(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 104081320 B
(45)授权公告日 2017.12.12

(21)申请号 201380006966.X
(22)申请日 2013.01.28
(65)同一申请的已公布的文献号
申请公布号 CN 104081320 A
(43)申请公布日 2014.10.01
(30)优先权数据
1201425.4 2012.01.27 GB
1205598.4 2012.03.29 GB
(85)PCT国际申请进入国家阶段日
2014.07.28
(86)PCT国际申请的申请数据
PCT/GB2013/050182 2013.01.28
(87)PCT国际申请的公布数据
WO2013/110955 EN 2013.08.01
(73)专利权人 触摸式有限公司
地址 英国伦敦
(72)发明人 本杰明·麦德洛克
约瑟夫·哈伊姆·本尼迪克特·奥斯本
(74)专利代理机构 永新专利商标代理有限公司 72002
代理人 王英
(51)Int.Cl.
G06F 3/023(2006.01)
(56)对比文件
CN 101122901 A,2008.02.13,
WO 2011143827 A1,2011.11.24,
CN 1518829 A,2004.08.04,
JP 3403838 B2,2003.05.06,
US 2011197128 A1,2011.08.11,
审查员 钟容

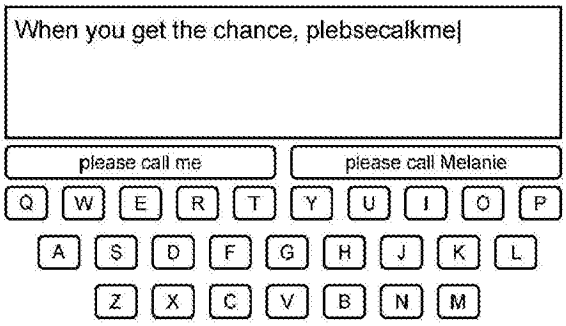
权利要求书4页 说明书22页 附图4页

(54)发明名称

用户数据输入预测

(57)摘要

一种向电子设备中输入文本的系统。所述系统包括:候选生成器(2),其用于由输入序列(20)生成一个或多个候选,所述输入序列(20)包括连续的字符序列,其中所述候选包括被一个或多个词条边界分隔开的两个以上的词条。所述候选生成器(2)将第一概率估量分配给所述候选,具体分配过程如下:在语境语言模型中搜索所述候选的一个或多个词条,其中所述语境语言模型包括词条序列,各所述词条序列包括对应的事件概率;并且将对应于来自所述语境语言模型的所述候选的一个或多个词条的概率分配给所述候选。所述优选生成器(2)根据对应的第一概率估量丢弃掉一个或多个候选。本发明还提供了一种在用户输入序列中推断出词条边界的对应方法。



CN 104081320 B

1.向电子设备中输入文本并且推断输入序列中的词条边界的系统，所述输入序列包括连续的字符序列，所述系统包括：

候选生成器，其用于实施以下操作：

根据所述输入序列生成一个或多个候选，其中，所述输入序列是用户当前正在试图输入且未提交的文本，而语境序列是已经由所述用户提交的文本，并且其中，每个候选包括被一个或多个词条边界分隔开的两个或更多个词条；

将第一概率估量分配给每一个候选，具体分配过程如下：

针对各个候选，在语境语言模型中搜索该候选的一个或多个词条，其中，所述语境语言模型包括词条序列，各所述词条序列包括对应的出现概率；并且

针对各个候选，将与来自所述语境语言模型的该候选的所述一个或多个词条相对应的概率分配给该候选；并且

根据对应的第一概率估量丢弃掉一个或多个候选。

2.根据权利要求1所述的系统，其中，针对各个候选：所述候选生成器在所述语境语言模型中搜索包含与该候选组合的语境序列的序列，其中，所述语境序列为在所述输入序列之前并且包括一个或多个由一个或多个词条边界分隔的词条的用户输入文本。

3.根据权利要求2所述的系统，其中，还包括：

输入序列生成器，其用于将用户输入信号转换成输入序列和语境序列。

4.根据权利要求3所述的系统，其中，所述输入序列生成器通过生成字符集合的序列将用户输入信号转换成输入序列，每个字符集合包括其内字符的概率分布，这样便存在与所述字符集合内的各字符相关的概率值。

5.根据权利要求4所述的系统，其中，所述候选生成器通过将输入序列转换成包括一条或多条路径的概率约束序列图，来根据所述输入序列生成一个或多个候选，其中，所述一条或多条路径对应于所述一个或多个候选。

6.根据权利要求5所述的系统，其中，所述概率约束序列图是有向非循环图的变体，所述有向非循环图包括一组节点和有向边，各所述有向边将一个节点与另一个节点连接，其中，在所述概率约束序列图中，各字符集合的字符被分配了一个节点，各节点的进入边对应于相关字符的概率。

7.根据权利要求6所述的系统，其中，所述概率约束序列图的各条路径被约束为具有相同的长度。

8.根据权利要求5、6或7所述的系统，其中，所述候选生成器将一个或多个词条边界节点插入所述概率约束序列图，所述一个或多个词条边界节点中的每一个具有所述输入序列中的出现概率。

9.根据权利要求8所述的系统，其中，所述候选生成器将词条边界节点插入至任意两个相邻节点之间，所述两个相邻节点对应于所述输入序列的任意两个相邻的字符集合。

10.根据权利要求8所述的系统，其中，所述候选生成器插入词条边界节点，所述词条边界节点作为代表所述输入序列字符集合的任意节点的替换节点。

11.根据权利要求6所述的系统，其中，所述候选生成器向所述概率约束序列图中插入一个或多个通配符节点，所述一个或多个通配符节点中的每一个可使任意字符被插入到除所述输入序列字符集合之外的所述概率约束序列图。

12.根据权利要求5所述的系统，其中，针对各个候选，所述候选生成器通过确定穿过代表该候选的所述概率约束序列图的路径的累积概率，生成该候选的第二概率估量。

13.根据权利要求12所述的系统，其中，所述候选生成器将各候选的第一、第二概率估量组合。

14.根据权利要求13所述的系统，其中，所述候选生成器通过下列方式丢弃掉候选：

确定最可能候选的组合概率；

确定被考虑到的候选的组合概率；

如果最可能候选的组合概率与被考虑到的候选的组合概率之比小于阈值，则丢弃掉候选。

15.根据权利要求12至14中任意一项所述的系统，其中，所述候选生成器确定未被丢弃掉的任意候选的第三概率。

16.根据权利要求15所述的系统，其中，通过以下方式确定所述第三概率：

针对一特定候选，在所述语境语言模型中搜索与该候选组合的包含语境序列以及语境序列的有效正字法变化和词汇变化的序列，以计算相应的语境概率。

17.根据权利要求16所述的系统，其中，候选的总概率为第二概率与第三概率之积。

18.根据权利要求17所述的系统，其中，一个或多个最可能候选由所述候选生成器输出至候选显示器，以将其呈现给用户。

19.根据权利要求1所述的系统，其中，所述词条边界为空格字符。

20.一种推断包含连续字符序列的输入序列中的词条边界的方法，所述方法包括：

使用候选生成器根据输入序列生成一个或多个候选，其中，所述输入序列是用户当前正在试图输入且未提交的文本，而语境序列是已经由所述用户提交的文本，并且其中，每个候选包括被一个或多个词条边界分隔开的两个或更多个词条；

将第一概率估量分配给每个候选，具体分配过程如下：

针对各个候选，在语境语言模型中搜索该候选的一个或多个词条，其中，所述语境语言模型包括词条序列，各所述词条序列包括对应的出现概率；并且

针对各个候选，将与来自所述语境语言模型的该候选的所述一个或多个词条相对应的概率分配给所述候选；并且

根据对应的第一概率估量丢弃掉一个或多个候选。

21.根据权利要求20所述的方法，其中，针对各个候选：在所述语境语言模型中搜索的步骤包括搜索包含与所述候选组合的语境序列的序列，其中，所述语境序列为在所述输入序列之前并且包括一个或多个由一个或多个词条边界分隔的词条的用户输入文本。

22.根据权利要求21所述的方法，其中，还包括：

使用输入序列生成器将用户输入信号转换成输入序列和语境序列。

23.根据权利要求22所述的方法，其中，将用户输入信号转换成输入序列的步骤包括：生成字符集合的序列，每个字符集合包括其内字符的概率分布，这样便存在与所述字符集合内的各字符相关的概率值。

24.根据权利要求23所述的方法，其中，根据所述输入序列生成一个或多个候选的步骤包括：将所述输入序列转换成包括一条或多条路径的概率约束序列图，其中，所述一条或多条路径对应于所述一个或多个候选。

25.根据权利要求24所述的方法，其中，所述概率约束序列图是有向非循环图的变体，所述有向非循环图包括一组节点和有向边，各所述有向边将一个节点与另一个节点连接，其中，在所述概率约束序列图中，各字符集合的字符被分配了一个节点，各节点的进入边对应于相关字符的概率。

26.根据权利要求25所述的方法，其中，所述概率约束序列图的各条路径被约束为具有相同的长度。

27.根据权利要求24、25或26所述的方法，其中，所述方法还包括：使用所述候选生成器将一个或多个词条边界节点插入所述概率约束序列图，所述一个或多个词条边界节点中的每一个具有所述输入序列中的出现概率。

28.根据权利要求27所述的方法，其中，将词条边界节点插入至任意两个相邻节点之间，所述两个相邻节点对应于所述输入序列的任意两个相邻的字符集合。

29.根据权利要求27所述的方法，其中，插入词条边界节点，所述词条边界节点作为代表所述输入序列字符集合的任意节点的替换节点。

30.根据权利要求20所述的方法，还包括：使用所述候选生成器向所述概率约束序列图中插入一个或多个通配符节点，所述一个或多个通配符节点中的每一个可使任意字符被插入到除所述输入序列字符集合之外的所述概率约束序列图。

31.根据权利要求24所述的方法，还包括：针对各个候选，使用所述候选生成器通过确定穿过代表该候选的所述概率约束序列图的路径的累积概率，生成该候选的第二概率估量。

32.根据权利要求31所述的方法，还包括：使用所述候选生成器确定各候选的第一、第二概率估量之积。

33.根据权利要求32所述的方法，其中，丢弃掉候选的步骤包括：

使用所述候选生成器确定最可能候选的组合概率；

使用所述候选生成器确定被考虑到的候选的组合概率；

如果最可能候选的组合概率与被考虑到的候选的组合概率之比小于阈值，则使用所述候选生成器丢弃掉候选。

34.根据权利要求31至33中任意一项所述的方法，还包括：使用所述候选生成器确定未被丢弃掉的任意候选的第三概率。

35.根据权利要求34所述的方法，其中，通过以下方式确定所述第三概率：

使用所述候选生成器在所述语境语言模型中搜索与候选组合的包含语境序列以及语境序列的有效正字法变化和词汇变化的序列，以计算相应的语境概率。

36.根据权利要求35所述的方法，其中，候选的总概率由第二概率与第三概率之积所决定。

37.根据权利要求36所述的方法，其中，还包括：将一个或多个最可能候选从所述候选生成器输出至候选显示器，以将其呈现给用户。

38.根据权利要求20所述的方法，其中，所述词条边界为空格字符。

39.一种推断包含连续字符序列的输入序列中的词条边界的装置，其包括：

用于执行权利要求20至38中任意一项所述方法的单元。

40.一种用户界面，其包括：

文本窗，其用于显示用户当前输入的文本；

预测窗，其用于显示候选，其中，所述候选是由用户意图表示的当前输入文本的预测，并且所述候选是通过以下获得的：根据输入序列产生候选，并基于每个候选的语境概率估量丢弃一个或多个候选，其中，所述输入序列是所述用户当前正在试图输入且未提交的文本，而语境序列是已经由所述用户提交的文本；

虚拟键盘，其用于接受用户输入的文本，其中，所述候选包括被一个或多个词条边界分割开的两个或更多个词条，并且所述虚拟键盘不包括代表所述词条边界的按键。

41.根据权利要求40所述的用户界面，其中，所述预测窗包括两个预测窗，第一个预测窗用于显示用户输入序列，而第二个预测窗用于显示所述用户输入序列的预测。

用户数据输入预测

技术领域

[0001] 本发明涉及一种根据用户输入文本预测多个词条的系统和方法，尤其涉及一种预测用户输入文本中一个或多个词条边界的方法和系统。

背景技术

[0002] 目前存在多种探测用户输入文本中词条边界的系统，例如，Google™的Android™ICS键盘，或Apple™的iOS键盘。但是，这些系统在词条边界的探测方法上具有一些局限性。

[0003] 以Google™的Android™ICS键盘为例，如果用户输入两个有效的属于基本词汇的单词，且未输入分隔空格或除空格之外的其他字符，该系统则会提供两个由空格分隔的单词，替代相关差错字符。但是，该系统中存在很多限制，尤其是：

[0004] ●如果任一单词为键入错误或拼写错误，则该系统无法工作；

[0005] ●该系统无法与其他输入操作函数一起工作，例如自动省略号插入函数；以及

[0006] ●该系统一次只能分析两个单词，即：其仅能识别一个单词边界。

[0007] Apple™的iOS键盘比Android™ICS键盘更加先进的地方在于：Apple™的iOS键盘可以补偿有限的键入错误或拼写错误。然而，Apple™的iOS键盘也只限于识别一个单词的边界。

[0008] 本发明的目的在于克服上述限制，以便可在无需明确插入词条边界的情况下允许用户输入完整的短语甚至整条消息。

发明内容

[0009] 在本发明的第一方面中，提供了一个向电子设备中输入文本的系统，其包括：

[0010] 候选生成器，其用于实施以下操作：

[0011] 由输入序列生成一个或多个候选，所述输入序列包括连续的字符序列，其中所述候选包括被一个或多个词条边界分隔开的两个以上的词条；

[0012] 将第一概率估量分配给所述候选，具体分配过程如下：

[0013] 在语境语言模型中搜索所述候选的一个或多个词条，其中所述语境语言模型包括词条序列，各所述词条序列包括对应的事件概率；并且

[0014] 将对应于来自所述语境语言模型的所述候选的一个或多个词条的概率分配给所述候选；并且

[0015] 根据对应的第一概率估量丢弃掉一个或多个候选。

[0016] 在一优选实施例中，所述一个或多个词条包括候选。优选地，针对各个候选，所述候选生成器在所述语境语言模型中搜索包含与所述候选组合的语境序列的序列，其中所述语境序列为用户输入文本，所述用户输入文本在所述输入序列之前并且包括一个或多个由一个或多个词条边界分隔的词条。

[0017] 所述系统还包括：输入序列生成器，其用于将用户输入信号转换成输入序列和语

境序列。所述输入序列生成器优选通过生成字符集合的序列将用户输入信号转换成输入序列，所述字符集合包括其内字符的概率分布，这样便存在与所述字符集合内的各字符相关的概率值。

[0018] 所述候选生成器优选通过将输入序列转换成包括一条或多条路径的概率约束序列图由输入序列生成一个或多个候选，其中所述一条或多条路径对应于所述一个或多个候选。所述概率约束序列图优选是为有向非循环图的变体，所述有向非循环图包括一组节点和有向边，各所述有向边将一个节点与另一个节点连接，其中，在所述概率约束序列图中，各字符集合的字符被分配了一个节点，各节点的进入边对应于相关字符的概率。所述概率约束序列图的各条路径被约束为具有相同的长度。

[0019] 所述候选生成器优选将一个或多个词条边界节点插入所述概率约束序列图，所述一个或多个词条边界节点具有所述输入序列中的事件概率t。优选地，所述候选生成器将词条边界节点插入至任意两个相邻节点之间，所述节点对应于所述输入序列的任意两个相邻的字符集合。此外，所述候选生成器插入词条边界节点，所述词条边界节点作为代表所述输入序列字符集合的任意节点的替换节点。

[0020] 所述候选生成器优选向所述概率约束序列图中插入一个或多个通配符节点，所述一个或多个通配符节点可使字符被插入到除所述输入序列字符集合之外的所述概率约束序列图。

[0021] 所述候选生成器优选通过确定穿过代表所述候选的所述概率约束序列图的路径的累积概率，生成所述候选的第二概率估量。所述候选生成器优选将各候选的第一、第二概率估量组合。

[0022] 优选地，所述候选生成器通过下列方式丢弃掉候选：

[0023] 确定最可能候选的组合概率；

[0024] 确定被考虑到的候选的组合概率；

[0025] 如果最可能候选的组合概率与被考虑到的候选的组合概率之比小于阈值t，则丢弃掉候选。

[0026] 所述候选生成器确定未被丢弃掉的任意候选的第三概率。通过以下方式确定所述第三概率：在所述语境语言模型中搜索与候选组合的包含语境序列以及语境序列的有效正字法变化和词汇变化的序列，其中所述语境序列为在所述输入序列之前的包含一个或多个由一个或多个词条边界分隔的词条的用户输入文本。候选的总概率为第二概率与第三概率之积。

[0027] 一个或多个最可能候选由所述候选生成器输出至候选显示器，以将其呈现给用户。

[0028] 在一实施例中，所述词条边界为空格字符。

[0029] 在本发明的第二方面中，提供了一种推断包含连续字符序列的输入序列中的词条边界的方法，所述方法包括：

[0030] 使用候选生成器由输入序列生成一个或多个候选，其中所述候选包括被一个或多个词条边界分隔开的两个以上的词条；

[0031] 将第一概率估量分配给所述候选，具体分配过程如下：

[0032] 在语境语言模型中搜索所述候选的一个或多个词条，其中所述语境语言模型包括

词条序列，各所述词条序列包括对应的事件概率；并且

[0033] 将对应于来自所述语境语言模型的所述候选的一个或多个词条的概率分配给所述候选；并且

[0034] 根据对应的第一概率估量丢弃掉一个或多个候选。

[0035] 在一实施例中，所述一个或多个词条包括候选。针对各个候选，在所述语境语言模型中搜索包含与所述候选组合的语境序列的序列，其中所述语境序列为用户输入文本，所述用户输入文本在所述输入序列之前并且包括一个或多个由一个或多个词条边界分隔的词条。

[0036] 上述方法优选包括：使用输入序列生成器将用户输入信号转换成输入序列和语境序列。将用户输入信号转换成输入序列的步骤优选包括：生成字符集合的序列，所述字符集合的序列包括其内字符的概率分布，这样便存在与所述字符集合内的各字符相关的概率值。

[0037] 由所述输入序列生成一个或多个候选的步骤包括：将所述输入序列转换成包括一条或多条路径的概率约束序列图，其中所述一条或多条路径对应于所述一个或多个候选。所述概率约束序列图优选为有向非循环图的变体，所述有向非循环图包括一组节点和有向边，各所述有向边将一个节点与另一个节点连接，其中，在所述概率约束序列图中，各字符集合的字符被分配了一个节点，各节点的进入边对应于相关字符的概率。所述概率约束序列图的各条路径优选被约束为具有相同的长度。

[0038] 上述方法优选还包括：使用所述候选生成器将一个或多个词条边界节点插入所述概率约束序列图，所述一个或多个词条边界节点具有所述输入序列中的事件概率t。优选将词条边界节点插入至任意两个相邻节点之间，所述节点对应于所述输入序列的任意两个相邻的字符集合。

[0039] 插入词条边界节点；所述词条边界节点作为代表所述输入序列字符集合的任意节点的替换节点。

[0040] 上述方法优选还包括：使用所述候选生成器向所述概率约束序列图中插入一个或多个通配符节点，所述一个或多个通配符节点可使任意字符被插入到除所述输入序列字符集合之外的所述概率约束序列图。

[0041] 上述方法优选还包括：使用所述候选生成器通过确定穿过代表所述候选的所述概率约束序列图的路径累积概率，生成所述候选的第二概率估量。使用所述候选生成器确定各候选的第一、第二概率估量之积。

[0042] 优选地，丢弃掉候选的步骤包括：

[0043] 使用所述候选生成器确定最可能候选的组合概率；

[0044] 使用所述候选生成器确定被考虑到的候选的组合概率；

[0045] 如果最可能候选的组合概率与被考虑到的候选的组合概率之比小于阈值t，则使用所述候选生成器丢弃掉候选。

[0046] 优选地，上述方法还包括：使用所述候选生成器确定未被丢弃掉的任意候选的第三概率。通过以下方式确定所述第三概率：使用所述候选生成器在所述语境语言模型中搜索与候选组合的包含语境序列以及语境序列的有效正字法变化和词汇变化的序列，其中所述语境序列为在所述输入序列之前的包含一个或多个由一个或多个词条边界分隔的词条

的用户输入文本。候选的总概率由第二概率与第三概率之积所决定。

[0047] 上述方法包括：将一个或多个最可能候选从所述候选生成器输出至候选显示器，以将其呈现给用户。

[0048] 在一优选实施例中，所述词条边界为空格字符。

[0049] 在本发明的第三方面中，提供了一种计算机程序产品，其包括：计算机可读介质，其上存储有能够使处理器执行上述方法的计算机程序。

[0050] 在本发明的第四方面中，提供了一种用户界面，其包括：

[0051] 文本窗，其用于显示用户当前输入的文本；

[0052] 预测窗，其用于显示候选，其中，所述候选是由用户意图表示的当前输入文本的预测；

[0053] 虚拟键盘，其用于接受用户输入的文本，其中所述虚拟键盘不包括代表空格键的按键。

[0054] 优选地，所述预测窗口包括两个预测子窗口，第一个预测子窗口用于显示用户输入序列，而第二个预测子窗口用于显示所述用户输入序列的预测。

附图说明

[0055] 参照下列附图，详细介绍本发明：

[0056] 图1为本发明的高级预测结构的原理图；

[0057] 图2a为本发明实例界面的示意图；

[0058] 图2b为本发明实例界面的示意图；

[0059] 图3为本发明系统的应用实例的示意图；

[0060] 图4为本发明方法的流程图。

具体实施方式

[0061] 本发明提供了一种诸如移动电话或平板电脑等电子设备的文本输入系统和方法，通过使用概率性语境模型推断文本输入中词条边界的可能位置。

[0062] 本发明系统包括一种根据基本词汇生成多词条边界推断结果的方法，所述方法根据基于语境语言数据过滤这些结果的概率方法。

[0063] 本发明系统优选与概率性字符处理结合，例如申请号为PCT/GB2011/001419的国际专利申请(其内容作为参考被全面引入本文)中披露的概率性字符处理，以生成一可以同时纠正键入错误或拼写错误的系统，而且本发明系统还具有其他通过概率推断得到的字符流的操作(例如，向英语中自动插入省略号)能力。因此本发明系统可以生成(并向用户或系统提供)高概率多词条预测。

[0064] 由此，本发明系统和方法既提供了通过去除手动分隔单词而提高用户输入率的手段，又向试图手动分隔单词但未能如愿(用户插入替代分隔符的伪字符或用户未能插入任何字符或诸如分隔符)的用户提供了纠正反馈。

[0065] 被赋予了一些用户文本输入实例的下列实施例演示了本发明系统和方法输出(一个或多个词条的文本预测)。该实施例演示了本发明系统/方法兼容诸如没有词条边界的用户输入文本、以伪字符代替边界的文本输入等多种文本输入脚本的能力。

[0066] 没有词条边界的用户输入文本实例如下：

[0067] 对于用户输入的“seeyoulater”，本发明系统/方法会预测/输出“see you later”。

[0068] 以伪字符代替边界的用户输入文本实例如下：

[0069] 对于用户输入的“seecyoublater”，本发明系统/方法会预测/输出“see you later”。

[0070] 本发明的系统和方法还可以实时解决前缀匹配的问题，例如：

[0071] 对于用户输入的“seeyoula”，本发明系统/方法会预测/输出“see you later”。

[0072] 本发明的系统/方法可以补偿键入错误/拼写错误，与此同时推断出单词边界和前缀，例如：

[0073] 对于用户输入的文本“seeyoulayer”，本发明系统/方法会预测/输出“see you later”。

[0074] 对于用户输入的文本“whatstheti”，本发明系统/方法会预测/输出“what's the time”。

[0075] 本发明的系统/方法可以生成呈现给用户的经过概率排名的预测集合。举例来说，根据用户输入“thedrmin”，本发明的系统/方法可输出下列排名预测：

[0076] “these mints”

[0077] “the seminar”

[0078] <等>

[0079] 一般而言，但并非绝对，本发明系统的概况如图1所示。

[0080] 上述系统优选扩张现有的概率性文本输入系统，诸如申请号为PCT/GB2011/001419的国际专利申请中披露的系统，该专利的内容作为参考被全面引入本文。但该系统也可被单独使用，例如仅加强诸如本申请背景技术中描述的现有的文本输入系统。

[0081] 如图1所示，本发明系统包括候选生成器2。优选地，本发明系统还包括输入序列生成器1和候选显示器3。候选显示器3用于将一个或多个候选40显示给用户。

[0082] 输入序列生成器1用于根据用户输入生成文本序列。考虑到不同的文本输入界面类型，用户输入可由对应于一系列不同电子设备交互类型的一系列用户信号10生成。举例来说：

[0083] ●本领域公知的QWERTY(或其他布局的)键盘；

[0084] ●本领域公知的概率性虚拟QWERTY(或其他布局的)键盘；

[0085] ●比较复杂的键盘按键模型(例如，申请号为1108200.5的英国专利申请，“用户输入预测”，其内容作为参考被全面引入本文)；

[0086] ●概率性连续击打键盘界面(例如，申请号为1108200.5的英国专利申请，“输入文本的系统和方法”，其内容作为参考被全面引入本文)。

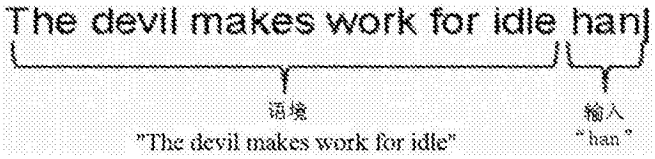
[0087] 输入序列生成器1接受由用户生成的输入信号10并返回结构化的给定语言中的字符的序列。输入序列生成器1输出两个结构化的字符序列：输入序列20和语境序列30。

[0088] 语境序列30包括固定的字符序列。输入序列20以本文提到的概率性字符串的形式出现。概率性字符串包括字符集合的序列，各字符集合具有其内字符的概率分布，从而使各字符集合中的字符均具有相关的概率值。下文中示出一实例，在该实例中，概率性字符串包

括遍及三个字符集合的序列，各序列(由{-}表示)包括两个以上的带有相关概率的字符，各相关概率之和为1：

[0089] {{{a,0.8},{s,0.1},{z,0.1}},{{r,0.9},{t,0.05},{f,0.05}},{{e,0.8},{w,0.2}}}

[0090] 举例来说，用户输入完序列“The devil makes work for idle han”时，光标立即出现在“han”之后。对于这一实例而言，语境序列30为截止于“idle”且包含“idle”的字符，而输入序列20为遍布于由字母‘h’、‘a’和‘n’想要的字符的分布：

[0091] 

[0092] 一般而言，语境序列30可被认为是已由用户提交的文本，而输入序列20可被认为是用户当前试图输入的文本。输入序列20为包含字符的连续序列。包含字符的连续序列是没有明确词条边界的字符连续序列。候选生成器2用于以包含两个以上词条的预测替换掉整个输入序列20。该预测的词条间具有一个或多个推断出的词条边界。上述预测包括空格对错误字符的替换、词条的纠正和/或标点符号的插入。

[0093] 举例来说，如果用户已向系统提交一段话“The devil makes”，并正在输入“workforidlehan”，则语境序列30为“The devil makes”，而输入序列20对应于“workforidlehan”。

[0094] 由此，输入序列生成器1生成对应于用户(例如，候选预测40的在先用户选择50)提交给系统的文本的语境序列30，以及对应于用户当前正在输入但还未提交给系统的文本的、以包含字符的连续序列的形式出现的输入序列20。

[0095] 语境及输入序列30,20由输入序列生成器1输出，并由候选生成器2当作输入接受。

[0096] 候选生成器2生成一个或多个候选预测40。候选预测40对应于多词条序列。在该序列中，根据需要推断词条边界。为了实现这一目的，候选生成器2根据用户计划/意图输入序列的可能性对序列进行排序。

[0097] 这等同于为受下列表达式支配的集合中的序列进行排序：

[0098] (1)P(s∈S|e,M)

[0099] 其中，e为观察到的证据，而M为一组受过训练的将被用于概率推理的模型。换句话说，在证据已知的前提下，本发明系统会评估抽出预测的所有序列集合的条件概率，以及评估序列的条件概率。使用贝氏定理(Bayes Theorem)重新整理表达式(1)后得到下列表达式：

[0100] $$\frac{P(e|s,M)P(s|M)}{P(e|M)} \quad (2)$$

[0101] 边缘化分母中的目标序列后得到下列表达式：

[0102] $$\frac{P(e|s,M)\,P(s|M)}{\sum_{j=1}^{|S|}P(e|s_j,M)\,P(s_j|M)} \quad (3)$$

[0103] 为了计算P(e|s,M)，在目标序列已知的前提下，假设证据被分成多个互斥集合[e_1...e_N]。互斥集合由某一相关模型下的分布独立生成。由此作出如下假设：

[0104] 假设1：在目标序列已知的前提下，证据可被分成多个不同集合，从而使各集合中的证据有条件地独立于所有其他证据。

[0105] 各e_i具有与其相关的模型M_i。

[0106] 这一独立假设的数学表达式如下：

[0107] $$P(e|s,M)=\prod_{i=1}^{N}[P(e_i|s,M_i\in M)] \quad (4)$$

[0108] 该表达式允许构建一框架。该框架允许组合一些具有适当模型的证据源。

[0109] 优选地，模型R∈M与先验目标序列相关。考虑到这一模型，上述假设，即表达式(3)可被重新整理如下：

[0110] $$\frac{P(s|R)\prod_{i=1}^{N}P(e_i|s,M_i)}{\sum_{j=1}^{|S|}P(s_j|R)\prod_{i=1}^{N}P(e_i|s_j,M_i)} \quad (5)$$

[0111] 涉及到s的表达式(5)的分母保持不变，由此不会对排序产生影响，而仅充当结果概率值的归一化因子。在运行上述系统时，这一概率值被估算为近似值，具体如下文中提供了归一化概率值的表达式(11)所述。然而，由于该分母不会对排序产生影响，文本预测排序可以充分实施，因此本发明系统无需估算分母。

[0112] 考虑到上述表达式(5)，先验目标序列P(s|R)必须与各证据源e_i的证据概率P(e_i|s,M_i)和相关模型M_i一起运算。

[0113] 本发明系统提供了两种计算证据概率的方法。该方法在概率框架中通过边缘化证据的候选解释：候选模型1和候选模型2计算证据概率。

[0114] 候选模型1

[0115] 在由单一证据源估算证据概率时，有益于根据‘候选’表示(表达)模型。这一过程是介于‘用户预期/意图’序列和观察到的证据之间的中间阶段。证据概率可被重新表达如下：

[0116] $$P(e|s,M)=\sum_{j=1}^{K}P(e|c_j,s,M_{candidate})P(c_j|s,M_{sequence}) \quad (6)$$

其中，c_j为单一候选，现在我们有两个已知证据源的M子模型：候选模型$M_{candidate}$和序列模型$M_{sequence}$。关键假设如下：

[0117] 假设2：在候选已知的前提下，所述模型下的概率被表示为候选的边缘概率，其中，证据有条件地独立于目标序列。

[0118] 应用这一假设，可从证据词条中除去对于s的依赖：

[0119]

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

[0120] 候选模型2

[0121] 候选模型的另一变体首先利用贝氏定理转换证据概率：

[0122]

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

[0123] 证据的条件序列概率可被重新表示如下：

[0124]

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

[0125] 其中，c_j为单一候选，现在我们有两个已知证据源的M子模型：候选模型$M_{candidate}$和序列模型$M_{sequence}$。在这种情况下，关键假设如下：

[0126] 假设3：在候选已知的前提下，所述模型下的概率被表示为候选的边缘概率，其中，目标序列有条件地独立于证据。

[0127] 应用这一假设，可从证据词条中除去对于s的依赖：

[0128]

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

[0129] 候选生成器2使用来自两个证据源的证据，即分别对应于由输入序列生成器1输出的语境序列30和输入序列20的语境证据源和输入证据源。

[0130] 如上所述，语境代表那些用户已经输入的被观察到的证据。如果语境序列包含多个单词，语境则以带有词条边界的字符序列的形式出现。输入代表那些用户当前正在输入的被观察到的证据，并以概率字符串的形式出现。

[0131] 带有这两种证据源的实例化表达式(5)生成：

[0132]

$$\frac{P(s|R)P(context|s,\ M_{context})P(input|s,\ M_{input})}{Z} \quad (11)$$

[0133] 在该表达式中，不会影响排序的可选Z代表归一化因子。作为参考被全文引入本文中的申请号为PCT/GB2011/001419的国际专利申请详细解释了归一化因子的估算。特别地，该归一化常量Z约等于：

[0134]

$$\sum_{j=1}^{|S|}P(s_j|R)P(context|s_j,M_{context})P(input|s_j,M_{input})$$

[0135] 这一近似值在上述系统中的应用如下。函数z是序列T集合的函数：

[0136]

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

[0137] Z的计算如下：

[0138] Z＝z(T)+z({u})*k

[0139] 其中，u代表“未知”序列，而k为|S|-|T|的估量，其中|S|为所有可能目标序列集合中的序列数量，而|T|为的某一序列的数量，潜在证据模型中的至少一个具有所述某一序列的“已知”估量。每一证据条件模型M会返回P(e|u,M)的估量，即：对一个“未知”序列的证据观察值结果的分布。实质上，这意味着每个证据条件模型对其自身的分布平滑性负责，但必须与k相关。其中，k与“未知”序列的总估计数量成正比。事实上，各模型“了解”序列S'集合，其中$S' \subset S$，而P(e|s,M)的估量为恒量，且与P(e|u,M)相等，尽管$s \in S'$。这一属性的平滑性使本系统考虑与各证据源相关的模型中的变化置信度。

[0140] 在上述框架中，本发明系统对下列三个模型实施操作：

[0141] 先验目标序列模型R包括：

[0142] ●一元模型-实现一个序列在某种语言中的分布，无需考虑语境，在内部中将各序列作为基本单元处理，举例来说，一元模型为n等于1的n元模型。

[0143] 输入证据模型M_{input}包括下列两种模型：

[0144] ●候选模型：$M_{input-candidate}$-实现输入观察值的条件分布，假定已知一特定候选解释；

[0145] ●序列模型：$M_{input-sequence}$-实现候选的条件分布，假定已知意图目标序列。

[0146] 语境证据模型$M_{context}$包括下列两种模型：

[0147] ●候选模型：$M_{context-candidate}$-实现语境观察值的条件分布，假定已知一特定候选解释；

[0148] ●序列模型：$M_{context-sequence}$-实现在某一语言或语言集合中的序列的条件分布，假定已知特定语境。

[0149] 候选生成器2计算表达式(11)分子中的各个估量，以确定具有最高级别概率的候选40。候选40中的一个或多个作为文本预测被显示在候选显示器3上。

[0150] 下面，将介绍各相关个体概率在表达式(11)中的计算过程。由此也将介绍下文各词条的计算过程：输入概率P(input|s,M_{input})，语境概率P(context|s,$M_{context}$)以及先验目标序列估量P(s|R)。Z的计算是可选的，因此不再赘述。

[0151] 先验目标序列

[0152] 先验目标序列的优选计算如下：

[0153] $$P(s|R) = \begin{cases} P(s|R_{unigram}) \text{如果 } (s \in V) \\ P(s|R_{character}) \text{否则} \end{cases}$$

[0154] 其中，V是包含在$R_{unigram}$中的序列集合，而模型的实施依据构建平滑的以频率为基础的一元语言模型已知技术以及平滑的马尔可夫链模型。下面列出了可实施这些模型的一些适用技术。但是，其他未列出的适当技术也可适用。

[0155] ●平滑的n元词条或字符模型(本领域公知技术)；

[0156] ●如<参考:申请号为0917753.6的英国专利申请>中披露的自适应多语言模型;

[0157] ●如<参考:Scheffler 2008>中披露的PPM(部分匹配预测)语言模型;

[0158] ●形态分析引擎,其用于根据句子的词汇组分依据一定概率地生成序列。

[0159] 通过囊括先验目标序列模型R,本发明系统改善了预期序列预测的准确性。此外,先验目标序列模型R使不可预见的目标序列的基于字符的推断成为可能,即:本发明系统能够更好地推断未知目标序列,以逼近所有可能的目标序列。

[0160] 输入概率P(input|s,M_{input})

[0161] 输入概率P(input|s,M_{input})由候选模型1,特别是表达式(7)实施估算。

[0162]

$$P(input|s,M_{input})=\sum_{j=1}^{k}P(input|c_j,M_{input-candidate})P(c_j|s,M_{input-sequence}) \quad (12)$$

[0163] 其中,表达式(7)的证据源e为输入,即输入序列20,而相关的模型为包含$M_{input-candidate}$和$M_{input-sequence}$的输入模型M_{input}。

[0164] 因此,为了确定输入概率P(input|s,M_{input}),需要先确定输入候选估量P(input|c_j,$M_{input-candidate}$)和输入序列估量$P(c_j,s_i|M_{input})$。下文将详细介绍这两种估量。

[0165] 输入候选估量P(input|c_j,$M_{input-candidate}$)

[0166] 作为参考被全篇引入本文中的申请号为PCT/GB2011/001419的英国专利披露了概率约束序列图(probabilistic constrained sequence graph,PCSG)-有向非循环图(directed acyclic graph,DAG)的变体。其中,各节点包含由一个或多个字符构成的子序列。由一个或多个字符构成的子序列对应于来自字符集合的单个字符,或对应于来自于某一序列的单个字符。所述某一序列穿过两个以上包含输入序列20的字符集合。根据包含输入序列20的字符集合中字符的概率分布,为同向节点的各个边分配概率。PCSG还具有这样一种特性:各路径被限制为同一长度。

[0167] 从形式上看,PCSG由4元组构成。该4元组包含节点集合N、根节点r、定向边E和(概率)参数集合θ:

[0168] G=(N,r,E,θ) (17)

[0169] 两个节点n和n'之间的边表示为(n→n'),而沿着边从n移动至n'的概率表示为P(n'|n)。经过G的路径起始于节点r,并恰好随着来自各访问过的节点的一条外出边延伸,直至到达未包含外出边的节点。G包含如下特性:

[0170] 1)G为有向非循环图;

[0171] 2)$\forall n\in N.\ \exists m.(m\rightarrow n)\in E\Rightarrow n=r$,即:除根节点之外的所有节点必须包含至少一条进入边;

[0172] 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在后继普通节点上立即重新组合。这一特性严格限制图结构并暗指所有路径具有同样的长度,降低了路径概率计算中的归一化要求。

[0173] 输入候选模型函数计算给定路径概率(等同于输入候选估量)

[0174] 如下：

[0175] $P(input|cj,Minput\text{-}candidate)=f_{input\text{-}candidate}(c_j)=P(p_j|G)$ 其中，$P(p_j|G)$ 为路径概率，作为路径中各条边的乘积被求出：

[0176] $$P(p_j|G)=P(n_1|r)\prod_{k=2}^{K}P(n_k|n_{k-1})$$

[0177] 其中，K为路径中边的数量。值得注意的是，这一优选公式实际上是节点间的隐性独立假设。下文将在PCSG实例的说明之后，介绍这一独立假设。

[0178] 由此，下列特性维持着边上的概率：

[0179] $$\forall n\in N.\ \sum_{(n\to m)\in\varepsilon}P(m|n)=1$$

[0180] 换句话说，来自给定节点n的所有外出边上的概率之和必须为1。这同样暗示出如下维持手段 $\Sigma_i P(p_i|G)=1$，也就是说，PCSG中所有路径的概率之和等于1。

[0181] 在本发明系统中，输入候选集合被表示为扩展的PCSG(EPCSG)。单一输入候选被表示为穿过EPCSG的唯一路径。对于各输入候选而言，其代表路径的归一化概率可被计算出。

[0182] 首先，根据输入序列20(可在图上表示为输入候选)生成PCSG。之后，以附加结构扩大PCSG。该附加结构允许使用更具有表现力但仍然受到约束的候选。

[0183] 在下文中提供了某一场景下使用的PCSG实例，其中的输入序列如下：

[0184] $$\begin{bmatrix}\{(H\to 0.5),(h\to 0.3),(g\to 0.1),(j\to 0.1)\}\\ \{(e\to 0.8),(w\to 0.1),(r\to 0.1)\}\end{bmatrix}$$

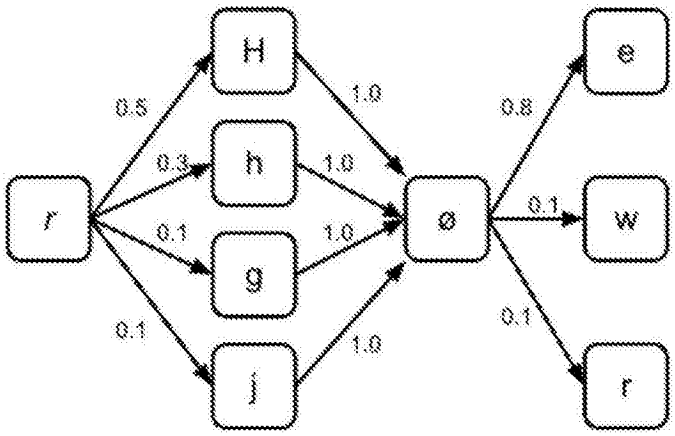
[0185] 该输入编制了一个场景。在该场景中，输入序列生成器估算的用户预期输入，例如，其后跟随着字符‘e’的字符‘H’，这样，观察到的输入事件预计包含分别为0.5和0.8的概率。然而，通过生成包含两个字符集合的概率字符串，输入序列生成器还考虑到用户预期输入除“H”和“e”之外的其它字符。这两个字符集合具有遍及每个给定集合中的字符的概率分布。

[0186] 生成这些概率分布的方法并不是本发明的主题，而是为了强调可以使用的一些技术，例如：

[0187] -可根据特定键盘布局上给定目标按键周围的字符生成分布，例如QWERTY键盘，如果用户敲击对应于“H”键的区域，则字符“G”和“J”有一定概率被包含在输入序列中。

[0188] -可根据(触摸屏虚拟键盘上的)触摸坐标与指定按键坐标之间的距离(或某种距离函数，例如square函数等)生成分布。

[0189] 未被扩大的PCSG由候选生成器2根据输入序列20生成，其中，输入序列中各字符集合内的字符各被分配了一个其在PCSG内的节点，看上去好似如下：

[0190] 

[0191] 各字符集合被作为一组同级节点添加至该图中。同级节点在无效节点之后、下一组同级节点之前立即重组。各同级节点的进入边概率对应于字符集合中字符的概率值。

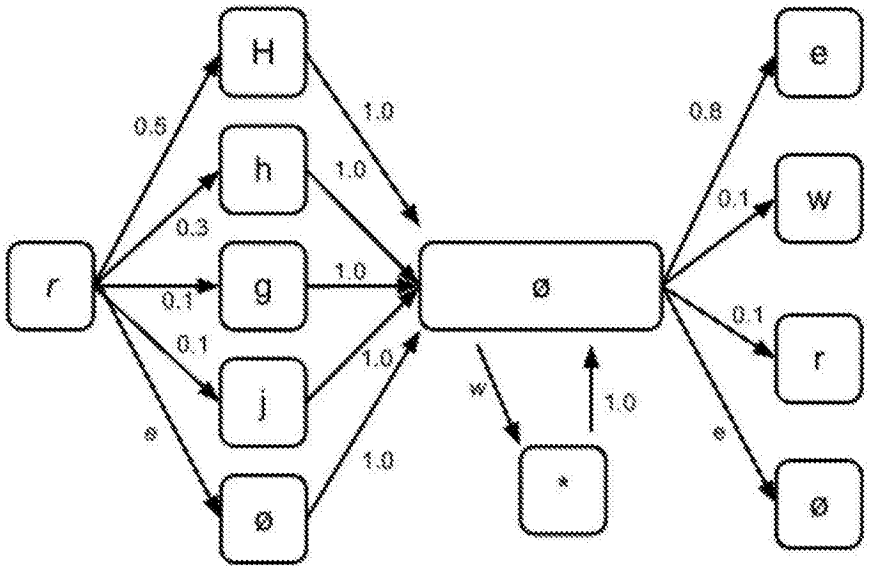
[0192] 如上所述，路径概率$P(p_j|G)$的优选构想实际上是节点间的含蓄独立假设。这种情况得到了具有遍及于其内字符的概率分布的各字符集合的证实。这种概率分布独立于其他字符集合字符的概率分布。从字符集合中生成的同级节点集合具有一概率分布。这一概率分布从其它同级节点分离而来，且独立于其他同级节点。从PCSG的构造可以看出，在一给定路径中只使用了一个来自于同级节点集合的节点。

[0193] 如作为参考被全篇引入本文中的申请号为PCT/GB2011/001419的英国专利所述，可使用两个附加结构来扩大PCSG，进而建立EPCSG。该EPCSG包括：

[0194] ●空节点子路径，其顾及了包含错误字符的输入序列；

[0195] ● ‘通配符’结构，其顾及了用户遗漏一个或多个目标序列字符的实例。

[0196] 用上述方法扩大的同一PCSG(以提供EPCSG)如下：

[0197] 

[0198] 凭借预设的概率惩罚e，本发明系统允许插入用于替代对应于输入序列的字符的空字符，以处理来自用户的错误输入。为了实现这一过程，候选生成器2在上述各同级节点集合中插入了作为额外节点的空字符节点(表示为“Ø”)。这样，便存在了依循包含一个或多个空字符的路径跳过一个或多个字符的穿过EPCSG的路径。

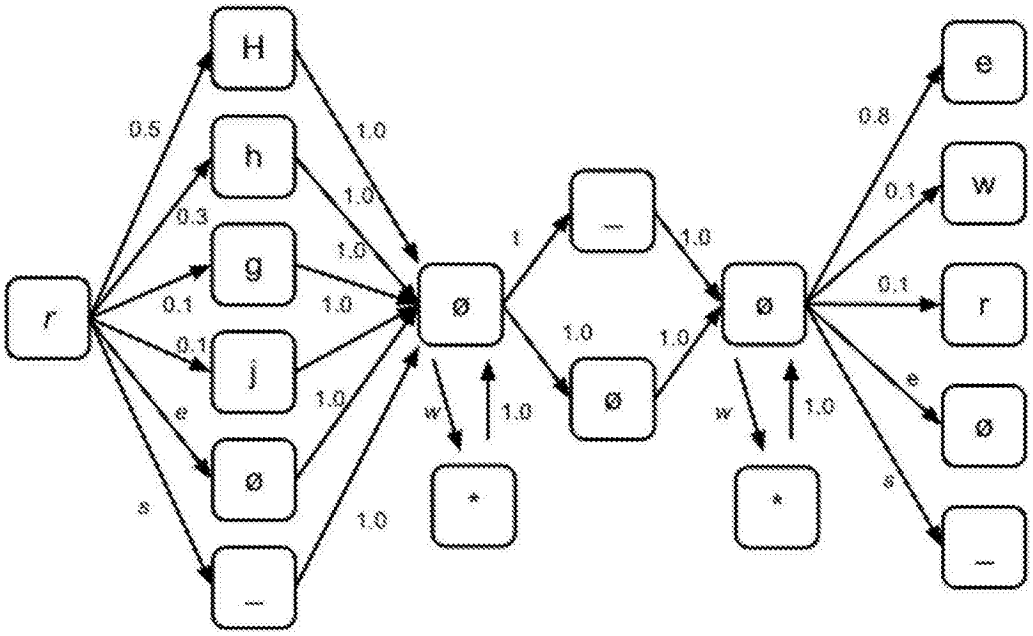
[0199] 同样，凭借预设的惩罚w，本发明系统顾及了在输入中被完全遗漏的目标序列中的

字符。以‘*’标注的节点为包含字符集合中每一可能字符的分支的简写(例如，‘通配符’)。根据上述的EPCSG，候选生成器2在各同级字符节点之间插入通配符结构(例如，代表通配符的节点)。该机构允许在并非单独根据输入预测出的候选中考虑被遗漏的字符。

[0200] 本发明系统的候选生成器2还将更多以词条边界形式出现的结构插入至EPCSG中。词条边界不需被当成诸如空格的字符。在某些语言中，单词之间并不存在空格。但本发明系统预测一个单词的结尾位置和另一个单词的开头位置，即：该系统识别出词条边界。为此，候选生成器2在上述的各同级节点中插入作为附加节点的词条边界(以‘_’作为标记)节点，以便对用户输入字符而非诸如空格等词条边界所产生的影响作出纠正。此外，候选生成器在各同级字符节点集合之间插入了另一词条边界节点。因为在通配符结构的任意使用之间必须考虑词条边界，这样单个通配符结构必须被分成两个相同结构，并需要在其内插入词条边界结构。词条边界结构仅由词条边界节点和空字符节点构成，这使得词条边界能够作为可选项使用。

[0201] 使用词条边界结构以及通配符结构和空字符结构扩大的EPCSG实例如下：

[0202]



[0203] 这些词条边界节点增加了两种主要含意：

[0204] ●预设的概率惩罚s，本发明系统可使用词条边界替换掉错误字符。这一过程的典型实例为：用户本打算插入间断符(break)，但却错误地插入了伪输入。

[0205] ●预设的概率惩罚t，词条间断符可被插入于字符之间，无论该间断符是否由EPCSG生成，或者是否由(存在于词条边界节点的任一侧)通配符分支生成。

[0206] 对于边特性$\forall n \in N.\ \sum_{(n\to m)\in E} P(m|n) = 1$而言，为了保持边概率之和不变，需要相应地对各边概率按比例进行调整，以使边概率之和仍为1。这一过程是可以实现的，因为插入结构e、s、t和w的概率是预先设置的。

[0207] EPCSG中的这些附加结构极大地增加了穿过图形的有效路径的数量。为了抵消这一问题，本发明提供了候选生成器2。该候选生成器2使用了新颖的概率修剪技术。该修剪技术既顾及了ESPCSG路径概率，又顾及了边界推断词条序列的语境概率。

[0208] 候选生成器2利用语境语言模型$M_{context}$生成各个多词条候选的语境概率估量。使

用语境语言模型$M_{context}$生成输入候选语境概率估量的过程明显违反了输入证据(输入序列20)和语境证据(语境序列30)可被分成非重叠性集合的假设。该非重叠性集合各自由被赋予目标序列的关联模型($M_{context}$,M_{input})下的某一分布生成。

[0209] 候选生成器2使用各个多词条候选的语境概率估量修剪穿过EPCSG的路径。下文中将详细介绍这一过程。对于路径的修剪极大地减少了搜索空间,由此使得对于(没有边界的)连续文本中的词条边界的计算识别成为可能。

[0210] 给定语境语言模型$M_{context}$,例如为$P(t_i)$、$P(t_i|t_{i-1})$和$P(t_i|t_{i-1},t_{i-2})$提供概率估算的三连词模型(尺寸n为3的n元模型),并给定待估算的k-长度序列,则语境概率估量如下:

[0211] (13)$P(t_k,t_{k-1},t_{k-2},\ldots,t_3,t_2,t_1)$

[0212] 该估量通过如下计算得出:

[0213] (14)

[0214] $P(t_k|t_{k-1},t_{k-2})*P(t_{k-1}|t_{k-2},t_{k-3})*\ldots*P(t_4|t_3,t_2)*P(t_3|t_2,t_1)*P(t_2|t_1)*P(t_1)$

[0215] 举例来说,如果语境序列30为"have you ever",而输入序列20的候选(即穿过EPCSG的一条路径)为"been to London",则候选生成器2会生成三个三连词:"you ever been"、"ever been to"和"been to London"。三连词"have you ever"完全由语境序列30构成。作为被提交的输入的任意此类三连词具有概率值1,由此不再需要考虑这一三连词,因为其无法改变结果。之后,通过将上述各三连词的概率值相乘,形成该实例的语境概率估量。

[0216] 上述过程反映出表达式(14)中的本实例k-长度序列的长度为5,该序列包括{you,ever,been,to,London}。

[0217] 在不存在语境序列但存在同一输入序列的情况下,候选生成器2针对一个三连词"been to London"的语境概率估量修剪候选。于是出现了一个较长的输入,例如"been to London to see"。候选可被针对三个三连词"been to London"、"to London to"和"London to see"进行修剪。

[0218] 任意适当的语境语言模型可被用于为候选提供概率估量。本发明系统并不限于使用三连词模型。该模型的使用仅用于举例说明。

[0219] 因此,本发明系统提供了一种修剪方法,该修剪方法基于输入序列完整内容以及包含语境序列30的语境序列的词条。

[0220] 这样,训练过的语境数据的使用可生成短语。这些短语未预先被模型识别出,并有可能被模型有意或无意地丢弃。

[0221] 为了修剪EPCSG路径,候选生成器通过确定给定候选的语境概率估量与给定候选的累积PCSG路径概率的乘积,将给定候选的语境概率估量与给定候选的累积PCSG路径概率组合。候选生成器2通过阈值t设置当前最可能序列与最不可能序列之比的下限,即:最可能路径与需要被修剪的路径的概率比。因此,如果保持下列关系的话,路径n1…nL则被修剪。

[0222] (15)

$$\frac{P(t_k^n,t_{k-1}^n,t_{k-2}^n,\ldots t_3^n,t_2^n,t_1^n)\,P(n_1|r)\,\prod_{j=2}^{L}P(n_j|n_{j-1})}{argmax_m\left[P(t_k^m,t_{k-1}^m,t_{k-2}^m,t_{k-2}^m,\ldots t_3^m,t_2^m,t_1^m)\,P(m_1|r)\,\prod_{j=2}^{L}P(m_j|m_{j-1})\right]}<t$$

[0223] 然而，除了上述与表达式14和15相关的修剪技术之外，其它修剪技术也可被使用。术语‘概率估量’是指真正数学意义上的概率或估算概率的其他形式，例如排名或计分。通过在表达式14中加权词条，可将分配给候选的概率估量或‘分数’朝向候选中早先词条偏置，以尽可能早地忽视掉大片的EPCSG(large swathes of EPCSG)，这样因为该方法无需横穿整个EPCSG的所有路径，由此可提高计算效率。此外，需要检索的词条序列可小于构成候选的全部词条序列，例如，需要检索的词条序列包括候选的正数第一个词条或正数的前两个词条。不管怎样，分配给候选的分数/概率估量仍通过检索语境模型$M_{context}$中的候选词条序列来确定。此外，替代表达式15的其他阈值界设置方法可被使用。

[0224] 在一优选实施例中，候选生成器2计算表达式14的概率估量，因为其之后可以通过估算用于确定语境概率特别是用于确定语境序列估量的同一个三连词，修剪输入候选。因为候选生成器2可缓存并再利用未被修剪的候选的三连词结果(而对于被修剪的候选，可丢弃三连词结果)，这样可降低所需的计算量。对于这一过程，将在下文中与表达式(17)一起介绍。

[0225] 修剪路径还可以降低候选生成器2(通过等式11)实施的预测计算量，因为在表达式12和16中只有很少的候选需要估算。

[0226] 有可能出现在EPCSG中的路径可由候选生成器2基于语境概率估量进行修剪。然而，如果此处并未完成这一修剪，那么相同的候选会着手从等式11的语境模型中接收非常低的分数(出于同样的原因，这些候选会被修剪)，这样会赋予这些候选相当低的概率，反而导致这些候选在后续阶段中被丢弃。

[0227] 输入序列估量$P(c_j|s,M_{input-sequence})$

[0228] 假定目标序列为已知，输入序列估量$P(c_j|s,M_{input-sequence})$是候选序列的分布，并可被估算为归一化指标函数：

[0229] (16) $$P(c_j|s,M_{input-sequence})=\frac{\delta(s,c_j)}{Z}$$

[0230] 其中，如果t’是t的前缀，则$\delta(t,t')=1$，否则等于0，而$Z=\Sigma_k\delta(s,c_k)$，即：所有候选之和。

[0231] 如果假定了候选的唯一性，并且候选集合被允许包括全部有效的序列，那么归一化因子Z＝length(s)可被重塑，因为对于长度为n的目标序列而言，恰好存在n个匹配的前缀，那么也即存在n个候选。

[0232] 语境概率$P(context|s,M_{context})$

[0233] 语境概率$P(context|s,M_{context})$由候选模型2估算，特别是由表达式(8)和(10)估算。在表达式(8)和(10)中，证据源e是语境：

[0234] (17) $$P(context|s,M_{context})=\frac{\sum_{j=1}^{n}P(s|c_j,M_{context-sequence})P(c_j|context,M_{context-candidate})P(context|M_{context})}{P(s|M_{context})}$$

[0235] 表达式17中对j求和涉及到对语境中的词汇及正字法变化求和。如果输入候选在修剪阶段已被丢弃，词条序列的这种变化则不再需要被估算。

[0236] 如表达式(17)所示，为了计算语境概率，候选生成器2计算如下估量，这些估量将

在下文中被详细介绍：语境序列估量P(s|c_j,$M_{context-sequence}$)、语境候选估量P(c_j|context,$M_{context-candidate}$)、先验语境估量P(context|$M_{context}$)以及先验目标序列估量P(s|$M_{context}$)。

[0237] 语境序列估量P(s|c_j,$M_{context-sequence}$)

[0238] 语境序列估量P(s|c_j,$M_{context-sequence}$)是在语境序列模型下目标序列s的概率，假定候选序列c_j是已知的。形式上，语境序列模型是返回被赋予语境序列的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$\theta_s$为模型参数。因此，语境序列概率可被计算为：$P(s|c_i,S)=f_S(S,c_i)$。

[0239] 目前存在一些可单独或组合使用的技术，例如：

[0240] ●n元语言模型(本领域公知技术)；

[0241] ●如申请号为PCT/GB2010/001898的专利申请中披露的自适应多语言模型；

[0242] ●本领域公知的PPM(部分匹配预测)语言模型；

[0243] ●本领域公知的衍生HMM(Hidden Markov Model，隐马尔可夫模型)概率词性标注器；

[0244] ●被适当配置的神经网络。

[0245] 语境候选估量P(c_j|context,$M_{context-candidate}$)

[0246] 语境候选估量P(c_j|context,$M_{context-candidate}$)是具有下列形式的函数：$f_{context-candidate}(t,context)=P(t|context,\theta_{context-candidate})$，其中t任意序列(arbitrary sequence)，$\theta_{context-candidate}$是模型参数。由此，语境候选条件估量可被计算为：

[0247] $P(c_j|context,M_{context-candidate})=f_{context-candidate}(c_j,context)$

[0248] 在考虑到语境序列中的这种变化时，以与使用输入候选估算技术相似的方式，将正在发生语境词条正字法变化(例如，首字母大写形式、缩写形式、首字母缩写形式)的图形分支与利用了EPCSGs的技术一起使用。

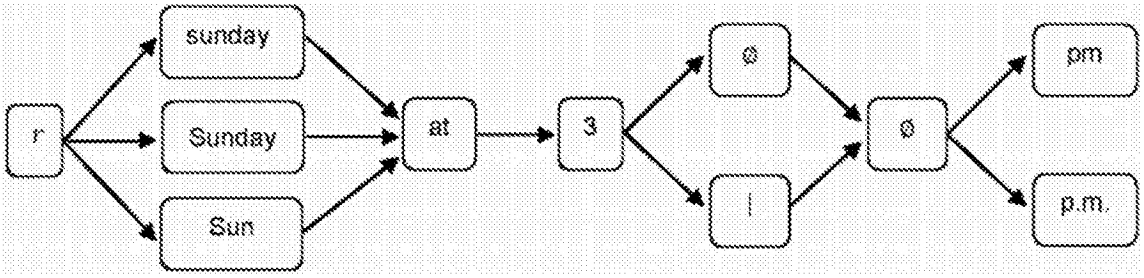
[0249] 下面，提供了下列12个语境候选序列的实例。

[0250]

“Sunday at 3pm”	“sunday at 3pm”	“Sun at 3pm”
“Sunday at 3 pm”	“sunday at 3 pm”	“Sun at 3 pm”
“Sunday at 3p.m.”	“sunday at 3p.m.”	“Sun at 3p.m.”
“Sunday at 3 p.m.”	“sunday at 3 p.m.”	“Sun at 3 p.m.”

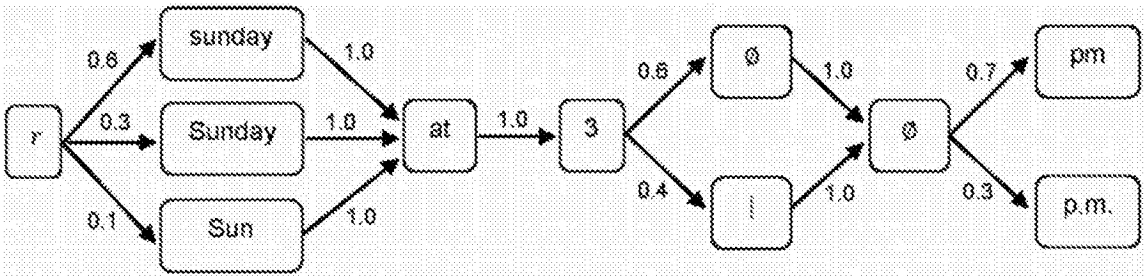
[0251] 这些语境候选序列可由下列PCSG(由‘|’表示明确的单词边界，由空字符‘Ø’表示空序列)表示：

[0252]



[0253] 概率被分配给取决于语境候选模型的边，(遵循(19))例如：

[0254]



[0255] 随后从PCSG中生成上述12个序列的候选概率如下(出于简洁考虑,只列出了三个实例):

[0256] P("sunday at 3pm"|"sunday at 3pm",C)=0.6*1.0*1.0*0.6*1.0*0.7=0.252

[0257] P("Sunday at 3pm"|"sunday at 3pm",C)=0.3*1.0*1.0*0.4*1.0*0.7=0.084

[0258] P("Sun at 3p.m."|"sunday at 3pm",C)=0.1*1.0*1.0*0.4*1.0*0.3=0.012

[0259] 用于构建PCSG并向节点分配概率的模型特性会根据本发明系统特定实例的不同而变化。上述图式对三个普通变化的实例进行了编码:

[0260] 单词边界上的分支(潜在、无歧义的);

[0261] 字形变化上的分支;

[0262] 词汇变化上的分支。

[0263] 不难理解,任意类型的变化可在上述框架内被编码。另一实例为依据在先建议分支。例如,如果系统已给出预测"on"和"in",而用户选择了"in",这样可对这一变化进行编码,使其成为分支,该分支带有分配给"in"的概率权重,以及分配给"on"的小概率,以代表用户意外接受错误建议的概率。在上述情况中,对下列原则进行了编码:

[0264] 首字母's'为小写的'sunday'的概率小于缩写形式'Sun',而缩写形式'Sun'的概率小于完整变体'Sunday';

[0265] "pm"与数字"3"分开的断词形式(tokenisation case)的概率略微小于"pm"与数字"3"未分开的断词形式;

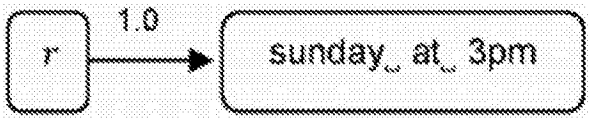
[0266] 句号变体"p.m."的概率稍微小于不存在句号的形式"pm"。

[0267] 按照下列方式,优选在算法上由首字母序列s构成语境候选PCSG的特定实例。

[0268] 通过将首字母序列s封装入连接于根节点的单一节点n^s,使首字母序列s转换为PCSG;

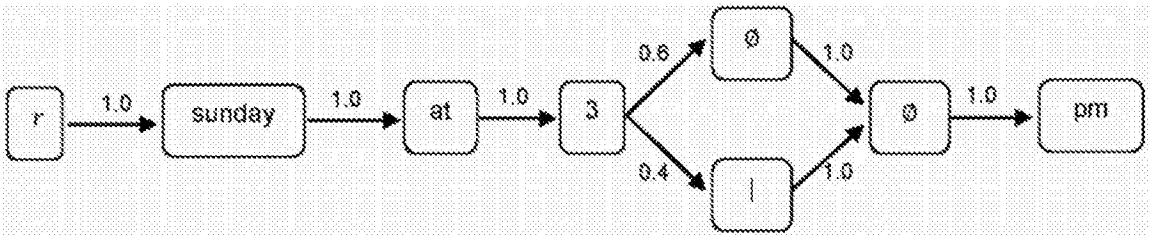
[0269] 通过在变化点上导入分支节点,反复拆解单一节点n^s。

[0270] 举例来说,考虑作用于原始序列"sunday at 3pm"上的PCSG构建算法。首先,步骤1:

[0271] 

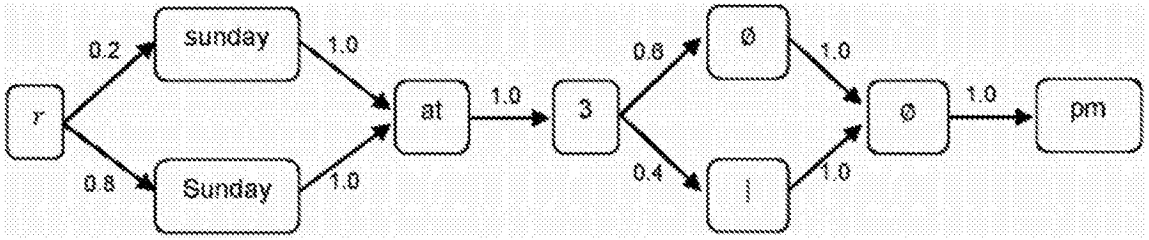
[0272] 本发明系统部署了概率断词器,结果如下:

[0273]



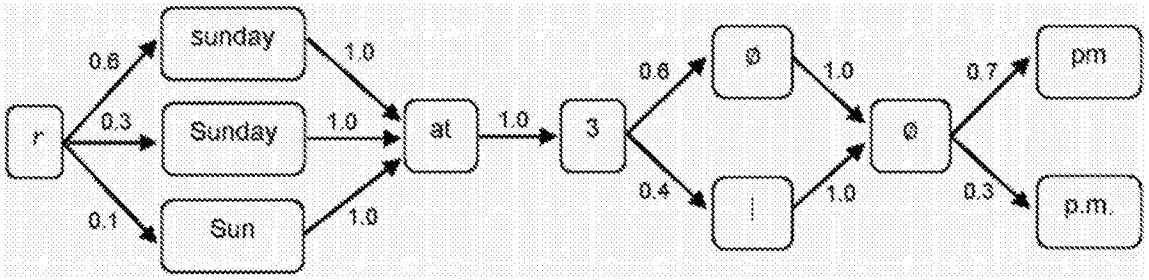
[0274] 值得注意的是，因为PCSG的上述特性3，使修改总是以出现分支-和-重新组合的结构插入形式出现，存在节点分支的特殊情况下，上述修改便于后续处理，因为其不会影响全部路径的概率。根据模型加入边概率，这一过程将在下文中详细介绍。继续这一算法，字形变体分析器被部署。

[0275]



[0276] 最后，词汇变体分析器被配置。

[0277]



[0278] 值得注意的是，因为PCSG的特性3，分支在再次形成分支之前必须集中。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。

[0279] 边概率优选被分配给PCSGs。边概率的分配优选参考了语境候选模型的参数。这些概率的直观解释有两种：

[0280] (1)这些概率表示用户打算输入被分配给特定分支的序列的概率估量。例如，如果用户已输入“Dear ben”，那么我们可赋予用户实际打算输入“Dear Ben”这种情况一定概率。

[0281] (2)这些概率表示特定分支为观察到的序列的有效正字法变体的“补偿”(backoff)概率。例如，如果用户已输入“See you on Thur”，那么“Thur”的可选正字法形式会是“Thurs”。

[0282] 被分配给特定边的概率还会受到被赋予某一背景模型信息的正字法变体的估算概率的影响。例如，语境序列模型S实际上可被重新使用，以获得不同正字法变体的概率估量，语境序列模型可与其他概率测度一起使用以生成分支概率。如此利用语境序列模型意味着语境候选模型C实际上包含语境序列模型S的实例，这明显违反了候选与序列模型之间的独立假设(上述特性7)。但是，实际上在语境中永远不会碰到这种假设，因此语境序列模

型的上述用法还是相对安全的。

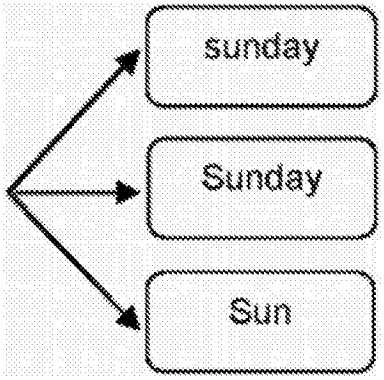
[0283] 一实例将有助于理解上述内容。在一优选实施例中，假设语境候选模型使用下列算法分配概率：

[0284] 被观察到的序列接受0.8概率；其它平均接受剩余概率；

[0285] 根据语境序列模型估量调整概率值；

[0286] 归一化概率值，以使概率值符合上述的PCSG特性(19)。

[0287] 从上述PCSG实例可知，下列分支可被考虑：

[0288] 

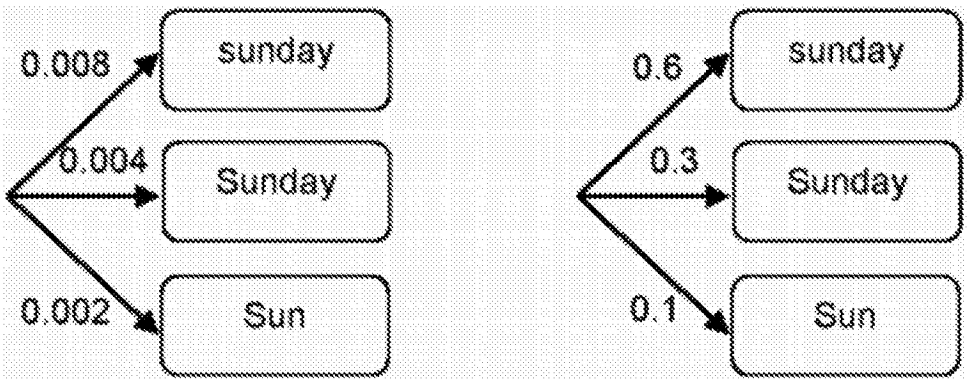
[0289] 因为“sunday”是最初观察值，因此其首先被上述算法的第一步骤分配了概率值0.8，而其他边则各被分配0.1。举例来说，由语境序列模型返回的估量如下：

[0290] $P(\text{"sunday"}|C^S)=0.01$

[0291] $P(\text{"Sunday"}|C^S)=0.04$

[0292] $P(\text{"Sun"}|C^S)=0.02$

[0293] 其中C^s表示在这种情况下语境候选模型正在使用语境序列模型。因此，在本实例中，被分配给各条边的未归一化概率及归一化(被四舍五入的)概率(分别)如下：

[0294] 

[0295] 先验语境估量$P(\text{context}|M_{\text{context}})$

[0296] 可通过归一化与语境相关的初始序列t近似求出先验语境估量$P(\text{context}|M_{\text{context}})$。

[0297] $$P(context|M_{context}) \cong \frac{freq(t)}{\sum_{t'} freq(t')} \qquad (21)$$

[0298] 其中freq(t)是训练数据中序列t的频率，而分母则是训练数据中所有序列的频率和。序列“t”为语境序列30。先验语境根据其内能够抽出预测的相关模型M_{context}包含给定语境序列30的概率，加权语境候选的概率值。为此，语境先验根据表达式估量加权预测值。

[0299] 实际上，这一估量可被平滑处理，例如通过估算不可见序列的出现频率实施这一平滑处理。

[0300] 先验目标序列估量P(s|Mcontext)

[0301] 类似的，可通过使用针对训练数据的平滑频率分析估算先验目标序列P(s|Mcontext)，即表达式(17)的分母，例如，可通过归一化语境训练数据中全部序列的目标序列频率，近似求出先验目标序列：

[0302] $$P(s|M_{context}) \cong \frac{freq(s)}{\sum_{s'} freq(s')}$$

[0303] 其中，freq(s)为训练数据中目标序列的频率，而分母为训练数据中全部目标序列的频率之和。该分母约等于训练数据(包括副本)中词条的总数量。

[0304] 候选显示器3

[0305] 候选生成器将作为一条或多条文本预测的一条或多条最高排名候选输出给候选显示器3，以将文本预测呈现给用户。最高排名的候选是那些包含通过等式(11)确定的最大概率值的候选。在等式(11)中Z的计算是优选过程但不是必要过程：

[0306] $$\frac{P(s|R)P(context|s, M_{context})P(input|s, M_{input})}{Z}$$

[0307] 在一优选实施例中，候选40被显示在虚拟键盘旁边的候选显示器3上，以供用户选择。当某一候选被选中时，该候选替换掉对应于生成该候选的输入序列20的那一部分输入文本。

[0308] 在一可选实施例中，用户在未选择候选40的情况下连续打字。如果在呈现给用户多词条候选之后连续提供不会改变最高排名候选的第一个词条的输入，则可从预测中提取出该第一个词条，并可使其“遵从”撰写中的文本(由此还构成了一部分语境)。图2a、2b中示出了一个实例。

[0309] 如图2a所示，候选生成器2向候选显示器3输出了两个排名最高的候选40，在本实例中分别为“please call me”和“please call Melanie”。

[0310] 可选地，如图2b所示，候选生成器2将排名最高的候选40连同逐字输入一起输出给候选显示器3。在本实例中，最高的候选40和逐字输入分别为“call me back”和“calkmebac”。

[0311] 值得注意的是，图2a、2b的用户界面示例并未包括传统的“空格”键。与候选显示器和选择界面组合使用的本发明系统支持那种不需要空格键的文本输入界面。没有空格键的键盘所带来的好处是其节省了宝贵的屏幕空间，使其它按键取得了更多的空间或缩小了屏幕的尺寸。

[0312] 本发明系统的使用实例

[0313] 考虑了用户打出消息“When you get this,send me a reply”的情形。进一步考虑了在某些步骤中，我们的输入显示栏出现了如下显示状态：

[0314] When you get this, send me are|

[0315] 在这种情况下，输入序列生成器1会输出下列语境30和输入序列20：

[0316] 语境序列30:“When you get this,send me”

[0317] ●输入序列20:{{a,0.8},{s,0.1},{z,0.1}},

[0318] {{r,0.9},{t,0.05},{f,0.05}},

[0319] {{e,0.8},{w,0.2}}

[0320] 如图3所示,候选生成器2计算出输入模型估量P(input|s,M_{input}),以生成下列带有相关概率的候选:

[0321] {are,0.576}

[0322] {arena,0.576}

[0323] {ate,0.032}

[0324] {a text,0.032}

[0325] {a reply,0.576}

[0326] {a repot,0.576}

[0327] 图3示出了部分图形估算的可视实例,其中突出显示了估算中各阶段上的候选概率。

[0328] 值得注意的是,如图3所示,一旦形成了多个词条“a repot”(与“a reply”和“a text”相比,其为一个非常小概率的双连词),因为该词条具有较低的语境概率,因此由候选生成器2对具有该词条进行修剪。

[0329] 在候选集的结尾,存在一些不属于预期序列的最高得分候选。

[0330] 然而,通过使用被赋予同一候选和语境序列30的语境模型$M_{context}$(本例中为三连词),候选生成器2可生成带有相关概率值的下列候选:

[0331] {[send me]are,0.0000023}

[0332] {[send me]arena,0.000000002}

[0333] {[send me]ate,0.00000007}

[0334] {[send me]a text,0.007}

[0335] {[send me]a reply,0.005}

[0336] 就被关注的语境而言,最佳的候选实际上是“a text”。最后,由候选生成器2将由两个模型生成的概率估量组合在一起。如果需要,当然可以在不影响概率估量相对排名的情况下对这些概率估量进行归一化处理。处于使本文简洁的考虑,在此不再赘述。

[0337] 由此生成的最终实例估量如下:

[0338] {are,0.00000132}

[0339] {arena,0.00000000159}

[0340] {ate,0.00000000224}

[0341] {a text,0.000224}

[0342] {a reply,0.00288}

[0343] 因此在这一基础实例中,如果用户仅仅击打了R键而不是T键,则模型的组合集合正确地选出作为头等预测的“a reply”,以及在本例中明显更像候选的第二预测“a text”。

[0344] 如果事实正相反,用户击打了T键而不是R键,则在支持T字符的情况下输入概率字符串的分布将被颠倒过来,头两个预测结果也被简单的颠倒过来。不管哪种方式,都使预期结果拥有了非常高的排名,并准备好将其通过候选显示器3呈献给用户。

[0345] 如图4所示的对于本发明系统的介绍，通用方法包括：使用候选生成器2从输入序列20中生成一个或多个候选的步骤(100)，其中各候选包括两个以上由一个多个词条边界分隔开的词条；将第一概率估量分配给各候选的步骤(200)，具体为：通过在语境语言模型中搜索该候选的一个或多个词条，将第一概率估量分配给各候选，其中，语境语言模型包括词条序列，各词条序列包括对应的事件概率；并将对应于来自语境语言模型的候选的一个或多个词条的概率分配给该候选。上述方法的最后一步包括：根据对应的第一概率估量丢弃掉一个或多个候选的步骤(300)。

[0346] 本发明方法的其它方面与上述系统相似，例如，在本发明方法的一个优选实施例中，生成一个或多个候选的步骤包括使用候选生成器2从输入序列中生成概率约束序列图(PCSG)并插入词条边界节点。

[0347] 如涉及到一个在PCSG中实施结构(特别是通配符分支)归一化以确定语境候选估量的系统的上文所述，在本发明方法的一个优选实例中，序列预测集合中的至少一个对应于由用户输入至用户界面中的文本的调整版本或纠正版本。

[0348] 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、改进等，均应包含在如权利要求所限定的本发明的保护范围之内。

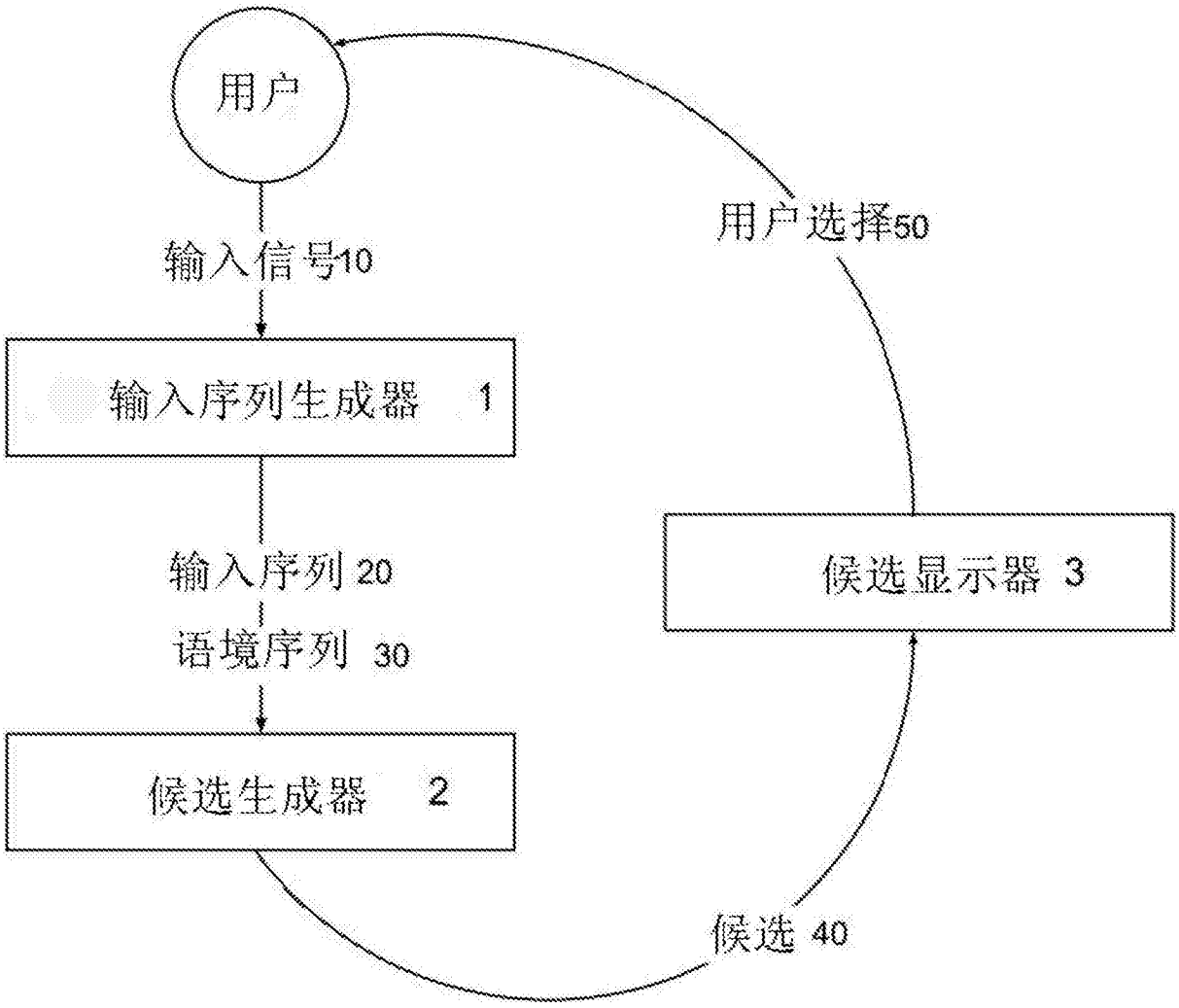

用户
输入信号10
输入序列生成器 1
输入序列 20
语境序列 30
候选生成器 2
候选 40
候选显示器 3
用户选择50

图1

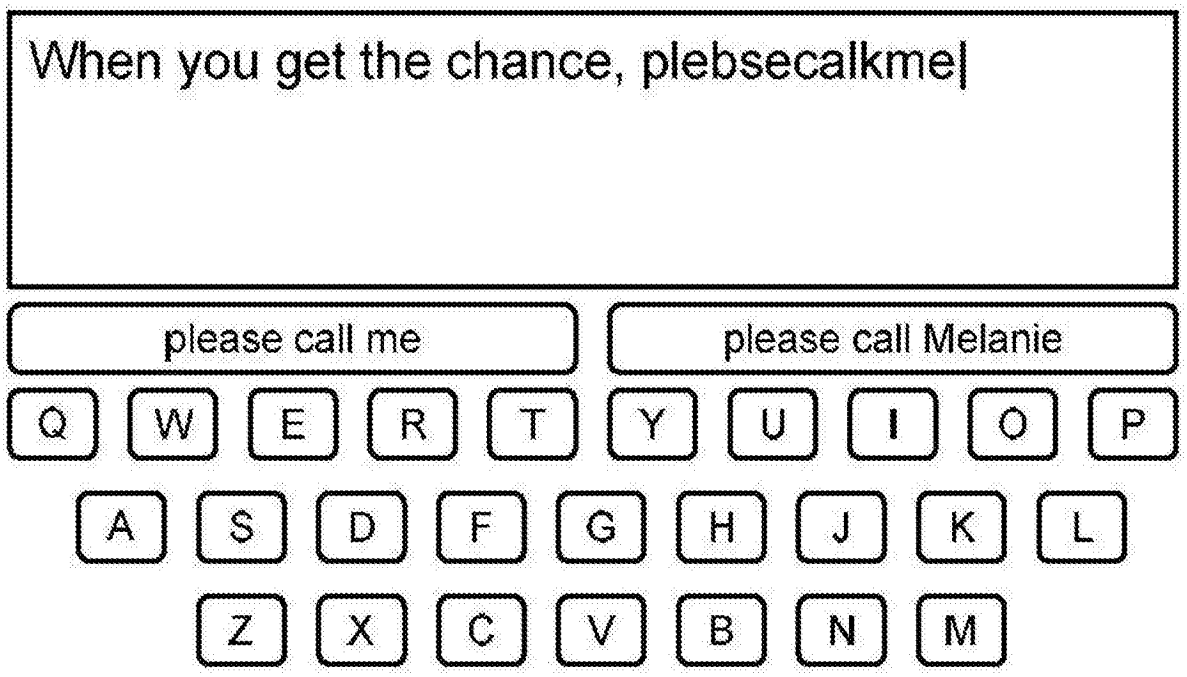

When you get the chance, plebsecalkme
please call me
please call Melanie
Q W E R T Y U I O P
A S D F G H J K L
Z X C V B N M

图2a

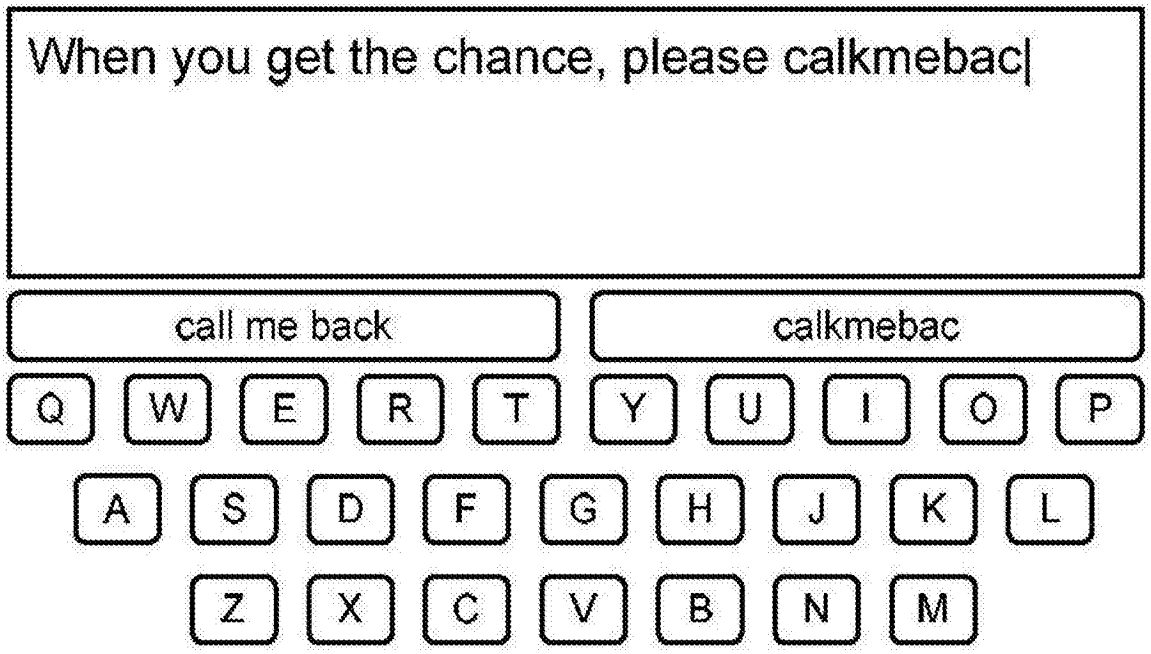

When you get the chance, please calkmebac
call me back
calkmebac
Q W E R T Y U I O P
A S D F G H J K L
Z X C V B N M

图2b

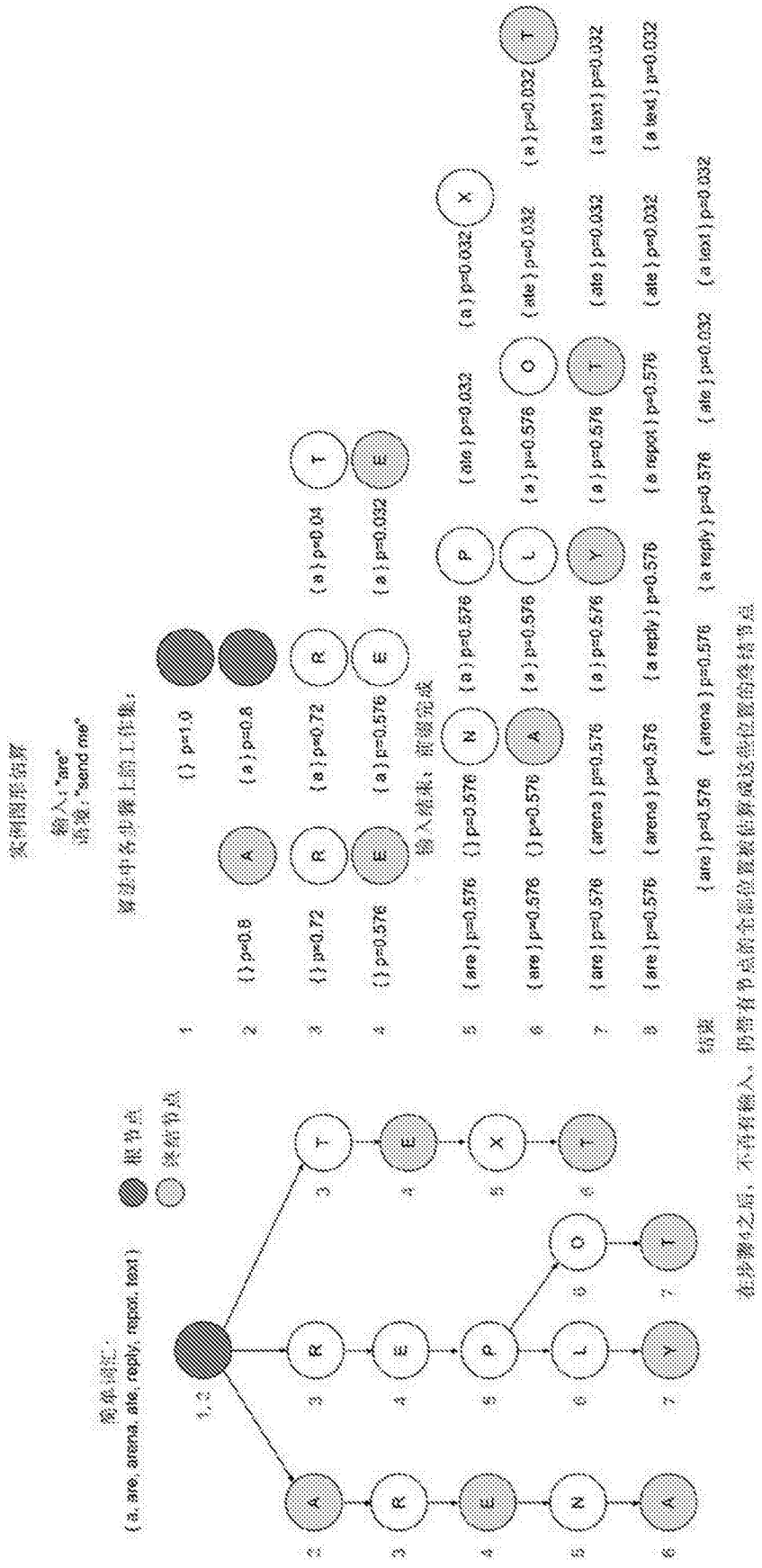

实例图形结果
输入："are"
语境："send me"
算法中各步骤上的工作集：
简单词汇：
{ a, are, arena, ate, reply, repot, text }
根节点
终结节点
输入结束，计算完成
在步骤4之后，不再有输入，仍然有节点所在位置的计算结果就是这些位置的终结节点。

图3

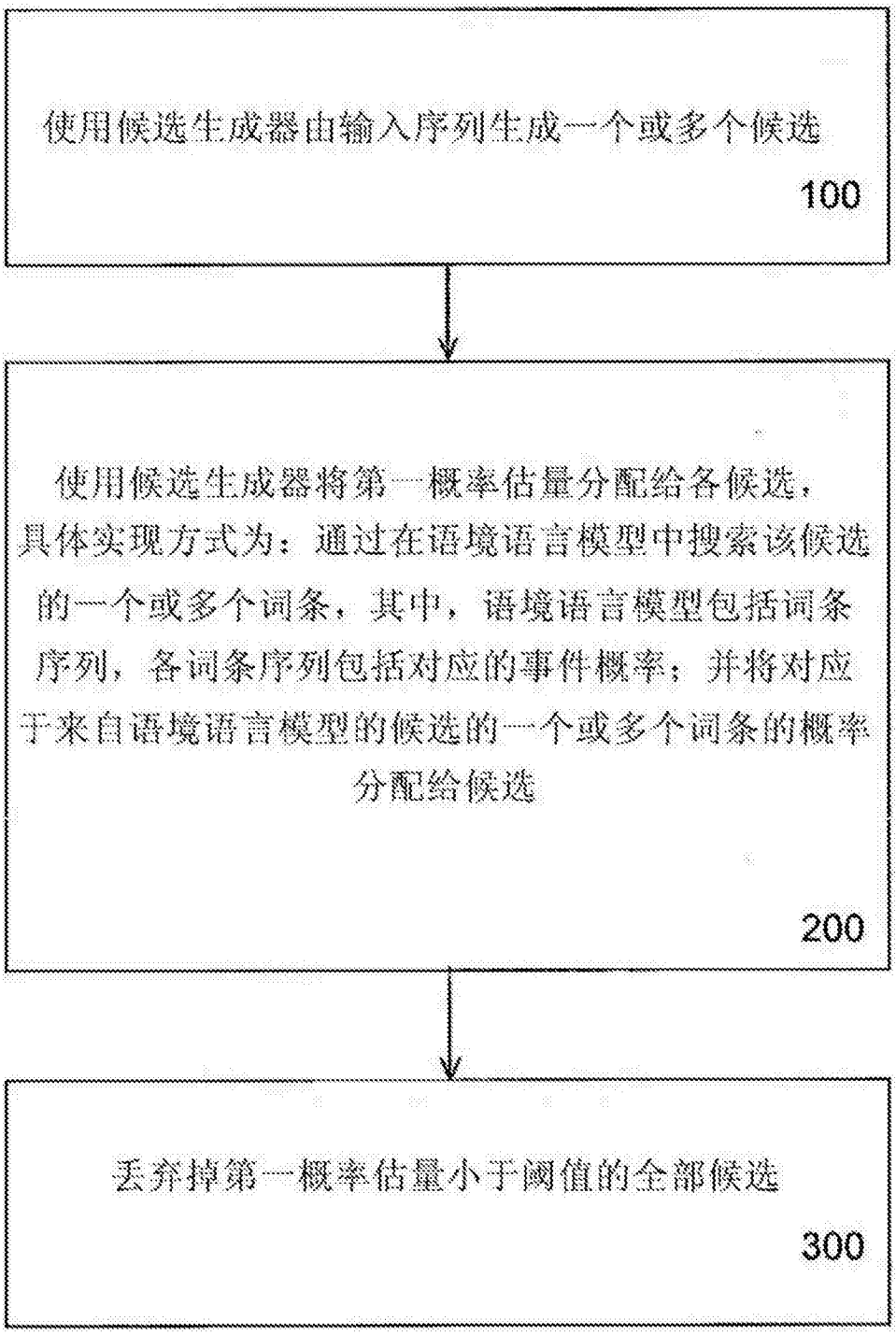

使用候选生成器由输入序列生成一个或多个候选
100
使用候选生成器将第一概率估量分配给各候选，具体实现方式为：通过在语境语言模型中搜索该候选的一个或多个词条，其中，语境语言模型包括词条序列，各词条序列包括对应的事件概率；并将对应于来自语境语言模型的候选的一个或多个词条的概率分配给候选
200
丢弃掉第一概率估量小于阈值的全部候选
300

图4