证据模型$M_{context}$包括：

[0127] ●候选模型-在给定特定候选释义的情况下，在语境观察值上实施条件分布。

[0128] ●序列模型-在给定特定语境的情况下，在语言或语言集合中的序列上实施条件分布。在图2中，将序列模型图解为语言模型，该语言模型在一优选实施例中包括一组对应于不同语言的语言模型，例如，LM_{French}，LM_{German}，$LM_{English}$等。

[0129] 输入证据模型M_{input}包括：

[0130] 候选模型-在给定特定候选释义的情况下，在输入观察值上实施条件分布。

[0131] 序列模型-在给定预期目标序列的情况下，在候选上实施条件分布。该模型在图2中被图解为“前缀匹配模型”。

[0132] 包含目标序列优先模型R的各模型可以视情况以用户输入的文本进行更新。通过使用动态语言模型，本系统可以更加精确地预测指定用户的预期的文本序列。

[0133] 各模型输出了一组序列S_R、$S_{context}$、S_{input}以及相关的概率估计P_R、$P_{context}$、P_{input}，用作概率生成器PG的输入。概率生成器PG将上述模型输出的概率估计P_R、$P_{context}$、P_{input}组合在一起，生成针对最终序列预测S_F的一组概率值P_F。

[0134] 最终序列预测S_F可通过用户界面显示给用户，以供用户浏览和选择，或可由本系统使用该最终序列预测S_F以自动更正错误的输入文本。一旦自动地或由用户选择了预测，该输入就优选地被加入语境证据，用以生成下一个预测。相反，如果用户通过输入更多字符

添加与当前词有关的下一输入，则该输入优选地被加入输入证据中以修改分配给预测的当前概率。

[0135] 下面详细介绍本实施例中的特殊系统是如何由数学基础生成的。

[0136] 带入两个证据源的实例化表达式(5)生成：

[0137]

$$\frac{P(s|R)P(context|s,M_{context})P(input|s,M_{input})}{Z} \quad (12)$$

[0138] 其中，Z＝归一化常数，约等于：

[0139] $$\sum_{j=1}^{|s|}P(s_j|R)P(context|s_j,M_{context})P(input|s_j,M_{input}) \quad (13)$$

[0140] 如下所述，该近似值应用于本系统。让我们设想一个序列集合T的函数z，例如：

[0141]

$$z(T)=\sum_{j=1}^{|T|}P(s_j|R)P(context|s_j,M_{context})P(input|s_j,M_{input}) \quad (14)$$

[0142] 求出z如下：

[0143] Z＝z(T)+z({u})*k (15)

[0144] 其中u表示一“未知”序列，而k为|S|-|T|的估计，其中|S|为所有可能的目标序列集合中的序列数量，而|T|是这样的序列的数量：即，对于此类序列，至少一个底层证据模型具有“已知”估计。各单独证据条件模型M会返回估计P(e|u,M)，即：在给定“未知”序列的情况下，证据观察值上的分布。这基本意味着各证据条件模型对其自身的分布平滑负责，但这必须与同“未知”序列的总估计数量成比例的k相关。实际上，各模型都会了解序列集合S’，其中$S'\subset S$，并且对于所有$s\notin S'$，估计P(e|s,M)将保持恒定并等于P(e|u,M)。该特性的平滑是本系统考虑与各证据源相关的模型中的信任级别的可变性所使用的一种手段。

[0145] 根据表达式(12)和(14)，为了确定上述特殊系统实例中的条件概率P(s∈S|e,M)，计算下列估计：目标序列优先P(s|R)；语境概率P(context|s,$M_{context}$)；以及输入概率P(input|s,M_{input})。下面将讨论这些估计以及如何计算这些估计。

[0146] 目标序列优先

[0147] 优选计算目标序列优先如下：

[0148] $$P(s|R)=\begin{cases}P(s|R_{unigram}) & if(s\in V)\\ P(s|R_{character}) & otherwise\end{cases}$$

[0149] 其中V是包含在$R_{unigram}$中的序列集合，而模型的实施则根据构建基于平滑频率的一元语言模型和平滑的马尔可夫链字符模型的公知技术实现。用于实施这些模型的一些应用技术在下文中列出。但，其他适合的技术并未列出。

[0150] ●平滑的n元(n-gram)词条或字符模型(本领域公知)；

[0151] ●如<英国专利申请号0917753.6的专利文献>中所记载的自适应多语言模型；

[0152] ●如<文献：Scheffler 2008>中记载的PPM(prediction by partial matching，部分匹配预测)语言模型；

[0153] ●在一定概率下由构成词法成分生成序列的形态分析引擎。

[0154] 通过包含目标序列优先模型R，本系统改善了预期序列预测的精度。此外，目标序列优先模型R能够实现不可见(unseen)目标序列的基于字符的推理，也就是说，本系统可更好地推断未知目标序列以在所有可能的目标序列上进行近似。

[0155] 语境概率

[0156] 优选地借助第二候选模型估计语境概率$P(context|s,M_{context})$以提出下列表达式(16)。尽管该方式为估计概率的优选手段，但本发明并不限于使用这种方式进行估计的概率。

[0157]

$$P(context|s,M_{context})=\frac{\sum_{j=1}^{K}P(s|c_j,M_{context-sequence})P(c_j|context,M_{context-candidate})P(context|M_{context})}{P(s|M_{context})} \quad (16)$$

[0158] 因此，为了确定语境概率，计算如下各项：语境序列估计$P(s|c_j,M_{context-sequence})$；语境候选估计$P(c_j|context,M_{context-candidate})$；语境优先估计$P(context|M_{context})$；以及目标序列优先估计$P(s|M_{context})$。下面将讨论这些估计以及如何计算这些估计。

[0159] 语境序列估计

[0160] 在给定特定候选序列c_j的情况下，语境序列估计$P(s|c_j,M_{context-sequence})$是语境序列模型下的目标序列s的概率。语境序列模型形式上是一种在给定语境序列的情况下返回目标序列的概率的函数，即$f_s(t_{target},t_{context})=P(t_{target}|t_{context},\theta_s)$，其中，$\theta_s$是模型的参数。因此，语境序列概率的计算为：$P(s|c_i,S)=f_s(s,c_i)$。可使用多种不同技术来计算这个估计。例如，语境训练数据上的平滑的频率分析，与等式(21)相似并如结合目标序列优先估计所表述的。可选地，可单独或结合使用下列任意项：

[0161] ●n元语言模型(本领域公知)；

[0162] ●如<英国专利申请号0917753.6的专利文献>中所记载的自适应多语言模型；

[0163] ●如<文献：Scheffler 2008>中记载的PPM(prediction by partial matching，部分匹配预测)语言模型；

[0164] ●生成HMM(generative Hidden Markov Model，隐马尔可夫模型)概率词性标注器<参考：008.LingPipe 4.1.0.http://alias-i.com/lingpipe(accessed September 26，2011)or Thede,S.M.,Harper,M.P.,1999>；

[0165] ●用于返回部分句子的概率的自然语言解析器，如RASP<参考：Briscoe,E.，J.Carroll and R.Watson 2006>；

[0166] ●被配置为接收表示语境序列和目标序列的特征作为输入并输出概率的神经网络(本领域公知技术)。

[0167] 本系统不限于上述技术，任何可用于计算语境序列概率的技术都适用于本系统。

[0168] 如上文所述，$M_{context-sequence}$可包括对应于多种不同语言的多个语言模型。为了确定表达式(16)的条件概率，可使用与词条相关的语言模型来确定该条件概率。作为一种解释，参考上述实例中的从英语模型($LM_{English}$)和法语模型(LM_{French})中抽出的预测词条“pain”。在这种情况下，表达式(16)被确定为$P(context|pain,LM_{English})$and $P(context|pain,LM_{French})$，其中从法语模型($LM_{French}$)中抽出的“Pain”不同于从英语模型($LM_{English}$)中抽出的“Pain”，尽管该预测在词法上相同。通过将词条与其出处模型相关联，本系统简化了词法相同的词条的处理方式，因为本系统仅保留了两条或两条以上词法相同的词条中的最

可能词条。此外，本系统简化了表达式(16)的条件概率计算。这种简化是可行的，因为尽管词条的词法相同，但词条在不同语言中具有不同的词义，由此可区别对待这种词条。

[0169] 这样转回至图2，由模型$M_{context}$生成的词条集合$S_{context}$可包括$M_{context}$中的任意语言模型(或候选模型)的词条。

[0170] 语境候选估计

[0171] 语境候选估计$P(c_j|context,M_{context-candidate})$是一种函数，其形式为$f_{context-candidate}(t)=P(t|\theta_{context-candidate})$，其中t为任意序列，$\theta_{context-candidate}$为模型参数。这样，语境候选条件估计的计算如下：$P(c_j|context,M_{context-candidate})=f_{context-candidate}(c_j)$。

[0172] 在一优选系统中，语境候选为序列，而语境候选集合表示为有向非循环图(DAG，directed acyclic graph)，有向非循环图中的各节点包括含有一个或多个字符的子序列。每个边被分配有概率，而在一优选实施例中，有向非循环图优选地还具有各路径被限定为同样长度的特殊属性。在本文中，此类DAG变体被称为概率性的受限序列图(PCSG，probabilistic constrained sequence graph)。各单独候选序列由穿过PCSG的唯一路径来表示，而语境候选模型函数对于给定候选的返回值被计算为语境候选模型的代表路径的概率。

[0173] 形式上，PCSG包括含有一组节点N的四元组、根节点r、一组定向边E以及一组参数(概率)θ：

[0174] $G=(N,r,E,\theta)$ (17)

[0175] 两个节点n、n'之间的边表示为(n→n')，沿该边从n移向n'的概率表示为P(n'|n)。穿过G的路径起始于节点r，并循一个从各访问过的节点向外伸出的边延伸，直到抵达不含有外出边(outgoing edge)的节点为止。G的属性如下：

[0176] 1)G为有向非循环图(DAG)；

[0177] 2)$\forall n\in N.\nexists m.(m\rightarrow n)\in E\Rightarrow n=r$，即：除根节点以外的所有节点必须具有至少一个进入边(incoming edge)；

[0178] 3)$\exists m,k\in N.\forall n\in N.(m\rightarrow n)\in E\Rightarrow(n\rightarrow k)\in E$，即：从给定节点分出的所有路径立即重新加入后继的公共路径。这一属性严格地限制了该图的结构，并暗示了所有路径具有相同的长度，减少了路径概率计算的归一化需要。

[0179] 语境候选模型函数计算给定路径的概率如下(等同于语境候选估计)：

[0180] $P(c_j|context,M_{context-candidate})=f_{context-candidate}(c_j)=P(p_j|G)$ (18)

[0181] 其中，$P(p_j|G)$为路径概率，计算为路径中各边的乘积：

[0182] $$P(p_j|G)=P(n_1|r)\prod_{k=2}^{K}P(n_k|n_{k-1}) \quad (19)$$

[0183] 其中，K为路径中的边的数量。值得注意的是，这一优选公式相当于节点间的含蓄的独立性假设。这是因为在这种情况下，候选序列的序列概率未被模型化，而候选中的变化概率被模型化。因此，下列属性用于边上的概率：

[0184] $$\forall n\in N.\sum_{(n\rightarrow m)\in E}P(m|n)=1 \quad (20)$$

[0185] 也就是说，所有出自给定节点n的外出边上的概率之和一定为1。这也意味着下列表达式有效：$\Sigma_i P(p_i|G)=1$，即：PCSG中所有路径的概率之和等于1。

[0186] 某一实例将帮助阐明这些概念。考虑下列12个语境候选序列：

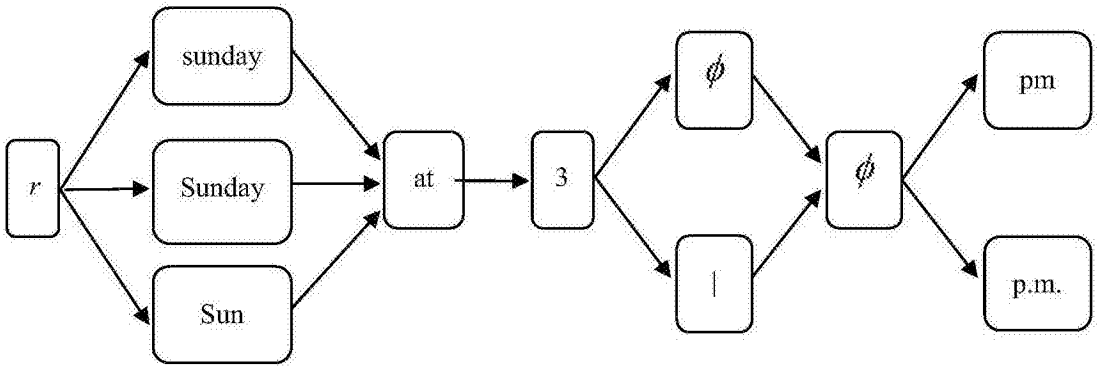
[0187] ●“Sunday at 3pm”●“sunday at 3pm”●“Sun at 3pm”

[0188] ●“Sunday at 3pm”●“sunday at 3pm”●“Sun at 3pm”

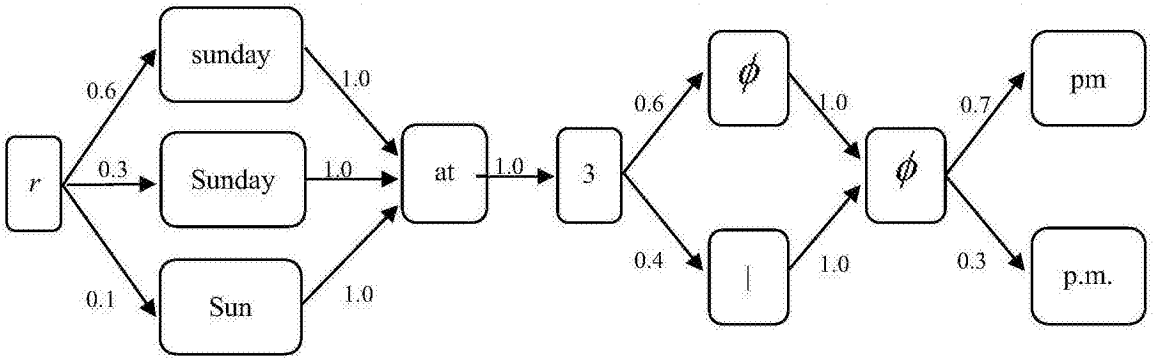
[0189] ●“Sunday at 3p.m.”●“sunday at 3p.m.”●“Sun at 3p.m.”

[0190] ●“Sunday at 3p.m.”●“sunday at 3p.m.”●“Sun at 3p.m.”

[0191] 这些语境候选序列可由下列PCSG(明确的词边界由‘|’表示，而空序列由“ϕ”表示)表示为：

[0192] 

[0193] 根据语境候选模型并采用表达式(19)，将概率分配给边，例如：

[0194] 

[0195] 然后，从PCSG中生成上述12个序列的候选概率如下(为了使表达简洁明了，只列出了三个例子)：

[0196] P("sunday at 3pm"|"sunday at 3pm",C)=0.6*1.0*1.0*0.6*1.0*0.7=0.252

[0197] P("Sunday at 3pm"|"sunday at 3pm",C)=0.3*1.0*1.0*0.4*1.0*0.7=0.084

[0198] P("sun at 3p.m."|"sunday at 3pm",C)=0.1*1.0*1.0*0.4*1.0*0.3=0.012

[0199] 用于构建DAG并向节点分配概率的模型细节将依赖于本系统的特殊实例进行变化。上述图式对三种一般变化的实例进行编码：

[0200] ●词边界上的分支(可能单意义)；

[0201] ●大小写(case)变化上的分支；

[0202] ●词法变化上的分支。

[0203] 不难理解，任意种类的变化均可在这种框架中编码。另一实例将转向之前的方案，例如，如果本系统已预测出“on”和“in”，而用户选择了“in”，则可以将其编码成除了带有分配给“in”的概率权重之外还带有分配给“on”的小概率的分支，以表示用户意外接受错误建议的可能性。在上述情况下，编入以下原则：

[0204] ●以小写字母‘s’开头的‘sunday’的可能性低于缩写形式的“Sun”，而缩写形式的“Sun”的可能性低于完整拼写变形‘Sunday’；

[0205] ●将“pm”与数字“3”拆分的分词情况(tokenization case)的可能性要略微低于不分词的情况；

[0206] ●句点变形“p.m.”的可能性稍微低于无句点形式“pm”。

[0207] 以下列方式，从起始序列s开始优选在算法上构建语境候选PCSG的特殊实例：

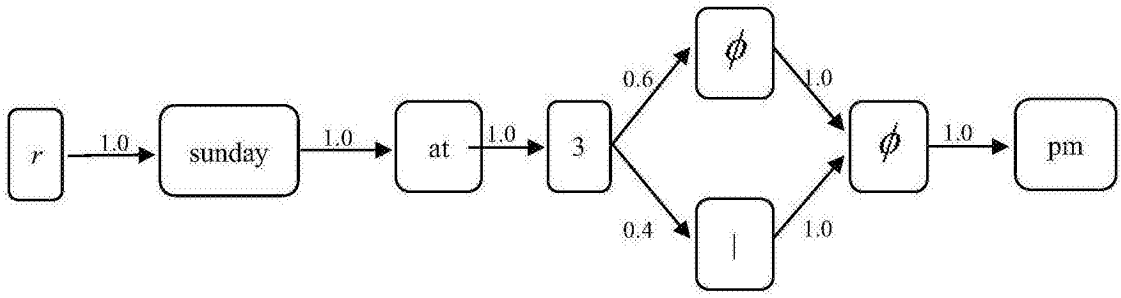
[0208] 1)通过将s封装在连接于根节点的节点n^s中，将s转换成PCSG；

[0209] 2)通过引入变化点上的分支节点，迭代拆析n^s。

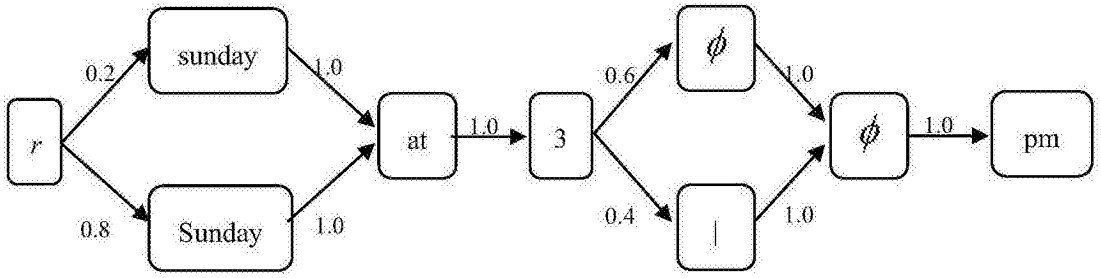
[0210] 举例来说，考虑对原始序列“sunday at 3pm”起作用的PCSG构造算法。首先，步骤1：

[0211] 

[0212] 本系统部署概率分词器，结果如下：

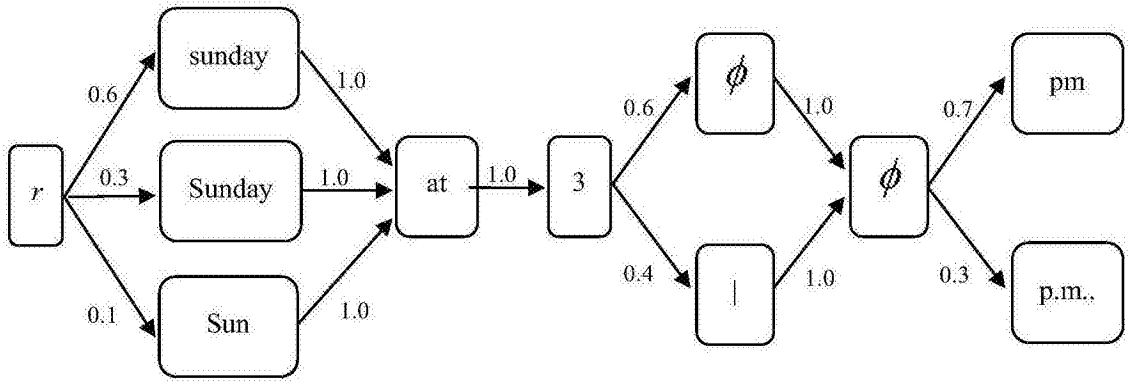
[0213] 

[0214] 值得注意的是，由于上述的PCSG的属性3，修改将总是表现为分支-并-重接(branch-and-rejoin)的结构插入形式，对于特殊情况，修改形式带有一节点的分支，这很便于后续处理，因为其不会影响到全部路径的概率。将在下文中更加详细地介绍根据模型加入边界概率。继续介绍该算法，大小写(case)变形分析器部署如下：

[0215] 

[0216] 最后，词法变形分析器部署如下：

[0217]



[0218] 值得注意的是，因为PCSG的属性3，各分支在重新分岔之前必须集中于一点。这意味着，在某些情况下，如果连续出现两个分支点，则必须插入空节点。

[0219] 边概率优选被分配给PCSG。关于语境候选模型的参数，可优选实施边概率分配。对于这些概率的直观解释有两点：

[0220] 1)它们表示分配给特定分支的用户预期序列的概率估计。例如，如果用户已输入“Dear ben”，我们或许会认为可能他们实际上想要输入“Dear Ben”；

[0221] 2)它们表示对于特定分支为观测到的序列的有效拼写变形的补偿(backoff)概率。例如，如果用户输入“See you on Thur”，则“Thur”的可选择的正确拼写形式可以为“Thurs”。

[0222] 在给定某一背景模型信息的情况下，分配给特定边的概率还会受到正确拼写变体的估计概率的影响。例如，实际上可重新使用语境序列模型s来求得不同正确拼写变体的概率估计。这种概率估计可与其他概率测量结合使用，生成分支概率。以这种方式使用语境序列模型意味着语境候选模型C实际包含语境序列模型S的实例，由此明显违背候选模型和序列模型之间的独立性假设(上文中的属性7)。然而，这种假设绝不会出现在语境情况下，因此相对安全。

[0223] 下列实例将有助于说明。在一优选实例中，假设语境候选模型使用下列算法分配概率：

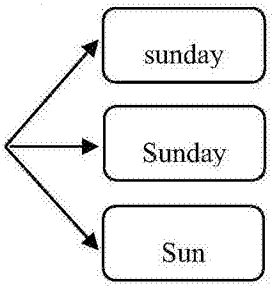
[0224] 1)观测到的序列得到概率0.8，其他序列均匀平分余数；

[0225] 2)用语境序列模型估计来缩放数值；

[0226] 3)归一化数值，使其满足上述PCSG属性(19)。

[0227] 根据上述PCSG实例，下列分支为：

[0228]



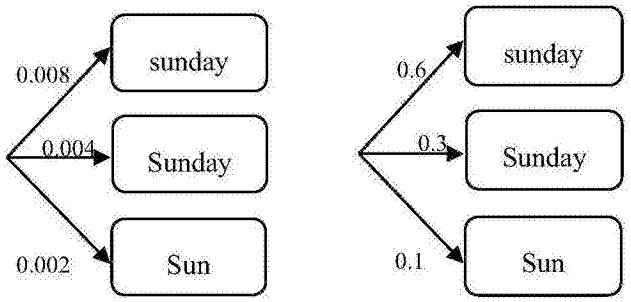
[0229] 因为“sunday”是原始观测值，首先由上述算法的步骤1为其分配概率值0.8，而其他边则各分配有概率值0.1。本实例中由语境序列模型返回的估计如下：

[0230] $P("sunday"|C^S)=0.01$

[0231] $P("Sunday"|C^S)=0.04$

[0232] $P("Sun"|C^S)=0.02$

[0233] 其中，C^S表示在这种情况下语境候选模型使用语境序列模型。因此，在本实例中，分配给各边的未归一化和归一化(经四舍五入)的概率(分别)如下：

[0234] 

[0235] 语境优先估计

[0236] 通过归一化与语境相关的初始序列t的频率，可近似得到语境优先估计$P(context|M_{context})$

[0237] $$P(context|M_{context})\cong\frac{freq(t)}{\sum_{t'}freq(t')} \qquad (21)$$

[0238] 其中freq(t)是训练数据中序列t的频率，分母是训练数据中所有序列的频率之和。表达式(21)中的序列“t”为作为本系统输入的当前语境。语境优先根据如下概率来加权预测的概率值：即抽出该预测的对应模型包括给定语境序列的相应概率。为了实现这一过程，语境优先根据表达式(21)的估计来加权预测值。

[0239] 在实际应用中，例如可通过在不可见序列上安置出现假设(occurrence assumption)或者退回至所有序列均不可见的实例中的受限(低位)估计来平滑该估计。举例来说，如果语境为三元模型，则预测引擎可退回构成二元或一元估计。

[0240] 语境优先提供了对偶函数：有助于归一化概率估计；并在语境模型无法提供有用信息时提供了简单的“模型检测”。如果语境序列估计无法提供信息(例如，当最后的词条对于N元模型而言是未知的)，语境优先估计会对具有最可能语境的模型进行大程度加权，使该模型的预测提升至其他模型的预测之上。“最可能的语境”是在多个模型集合上的估计(21)的最大值，例如语言模型集合$LM_{English}$，LM_{French}，LM_{German}。举例来说，如果语境为“The dog chased”，则同出现在法语中相比可预料这语境更有可能出现在英语中。因此，表达式(21)的条件概率对于$LM_{English}$而言将是最大的，而概率生成器由此会大程度加权来自$LM_{English}$的预测的概率值而非来自LM_{French}的预测的概率值，因此，$LM_{English}$更受语境优先估计的“偏爱”。

[0241] 因此，给定语境，语境优先估计大程度加权来自与多种语言相关的多个语言模型中的最合适的语言模型。这样，语境优先估计可以检测到某人正在输入的文本所属的语言。

[0242] 目标序列优先估计

[0243] 可按照与语境优先估计，表达式(21)同样的方式，使用经平滑的训练数据频率分析估计目标序列优先估计$P(s|M_{context})$，例如：可通过归一化语境训练数据中的所有序列上的目标序列频率，来近似得到目标序列优先：

[0244] $$P(s \mid M_{context}) \cong \frac{freq(s)}{\sum_{s'} freq(s')}$$

[0245] 其中freq(s)为训练数据中目标序列的频率，分母为训练数据中所有目标序列的全部频率之和。分母大致相当于训练数据中的词条(包括重复的词条)总数。

[0246] 输入概率

[0247] 利用第一候选模型估计输入概率P(input|s,M_{input})：

[0248]

$$P(input \mid s, M_{input}) = \sum_{j=1}^{K} P(input \mid c_j, M_{input-candidate}) P(c_j \mid s, M_{input-sequence}) \quad (22)$$

[0249] 因此，为了确定输入概率，需要计算下列估计：输入候选估计P(input|c_j,$M_{input-candidate}$)以及输入序列估计P(c_j|s,$M_{input-sequence}$)。下面介绍这两种估计。

[0250] 输入候选估计

[0251] 输入候选估计P(input|c_j,$M_{input-candidate}$)被定义为观测到的输入事件和序列上的函数：$f_{input-candidate}$(i,t)=P(i|t,$\theta_{input-candidate}$)，其中$\theta_{input-candidate}$是模型的参数。任意输入观测值i在输入序列预期结构(ISIS,input sequence intention structure)中编码。该输入序列预期结构是被映射到概率上的多个序列集合的有序列表：

[0252] {(t_{11}→P(i_1|t_{11}),(t_{12}→P(i_1|t_{12}),...},{(t_{21}→P(i_2|t_{21}),(t_{22}→P(i_2|t_{22}),...},...

[0253] 值得注意的是，各估计具有形式P(i_j|t_{jk})，即：如果用户已打算输入序列t_{jk}，我们应该观测到输入事件i_j的概率是多少。考虑下列ISIS实例：

[0254] $$\begin{bmatrix} \{(H \rightarrow 0.5),(h \rightarrow 0.3),(g \rightarrow 0.1),(j \rightarrow 0.1)\} \\ \{(e \rightarrow 0.8),(w \rightarrow 0.1),(r \rightarrow 0.1)\} \end{bmatrix}$$

[0255] 该ISIS实例为这样一种方案编码，在该方案中本系统估计用户是否打算输入例如其后跟着字符‘e’的字符‘H’，这样预计观测到的输入事件分别具有概率0.5和0.8。

[0256] 生成这些概率分布的方法并不是本发明的主题。更确切地说，突出了一系列适用的技术，举例来说：

[0257] -可根据围绕特定键盘布局上给定目标关键字的字符生成概率分布，例如QWERTY键盘，如果用户敲击与“H”键对应的区域，则在ISIS中可能会包含带有一定概率的字符“G”和“J”。

[0258] -可根据触摸坐标(触摸屏虚拟键盘)和指定按键坐标之间的距离(或某种距离函数，例如平方等)生成概率分布。

[0259] 在优选的系统中，输入候选为序列，而输入候选集合表示为扩展的PCSG(EPCSG)。EPCSG是一种PCSG，但带有一违背标准PCSG属性(在下文中定义)的附加结构。如同语境情况，由穿过EPCSG的唯一路径代表各候选序列，而给定候选的输入候选模型函数返回值被计算为其代表路径的归一化概率。

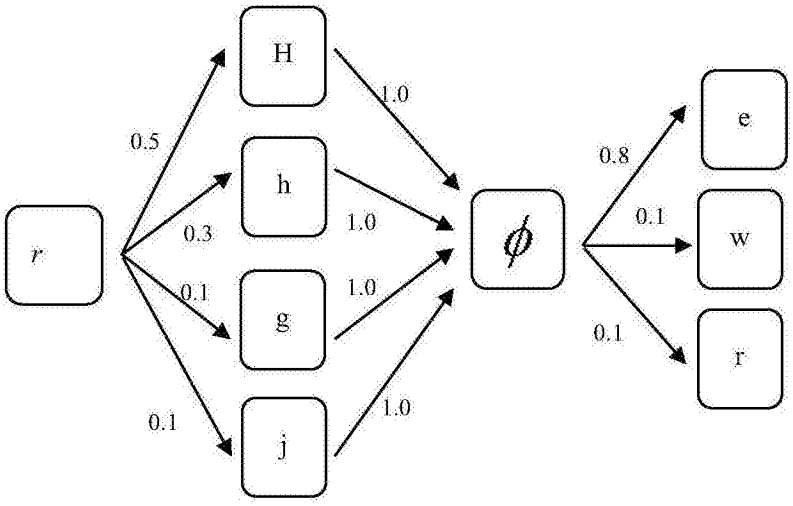
[0260] 输入候选EPCSG生成过程以本系统通过与用户交互生成的序列概率对集合的有序列表开始，其中各子集代表用户输入序列预期上的概率分布。

[0261] 由输入ISIS生成输入候选EPCSG的算法包括两步：

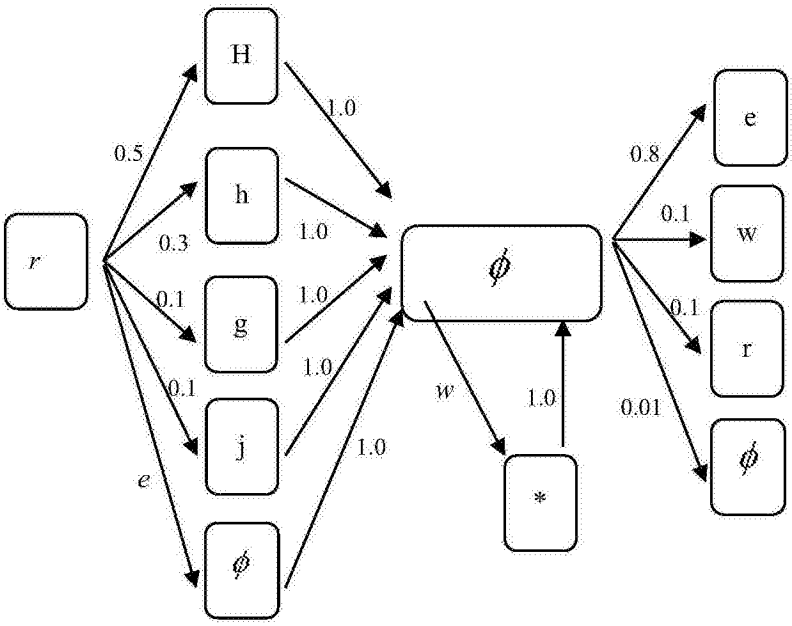
[0262] 1)将ISIS转换成PCSG；

[0263] 2)插入附加的广义结构(generalizing structure)，从而生成EPCSG。

[0264] 步骤1是直截了当的。以新PCSG的根节点开始，该算法为ISIS中的各分布构建分支。步骤1中的上述ISIS结果如下：

[0265] 

[0266] 步骤2使用两个附加结构修整既有的PCSG。第一个结构为空节点子路径(其适合PCSG框架)，第二个结构为‘通配符’结构(将PCSG转换成EPCSG)。步骤2的应用实例如下：

[0267] 

[0268] 通配符的符号(表示为‘*’)实际上是包括/生成字符集合中各符号的分支的简单表达方式。通配符结构是一个受限的循环，因此违背了标准PCSG的非循环属性。EPCSG扩展允许仅在收敛点使用通配符循环，数值e、w是预先规定的概率常量。值得注意的是，在这种情况下，各分支点具有空节点附加(在该情况下为两个)，而各收敛点具有通配符附加(在该情况下为一个)。这些广义结构顾及到用户会遗漏目标序列(带有通配符概率w)中的一个或多个字符或插入一个或多个错误字符(带有孔节点概率e)。不难理解，如何将这些额外的结构加入PCSG的细节可根据计算资源、序列模型长度等随着本系统的不同实例进行变化。

[0269] 空节点子路径使本系统可以丢弃用户错误输入的字符，不然该错误字符会导致不

正确的链经过PCGS。

[0270] 凭借这些附加的广义结构(尤其是通配符分支),可使经过PCSG的路径的数量迅速增加。例如,假设大小为50的字符集,可有1020条不同路径经过上述简化的PCSG。对于实际的ISIS而言,存在数十条甚成百上千条不同的路径。该优选系统优选地以单独或组合的方式使用下列技术处理这种组合激增。

[0271] ●使用特里结构(trie)(单词查找树,本领域公知技术)忽略掉预测词表里那些不是序列前缀的路径;

[0272] ●使用概率阈值删除那些相对不大可能的路径。阈值被设置为当前最可能序列与可能性较低的序列的微分之比。给定阈值t以及当前调查的路径长度L,如果保持下列关系,则删除路径$n_1 \cdots n_L$:

[0273] $$\frac{P(n_1|r)\prod_{j=2}^{L}P(n_j|n_{j-1})}{\arg\max_m[P(m_1|r)\prod_{j=2}^{L}P(m_j|m_{j-1})]}<t \qquad (23)$$

[0274] ●输入序列模型T同样被用于概率阈值。给定不同或受限的阈值t,以及由长度为L的所有路径形成的序列集合:$\{c_1,\cdots,c_K\}$,如果保持下列关系,则删除表示特定序列c_p的给定路径p:

[0275] $$\frac{P(c_p|T)}{\arg\max_j[P(c_j|T)]}<t \qquad (24)$$

[0276] 还可以单独部署或与上述其中一项或全部技术组合部署其他适用于处理组合激增的技术。

[0277] 输入序列估计

[0278] 在给定目标序列的情况下,输入序列估计$P(c_j|s,M_{input-sequence})$是候选序列上的分布,且可被估计为归一化的指标函数:

[0279] $$P(c_j|s,M_{input-sequence})=\frac{\delta(s,c_j)}{z} \qquad (25)$$

[0280] 其中,如果t'为t的前缀,则$\delta(t,t')=1$,否则$\delta(t,t')=0$,而$z=\Sigma_k\delta(s,c_k)$,即所有候选之和。

[0281] 值得注意的是,如果呈现出候选的唯一性,并允许候选集合包括所有可能的序列,则可重新计算归一化因数:Z=length(s)。举例来说,给定目标序列"the",总会正好有三个匹配候选:"t"、"th"和"the"。

[0282] 因此,本发明提供了一种常规的文本预测引擎和系统,以及该文本预测引擎或系统的特定实例,该文本预测引擎或系统能够生成一组分别带有相关概率值P_F的序列预测S_F。

[0283] 本发明还提供了一种用于处理用户文本输入的相应方法。返回至图1和上述系统,该方法包括接收输入至诸如电子设备等用户界面的文本输入;使用文本预测引擎生成序列预测S_F和相关的概率值P_F;并将该序列预测提供至所述用户界面。

[0284] 正如对于本系统的介绍,一般方法包括:通过包含一个或多个模型的文本预测引擎生成序列预测及相关概率值。在一优选实施例中,该方法包括:由目标优先模型R和至少

一个使用至少一个证据源e_1,e_2,…等等生成预测的模型M_1,M_2,…等等,来生成序列预测。如上所述,关于该系统,以及尤其是表达式(12)至(15),该方法包括:通过估计由n个最可能序列预测的概率值与表示剩余可能的序列预测的常量之和求出的概率值归一化因数生成归一化概率值。

[0285] 参照图2,在上述优选实施例中,最终的预测集合S_F以及相关的概率值P_F由概率生成器PG根据分别从目标优先模型R、语境模型$M_{context}$和输入模型M_{input}抽出的预测集合S_R、$S_{context}$、S_{input}生成。在这一实施例中,用户输入序列的语境用作从语境模型$M_{context}$中抽出预测的证据,而与用户正在尝试输入的当前词相关的用户输入序列用作从输入模型M_{input}中抽出预测的证据。

[0286] 该方法的其他方面与上述系统类似,例如,在该方法的某一实施例中,如果序列预测的对应概率值分别大于或等于第一阈值,则仅将这些序列预测提供给用户界面。

[0287] 如以上结合在PCSG中实施广义结构以确定语境候选估计的系统所述的,在该方法的一优选实施例中,该组序列预测中的至少一个对应于由用户输入至用户界面的文本输入的调整或修正版本。

[0288] 根据上述系统的描述进行类推,可很容易地确定本发明方法的其他方面。

[0289] 如下是在本申请中声明的上述实施例的非详尽无遗的权利要求书:

[0290] 1.一种系统,包括:

[0291] 文本预测引擎,其包括:

[0292] 至少一个用于从证据源中生成一组带有相关概率估计的序列的模型;

[0293] 用于生成一组带有相关概率估计的序列的模型;以及

[0294] 概率生成器,其用于接收带有相关概率估计的各组序列并生成带有相关概率值的序列预测。

[0295] 2.实施例1的系统包括多个模型,用于由多个证据源生成多个带有相关概率估计的序列集合。

[0296] 3.在实施例1或2的系统中,概率生成器用于根据任意数量个独立证据源生成一组序列预测。

[0297] 4.在实施例3的系统中,其中一个证据源包括用户输入文本。该用户输入文本可通过用户选择、字符输入、语音识别等方式输入。

[0298] 5.一种系统,包括:

[0299] 用户界面,用于接收用户输入文本;

[0300] 根据第一方面的文本预测引擎或任意其他合适的文本预测引擎,用于接收来自于用户界面的文本输入并生成带有相关概率值的序列预测。

[0301] 6.在实施例5的系统中,该文本预测引擎包括:

[0302] 语境模型,用于接收由用户输入的文本并生成一组序列和相关的概率估计;

[0303] 输入模型,用于接收由用户输入的文本并生成一组序列和相关的概率估计;

[0304] 一模型,用于生成一组序列和相关的概率估计;以及

[0305] 概率生成器,用于从上述多个模型中接收各组序列和相关的概率估计并生成一组序列预测和相关的概率值。

[0306] 7.在实施例6的系统中,所述用户界面用于显示上述序列预测以供用户选择。

[0307] 8.在实施例7的系统中，该系统根据所述概率值为所述序列预测排序并将所述序列预测作为有序集合显示出来。

[0308] 9.在实施例6至8中任意一个实施例的系统中，其中，该系统优选使用所述序列预测自动修正已输入至用户界面的错误输入文本。

[0309] 10.在实施例6至8中任意一个实施例的系统中，所述文本预测引擎用于生成基于任意个独立证据源的序列预测。

[0310] 11.一种处理用户文本输入的方法，包括：

[0311] 接收输入至用户界面的文本；

[0312] 利用预测引擎生成序列预测和相关的概率值；

[0313] 将该序列预测提供给所述用户界面。

[0314] 12.在实施例11的方法中，包括：将所述序列预测显示在所述用户界面以供用户选择。

[0315] 13.在实施例12的方法中，包括：根据所述序列预测的相关概率值，为所述序列预测排序。

[0316] 14.在实施例13的方法中，包括：显示排好序的序列预测以供用户选择。

[0317] 15.在实施例11至14中任意一个实施例的方法中，包括利用所述序列预测自动修正已输入至用户界面的错误输入文本的步骤。

[0318] 16.在前述任意一个实施例中使用的文本预测引擎中，包括：语境模型、输入模型、目标优先模型和概率生成器。

[0319] 17.在实施例16的文本预测引擎中，所述语境模型和所述输入模型用户接收由用户输入的文本并生成一组序列和相关的概率估计。

[0320] 18.在实施例16或17的文本预测引擎中，所述目标优先模型用于生成一组序列和相关的概率估计。

[0321] 19.在实施例16至18中的一个实施例的文本预测引擎中，所述概率生成器用于从所述模型中接收各组序列和相关的概率估计，并生成各对应有概率值的一组序列预测。

[0322] 以上所述仅为本发明的较佳实施例而已，凡在本发明的精神和原则之内，所作的任何修改、等同替换等，均应包含在本发明的保护范围之内。

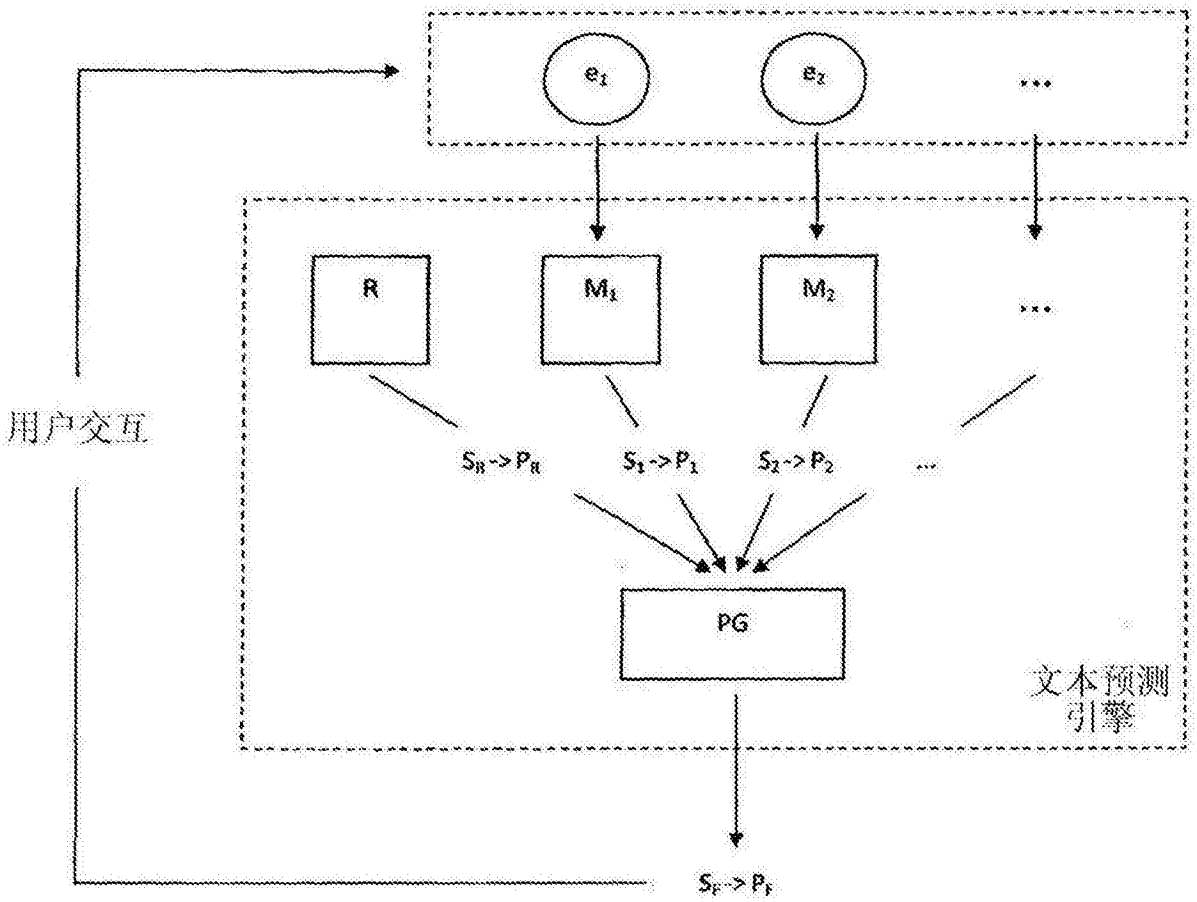


图1

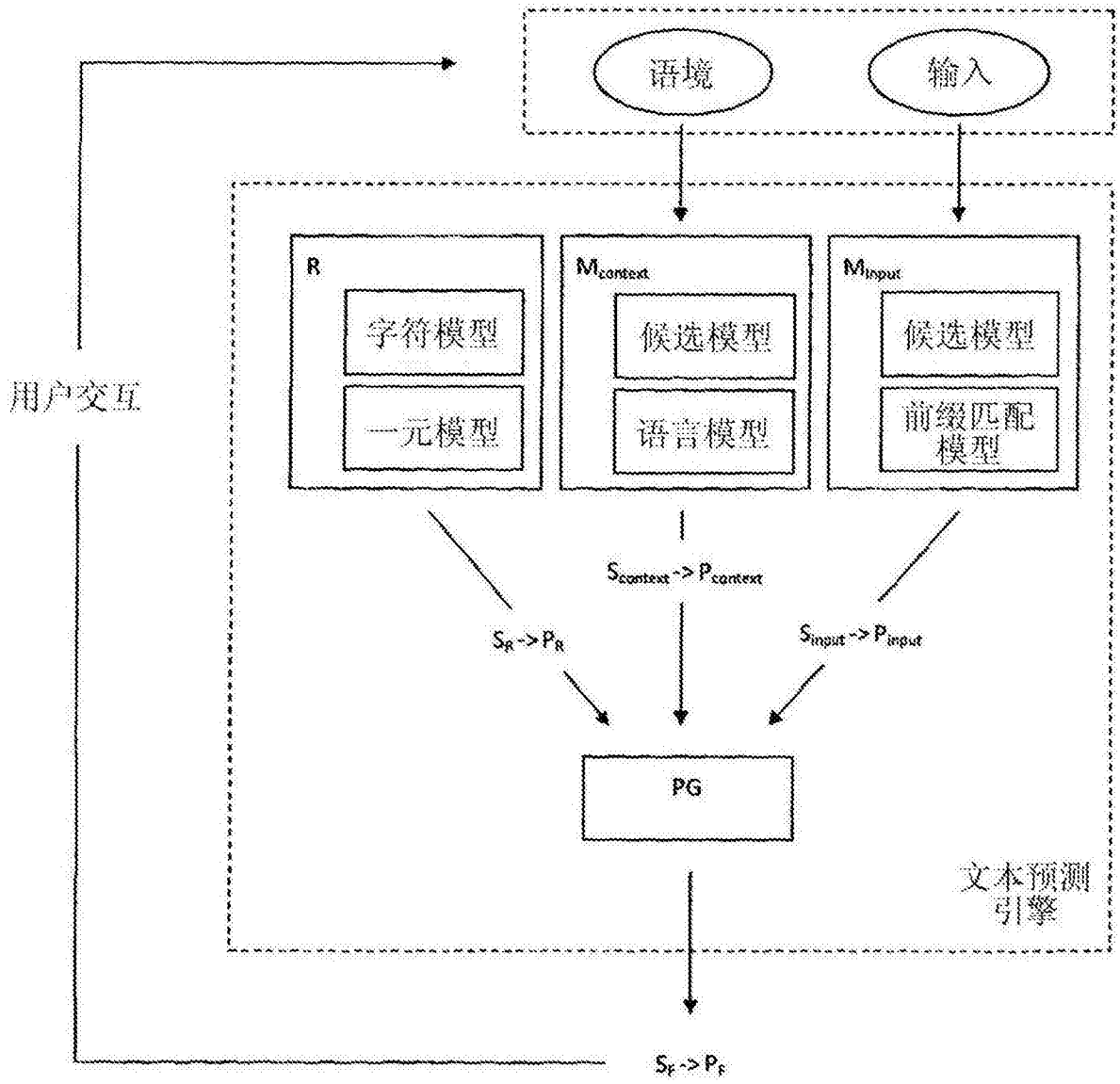

语境
输入
R
字符模型
一元模型
$M_{context}$
候选模型
语言模型
M_{input}
候选模型
前缀匹配模型
$S_{context} -> P_{context}$
$S_R -> P_R$
$S_{input} -> P_{input}$
PG
文本预测引擎
用户交互
$S_F -> P_F$

图2