



(12) 发明专利申请

(10) 申请公布号 CN 114154487 A

(43) 申请公布日 2022.03.08

(21) 申请号 202111489580.1

(22) 申请日 2021.12.08

(71) 申请人 和美(深圳)信息技术股份有限公司

地址 518040 广东省深圳市福田区沙头街
道泰然工贸园苍松大厦1301室

(72) 发明人 张俊峰 王艳飞 胡茂海 胡碧峰
刘庆

(74) 专利代理机构 北京锤维联合知识产权代理
有限公司 11579

代理人 安娜

(51) Int. Cl.

G06F 40/232 (2020.01)

G06N 3/04 (2006.01)

G06N 3/08 (2006.01)

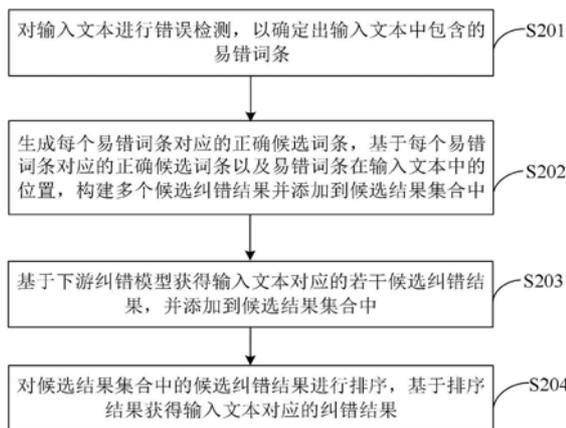
权利要求书2页 说明书13页 附图2页

(54) 发明名称

文本自动纠错方法、装置、电子设备及存储
介质

(57) 摘要

本申请涉及一种文本自动纠错方法、装置、
电子设备及存储介质,结合了字词层面和句子层
面的纠错方式进行纠错,在保证纠错效率的前提
下,可以对文本进行更全面有效的纠错,该方法
包括:对输入文本进行错误检测,以确定出所述
输入文本中包含的易错词条;生成每个易错词条
对应的正确候选词条,基于每个易错词条对应的
正确候选词条以及易错词条在所述输入文本中
的位置,构建多个候选纠错结果并添加到候选结
果集合中;基于下游纠错模型获得所述输入文本
对应的若干候选纠错结果,并添加到所述候选结
果集合中;对所述候选结果集合中的候选纠错结
果进行排序,基于排序结果获得所述输入文本对
应的纠错结果。



1. 一种文本自动纠错方法,其特征在于,包括:

对输入文本进行错误检测,以确定出所述输入文本中包含的易错词条;

生成每个易错词条对应的正确候选词条,基于每个易错词条对应的正确候选词条以及易错词条在所述输入文本中的位置,构建多个候选纠错结果并添加到候选结果集合中;

基于下游纠错模型获得所述输入文本对应的若干候选纠错结果,并添加到所述候选结果集合中;

对所述候选结果集合中的候选纠错结果进行排序,基于排序结果获得所述输入文本对应的纠错结果。

2. 根据权利要求1所述的方法,其特征在于,所述对输入文本进行错误检测,包括:

对所述输入文本进行分词处理,以获得若干分词;

对于包含拼音的分词,将分词中的拼音转换为中文,并将转换后的分词确定为易错词条;

对于仅包含中文的分词,将分词转换为对应的拼音,再将对应的拼音转换为中文,以获得转换后的分词,若转换后的分词和对应的原始分词不一致、且转换后的分词和原始分词的文本相似度大于相似度阈值,则将转换后的分词确定为易错候选词条,获取由易错候选词条对应的原始分词和位于原始分词前后的若干分词组成的原始文本,用易错候选词条替换原始文本中的原始分词,以获得拼接文本,利用语言模型分别计算原始文本和拼接文本的语言得分,若拼接文本的语言得分大于原始文本的语言得分,则将原始分词确定为易错词条。

3. 根据权利要求2所述的方法,其特征在于,所述将分词中的拼音转换为中文,包括:

基于预先构建的初始概率分布、状态转移概率分布和状态-观测值概率分布,结合维特比算法,找到将分词中的拼音转换为中文的最优路径,用最优路径对应的中文替换分词中的拼音,以获得转换后的分词,其中,所述初始概率分布记录了每个字处于文本开始位置的概率,所述状态转移概率分布记录了一个字转移到另外一个字的概率,所述状态-观测值概率分布记录了某个字是某个读音的概率分布。

4. 根据权利要求1所述的方法,其特征在于,所述错误检测包括以下至少一种检测方式:

检测所述输入文本中是否存在属于易错混淆集合的词语,将属于易错混淆集合的词语确定为易错词条;或

对所述输入文本进行分词处理,将不属于预设词典的分词确定为易错词条;或

按照2-gram模型将所述输入文本切分成若干二元分词,利用语言模型获得每个二元分词的语言得分;按照3-gram模型将所述输入文本切分成若干三元分词,利用语言模型获得每个三元分词的语言得分;针对所述输入文本中每个位置的字,根据包含该位置的字的二元分词和三元分词的语言得分,确定该位置的字的语言得分;基于每个位置的字的语言得分和预设的得分阈值,确定出所述输入文本中的易错位置,基于易错位置的字获得易错词条。

5. 根据权利要求1至4任一项所述的方法,其特征在于,所述生成每个易错词条对应的正确候选词条,包括:

对易错词条中的每个字,用对应的同音字、音似字、形似字、易混淆字进行替换,以获得

易错词条对应的若干正确候选词条；和/或

通过预测模型对所述输入文本中易错词条所在位置进行掩码操作并生成易错词条所在位置对应的预测结果,将预测结果作为易错词条对应的正确候选词条。

6. 根据权利要求1至4任一项所述的方法,其特征在于,所述下游纠错模型包括基于Bi-LSTM的检测网络和基于ALBERT的纠错网络;

所述基于下游纠错模型获得所述输入文本对应的若干候选纠错结果,包括:

将所述输入文本对应的向量表示 $E = (e_1, e_2, e_3, e_4, \dots, e_n)$ 输入所述检测网络,以获得所述输入文本中每个字的错误概率,其中, e_i 表示所述输入文本中第*i*个字对应的向量;

基于如下公式获得所述输入文本中每个字的新向量:

$$e'_i = p_i * e_{i,mask} + (1 - p_i) * e_i,$$

其中, e'_i 表示第*i*个字的新向量, $e_{i,mask}$ 表示通过预测模型对向量表示 E 中的 e_i 进行掩码操作后获得的针对 e_i 所在位置的预测向量, p_i 表示第*i*个字的错误概率;

将 $E' = (e'_1, e'_2, e'_3, e'_4, \dots, e'_n)$ 输入所述纠错网络,获得若干候选纠错结果。

7. 根据权利要求1至4任一项所述的方法,其特征在于,所述对所述候选结果集合中的候选纠错结果进行排序,包括:

基于各个候选纠错结果的局部特征和全局特征,对所述候选结果集合中的候选纠错结果进行排序,其中,所述局部特征包括:候选纠错结果中各个纠错后的词条的频次特征、切词特征、音形特征和PMI特征中的至少一种,所述全局特征包括以下至少一种:候选纠错结果的语言模型得分,候选纠错结果和对应的原始文本之间的相似度。

8. 一种文本自动纠错装置,其特征在于,包括:

错误检测模块,用于对输入文本进行错误检测,以确定出所述输入文本中包含的易错词条;

第一候选模块,用于生成每个易错词条对应的正确候选词条,基于每个易错词条对应的正确候选词条以及易错词条在所述输入文本中的位置,构建多个候选纠错结果并添加到候选结果集合中;

第二候选模块,用于基于下游纠错模型获得所述输入文本对应的若干候选纠错结果,并添加到所述候选结果集合中;

排序模块,用于对所述候选结果集合中的候选纠错结果进行排序,基于排序结果获得所述输入文本对应的纠错结果。

9. 一种电子设备,包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序,其特征在于,所述处理器执行所述计算机程序时实现权利要求1至7任一项所述方法的步骤。

10. 一种计算机可读存储介质,其上存储有计算机程序指令,其特征在于,该计算机程序指令被处理器执行时实现权利要求1至7任一项所述方法的步骤。

文本自动纠错方法、装置、电子设备及存储介质

技术领域

[0001] 本申请涉及自然语言处理技术领域,尤其涉及一种文本自动纠错方法、装置、电子设备及存储介质。

背景技术

[0002] 随着信息技术的快速发展,越来越多的用户可以通过计算机访问各种信息系统或网站。然而,用户在网站搜索内容时难免会输入错误的搜索数据或查询内容,从而导致系统或网站向用户展示了不是用户想要的的数据。目前主流的技术方案中,大多比较依赖用户的搜索和点击日志数据,基于搜索和点击日志数据通过概率学方式将较为可能的候选项返回给用户。针对在没有用户日志的前提,项目处于冷启动阶段时行之有效的搜索自动纠错方案相对较少且纠错效果不佳。

发明内容

[0003] 本申请实施例提供一种文本自动纠错方法、装置、电子设备及存储介质,提供一种在项目冷启动阶段时行之有效的文本自动纠错方案。

[0004] 第一方面,本申请一实施例提供了一种文本自动纠错方法,包括:

[0005] 对输入文本进行错误检测,以确定出所述输入文本中包含的易错词条;

[0006] 生成每个易错词条对应的正确候选词条,基于每个易错词条对应的正确候选词条以及易错词条在所述输入文本中的位置,构建多个候选纠错结果并添加到候选结果集合中;

[0007] 基于下游纠错模型获得所述输入文本对应的若干候选纠错结果,并添加到所述候选结果集合中;

[0008] 对所述候选结果集合中的候选纠错结果进行排序,基于排序结果获得所述输入文本对应的纠错结果。

[0009] 可选地,所述对输入文本进行错误检测,包括:

[0010] 对所述输入文本进行分词处理,以获得若干分词;

[0011] 对于包含拼音的分词,将分词中的拼音转换为中文,并将转换后的分词确定为易错词条;

[0012] 对于仅包含中文的分词,将分词转换为对应的拼音,再将对应的拼音转换为中文,以获得转换后的分词,若转换后的分词和对应的原始分词不一致、且转换后的分词和原始分词的文本相似度大于相似度阈值,则将转换后的分词确定为易错候选词条,获取由易错候选词条对应的原始分词和位于原始分词前后的若干分词组成的原始文本,用易错候选词条替换原始文本中的原始分词,以获得拼接文本,利用语言模型分别计算原始文本和拼接文本的语言得分,若拼接文本的语言得分大于原始文本的语言得分,则将原始分词确定为易错词条。

[0013] 可选地,所述将分词中的拼音转换为中文,包括:

[0014] 基于预先构建的初始概率分布、状态转移概率分布和状态-观测值概率分布,结合维特比算法,找到将分词中的拼音转换为中文的最优路径,用最优路径对应的中文替换分词中的拼音,以获得转换后的分词,其中,所述初始概率分布记录了每个字处于文本开始位置的概率,所述状态转移概率分布记录了一个字转移到另外一个字的概率,所述状态-观测值概率分布记录了某个字是某个读音的概率分布。

[0015] 可选地,所述错误检测包括以下至少一种检测方式:

[0016] 检测所述输入文本中是否存在属于易错混淆集合的词语,将属于易错混淆集合的词语确定为易错词条;或

[0017] 对所述输入文本进行分词处理,将不属于预设词典的分词确定为易错词条;或

[0018] 按照2-gram模型将所述输入文本切分成若干二元分词,利用语言模型获得每个二元分词的语言得分;按照3-gram模型将所述输入文本切分成若干三元分词,利用语言模型获得每个三元分词的语言得分;针对所述输入文本中每个位置的字,根据包含该位置的字的二元分词和三元分词的语言得分,确定该位置的字的语言得分;基于每个位置的字的语言得分和预设的得分阈值,确定出所述输入文本中的易错位置,基于易错位置的字获得易错词条。

[0019] 可选地,所述生成每个易错词条对应的正确候选词条,包括:

[0020] 对易错词条中的每个字,用对应的同音字、音似字、形似字、易混淆字进行替换,以获得易错词条对应的若干正确候选词条;和/或

[0021] 通过预测模型对所述输入文本中易错词条所在位置进行掩码操作并生成易错词条所在位置对应的预测结果,将预测结果作为易错词条对应的正确候选词条。

[0022] 可选地,所述下游纠错模型包括基于Bi-LSTM的检测网络和基于ALBERT的纠错网络;

[0023] 所述基于下游纠错模型获得所述输入文本对应的若干候选纠错结果,包括:

[0024] 将所述输入文本对应的向量表示 $E = (e_1, e_2, e_3, e_4, \dots, e_n)$ 输入所述检测网络,以获得所述输入文本中每个字的错误概率,其中, e_i 表示所述输入文本中第*i*个字对应的向量;

[0025] 基于如下公式获得所述输入文本中每个字的新向量:

$$[0026] \quad e'_i = p_i * e_{i, \text{mask}} + (1 - p_i) * e_i,$$

[0027] 其中, e'_i 表示第*i*个字的新向量, $e_{i, \text{mask}}$ 表示通过预测模型对向量表示 E 中的 e_i 进行掩码操作后获得的针对 e_i 所在位置的预测向量, p_i 表示第*i*个字的错误概率;

[0028] 将 $E' = (e'_1, e'_2, e'_3, e'_4, \dots, e'_n)$ 输入所述纠错网络,获得若干候选纠错结果。

[0029] 可选地,所述对所述候选结果集合中的候选纠错结果进行排序,包括:

[0030] 基于各个候选纠错结果的局部特征和全局特征,对所述候选结果集合中的候选纠错结果进行排序,其中,所述局部特征包括:候选纠错结果中各个纠错后的词条的频次特征、切词特征、音形特征和PMI特征中的至少一种,所述全局特征包括以下至少一种:候选纠错结果的语言模型得分,候选纠错结果和对应的原始文本之间的相似度。

[0031] 第二方面,本申请实施例提供了一种文本自动纠错装置,包括:

[0032] 错误检测模块,用于对输入文本进行错误检测,以确定出所述输入文本中包含的易错词条;

[0033] 第一候选模块,用于生成每个易错词条对应的正确候选词条,基于每个易错词条对应的正确候选词条以及易错词条在所述输入文本中的位置,构建多个候选纠错结果并添加到候选结果集合中;

[0034] 第二候选模块,用于基于下游纠错模型获得所述输入文本对应的若干候选纠错结果,并添加到所述候选结果集合中;

[0035] 排序模块,用于对所述候选结果集合中的候选纠错结果进行排序,基于排序结果获得所述输入文本对应的纠错结果。

[0036] 第三方面,本申请一实施例提供了一种电子设备,包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序,其中,处理器执行计算机程序时实现上述任一种方法的步骤。

[0037] 第四方面,本申请一实施例提供了一种计算机可读存储介质,其上存储有计算机程序指令,该计算机程序指令被处理器执行时实现上述任一种方法的步骤。

[0038] 第五方面,本申请一实施例提供了一种计算机程序产品或计算机程序,该计算机程序产品或计算机程序包括计算机指令,该计算机指令存储在计算机可读存储介质中。计算机设备的处理器从计算机可读存储介质读取该计算机指令,处理器执行该计算机指令,使得该计算机设备执行上述任一种TCP传输性能的控制的各种可选实现方式中提供的方法。

[0039] 本申请实施例提供的方案,一方面在字、词粒度层面对可能错误的位置做扩展召回生成字、词的纠错结果,另一方面利用下游纠错模型在句子层面直接给出可能的纠错结果,在保证纠错效率的前提下,可以对文本进行更全面有效的纠错,简单高效,具有高度可扩展性,在丰富若干领域专业词汇后可迅速落地应用场景。

附图说明

[0040] 为了更清楚地说明本发明实施例的技术方案,下面将对实施例中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图仅仅是本发明的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其它的附图。

[0041] 图1为本申请实施例提供的文本自动纠错方法的应用场景示意图;

[0042] 图2为本申请实施例提供的一种文本自动纠错方法的流程示意图;

[0043] 图3为本申请实施例提供的下游纠错模型的结构示意图;

[0044] 图4为本申请实施例提供的文本自动纠错装置的结构示意图;

[0045] 图5为本申请实施例提供的电子设备的结构示意图。

具体实施方式

[0046] 下面结合附图对本发明实施例进行详细描述。

[0047] 需说明的是,在不冲突的情况下,以下实施例及实施例中的特征可以相互组合;并且,基于本公开中的实施例,本领域普通技术人员在没有作出创造性劳动前提下所获得的所有其他实施例,都属于本公开保护的范围。

[0048] 需要说明的是,下文描述在所附权利要求书的范围内的实施例的各种方面。应显而易见,本文中所描述的方面可体现于广泛多种形式中,且本文中所描述的任何特定结构

及/或功能仅为说明性的。基于本公开,所属领域的技术人员应了解,本文中所描述的一个方面可与任何其它方面独立地实施,且可以各种方式组合这些方面中的两者或两者以上。举例来说,可使用本文中所阐述的任何数目个方面来实施设备及/或实践方法。另外,可使用除了本文中所阐述的方面中的一或多者之外的其它结构及/或功能性实施此设备及/或实践此方法。

[0049] 语言模型(Language Model,LM)的目的是建立一个能够描述给定词序列在语言中的出现的概率的分布。也就是说,语言模型是描述词汇概率分布的模型,一个能可靠反映语言识别时用词的概率分布的模型。语言模型在自然语言处理中占有重要的地位,在语音识别、机器翻译等领域得到了广泛应用。例如,利用语言模型能够得到语音识别多种词序列中可能性最大的一个词序列,或者给定若干词,预测下一个最可能出现的词语等。常用的语言模型包括N-Gram LM(N元语言模型)、Big-Gram LM(二元语言模型)、Tri-Gram LM(三元语言模型)。

[0050] N-gram是一种基于统计语言模型的算法,又被称为一阶马尔科夫链。它的基本思想是将文本里面的内容按照字节进行大小为N的滑动窗口操作,形成了长度是N的字节片段序列。每一个字节片段称为gram,对所有的gram的出现频度进行统计,并且按照事先设定好的阈值进行过滤,形成关键gram列表,也就是这个文本的向量特征空间。列表中的每一种gram就是一个特征向量维度。

[0051] 下面对本申请实施例的技术方案能够适用的应用场景做一些简单介绍,需要说明的是,以下介绍的应用场景仅用于说明本申请实施例而非限定。在具体实施时,可以根据实际需要灵活地应用本申请实施例提供的技术方案。

[0052] 参考图1,其为本申请实施例提供的文本自动纠错方法的应用场景示意图。其中,终端设备101和服务器102之间可通过无线或有线网络连接,终端设备101包括但不限于桌面计算机、移动电话、移动电脑、平板电脑、媒体播放器、智能可穿戴设备、智能电视、机器人等电子设备,服务器102可以是独立的物理服务器,也可以是多个物理服务器构成的服务器集群或者分布式系统,还可以是提供云服务、云数据库、云计算、云函数、云存储、网络服务、云通信、中间件服务、域名服务、安全服务、CDN、以及大数据和人工智能平台等基础云计算服务的云服务器。

[0053] 在搜索场景下,用户可通过终端设备101的搜索框输入文本,后台可对用户实时输入的文本进行错误检测,对错误文本进行纠错,并将纠错结果提供给用户选择,用户从中选择正确的文本后,终端设备101向服务器102发送用户选择的文本,服务器102根据该文本从网络或数据库中搜多相关内容并反馈给终端设备101。在文档编辑器中,终端设备101可对用户输入的文本进行错误检测,并对错误文本进行纠错,将可能的正确结果展示给用户,用户可从中选择一个文本替换原有的输入文本,实现文档编辑器自动对文本进行错误检测和纠错的功能,为用户提供更多便利。

[0054] 当然,本申请实施例提供的方法并无限用于图1所示的应用场景中,还可以用于其它可能的应用场景,本申请实施例并不进行限制。对于图1所示的应用场景的各个设备所能实现的功能将在后续的方法实施例中一并进行描述,在此先不过多赘述。

[0055] 为进一步说明本申请实施例提供的技术方案,下面结合附图以及具体实施方式对此进行详细的说明。虽然本申请实施例提供了如下述实施例或附图所示的方法操作步骤,

但基于常规或者无需创造性的劳动在所述方法中可以包括更多或者更少的操作步骤。在逻辑上不存在必要因果关系的步骤中,这些步骤的执行顺序不限于本申请实施例提供的执行顺序。

[0056] 下面结合图1所示的应用场景,对本申请实施例提供的技术方案进行说明。本申请实施例提供的任一文本自动纠错方法,可应用于终端设备或服务器等具备运算能力的电子设备,本申请不作限定。

[0057] 参考图2,本申请实施例提供一种文本自动纠错方法,具体包括以下步骤:

[0058] S201、对输入文本进行错误检测,以确定出输入文本中包含的易错词条。

[0059] 具体实施时,在进行错误检测之前,可先对输入文本进行预处理,例如:对输入文本做格式化字符串操作,具体包括编码统一、全角转半角、大写转小写、繁体转简体等操作,排除特殊字符、特殊格式对后续处理的干扰。然后,再对预处理后的输入文本进行错误检测。

[0060] 具体实施时,若输入文本较长,可根据语法特点将输入文本拆分成短句,然后分别对每个短句进行错误检测;若输入文本较短,可以直接对输入文本进行错误检测。

[0061] 具体实施时,可构建一个易错词条集合,用于存储检测出的易错词条以及易错词条在输入文本中出现的位置等信息,方便后续调用这些数据。

[0062] S202、生成每个易错词条对应的正确候选词条,基于每个易错词条对应的正确候选词条以及易错词条在输入文本中的位置,构建多个候选纠错结果并添加到候选结果集合中。

[0063] S203、基于下游纠错模型获得输入文本对应的若干候选纠错结果,并添加到候选结果集合中。

[0064] 其中,下游纠错模型是在开源数据和预训练模型的基础上,通过训练获得的针对文本纠错下游任务的新模型,该模型基于预训练模型构建下游任务的思想,选用ALBERT-tiny,在保证文本纠错能力的同时,还可以一定程度上避免预训练模型过大而导致模型预测效率低的问题。参考图3,下游纠错模型主要包括两个部分:基于Bi-LSTM的检测网络和基于ALBERT的纠错网络,检测网络用于预测文本中每个字的错误概率,纠错网络用于预测错误字的修正概率,检测网络的预测结果将作为纠错网络的输入。

[0065] 下游纠错模型的训练数据主要来源于已有的易混淆集合、同音字集合、同形字集合等,利用公开和网络爬虫获取的百万级新闻数据集,按照一定比例随机替换成易混淆集合中词语、同音字、同形字、随机字符等,辅以部分人工标注构建而成。

[0066] 具体实施时,可采用现有的向量化方式(如word2vec、bert、xlnet、electra等模型),将输入文本转换为向量表示 $E = (e_1, e_2, e_3, e_4, \dots, e_n)$,其中, e_i 表示输入文本中第*i*个字对应的向量, e_i 具体是字的向量(Word Embedding)、位置向量(Position Embedding)、字类型向量(Segment Embedding)的和,这个和ALBERT的输入一致。然后,将输入文本对应的向量表示 $E = (e_1, e_2, e_3, e_4, \dots, e_n)$ 输入检测网络。以获得输入文本中每个字的错误概率,并基于如下公式获得输入文本中每个字的新向量: $e'_i = p_i * e_{i,mask} + (1 - p_i) * e_i$,其中, e'_i 表示第*i*个字的新向量, $e_{i,mask}$ 表示通过预测模型对向量表示 E 中的 e_i 进行MASK操作后获得的针对 e_i 所在位置的预测向量, p_i 表示第*i*个字的错误概率。最后,将 $E' = (e'_1, e'_2, e'_3, e'_4, \dots, e'_n)$ 输入纠错网络,获得若干候选纠错结果。

[0067] 检测网络的输入是以字为单位的向量表示 $E = (e_1, e_2, e_3, e_4, \dots, e_n)$, 输出为一个标签序列 $L = (l_1, l_2, l_3, \dots, l_n)$, 其中, l_i 表示第 i 个字 x_i 的标签, $l_i = 1$ 表示字有错误, $l_i = 0$ 表示字无错误, 对于每个字, p_i 表示字 x_i 有错误标签的概率, p_i 越高表示字 x_i 出错的概率越高。检测网络是一个双向的LSTM, 每个位置的输出为前向和后向隐状态的拼接向量, 即:

$$[0068] \quad \vec{h}_i^d = LSTM(\vec{h}_{i-1}^d, e_i)$$

$$[0069] \quad \overleftarrow{h}_i^d = LSTM(\overleftarrow{h}_i^d, e_i)$$

$$[0070] \quad \mathbf{h}_i^d = \left[\vec{h}_i^d, \overleftarrow{h}_i^d \right]$$

[0071] 然后每个位置的隐状态, 连接全连接层, 输出 l_i 为错误字符的概率

$$[0072] \quad p_i = p_d(l_i = 1|X) = \sigma(W_d \mathbf{h}_i^d + b_d)$$

[0073] 然后对输入的向量表示 E 中的 e_i 进行MASK操作后, 对MASK操作后的向量进行预测, 获得 e_i 所在位置的预测向量, 即 $e_{i, \text{mask}}$ 。之后, 以 p_i 为权重对 $e_{i, \text{mask}}$ 和 e_i 加权求和:

$$[0074] \quad e'_i = p_i * e_{i, \text{mask}} + (1 - p_i) * e_i$$

[0075] 如果 i 位置错误的概率越高, 那么新向量的 e'_i 就和 $e_{i, \text{mask}}$ 越接近, 否则就和原始输入的 e_i 比较接近。

[0076] 纠错网络是一个基于ALBERT的多分类的序列模型, 输入是检测网络输出的新向量表示 $E' = (e'_1, e'_2, e'_3, e'_4, \dots, e'_n)$, 输出目标是正确的字符序列, 也就是纠错后的文本。纠错网络的结构主要参考预训练模型ALBERT-tiny的结构, ALBERT的最后一层, 通过全连接网络计算每个位置的最大概率的候选字, 在输入向量和检测网络最后一层的输出之间使用残差网络结构进行连接, 纠错网络的隐含状态与检测网络的输入通过残差连接的线性组合, 这样可以在训练时防止梯度消失, 最后一层使用softmax函数。

[0077] 基于预训练模型构建的下游纠错模型, 由进行错误检测的检测网络和进行错误纠正的纠错网络两个模块组成, 在保证纠错效率的前提下, 可以对输入文本进行有效的纠错。

[0078] S204、对候选结果集中的候选纠错结果进行排序, 基于排序结果获得输入文本对应的纠错结果。

[0079] 具体实施时, 可在构建的数据集上, 取多个维度特征对候选结果集中的候选纠错结果进行排序模型的训练, 基于该排序模型可以根据各个的候选纠错结果的特征, 给出各个候选纠错结果的置信度, 根据置信度从高到低的顺序对候选结果集合进行排序, 获得排序结果。

[0080] 在搜索场景中, 可以基于针对候选结果集合的排序结果, 将排在前 N 个的候选纠错结果展示给用户, 由用户从中选择合适的文本。在用户没有选择的情况下, 可自动选择置信度最高的候选纠错结果作为纠错结果。

[0081] 本申请实施例的文本自动纠错方法, 一方面在字、词粒度层面对可能错误的位置做扩展召回生成字、词的纠错结果, 另一方面利用下游纠错模型在句子层面直接给出可能的纠错结果, 在保证纠错效率的前提下, 可以对文本进行更全面有效的纠错, 简单高效, 是一种在项目冷启动阶段时行之有效的文本自动纠错方案。

[0082] 具体实施时, 可通过以下提供的一种或多种错误检测方式, 检测出输入文本中的

错误词条：

[0083] 第一种错误检测方式：基于预先构建的易错混淆集合对输入文本进行错误检测。

[0084] 其中，易错混淆集合中包括大量预先搜集的易发生错误或易混淆的词语，例如“负责”的正确表达是“富贵”，可从网络渠道获取易错混淆的词语，并通过人工整理的方式获得易错混淆集合，使用时，检测输入文本中是否存在属于易错混淆集合的词语，将属于易错混淆集合的词语确定为易错词条。具体地，可先对输入文本进行分词处理，以获得若干分词，检测获得的分词是否出现的易错混淆集合中，将出现在易错混淆集合中的分词确定为易错词条。

[0085] 第二种错误检测方式：基于预设词典对输入文本进行错误检测。

[0086] 其中，预设词典为自定义的词典，可包含常用词语以及某个领域的专用词语，包含的专用词语可根据本申请的方法所应用的领域进行调整，不作限定。具体地，可先对输入文本进行分词处理，以获得若干分词，将不属于预设词典的分词确定为易错词条。

[0087] 具体实施时，可通过现有的分词器（如结巴jieba分词器）和实体识别模型（如开源LTP实体识别工具）对输入文本进行分词处理。

[0088] 第三种错误检测方式：利用语言模型在字级别上对输入文本进行错误检测。

[0089] 具体实施时，按照2-gram模型将输入文本切分成若干二元分词，利用语言模型获得每个二元分词的语言得分；按照3-gram模型将输入文本切分成若干三元分词，利用语言模型获得每个三元分词的语言得分；针对输入文本中每个位置的字，根据包含该位置的字的二元分词和三元分词的语言得分，确定该位置的字的语言得分；基于每个位置的字的语言得分和预设的得分阈值，确定出输入文本中的易错位置，基于易错位置的字获得易错词条。

[0090] 下面以输入文本“我爱中国”为例，对第三种错误检测方式的具体实现过程进行说明。

[0091] 首先，按照2-gram模型，“我爱中国”可以拆分成“我爱”、“爱中”、“中国”3个二元分词，利用语言模型获得这3个二元分词的语言得分 $[a_1, a_2, a_3]$ ，按照3-gram模型，“我爱中国”可以拆分成“我爱中”、“爱中国”两个三元分词，利用语言模型获得这两个三元分词的语言得分 $[b_1, b_2]$ 。

[0092] 然后，利用二元分词和三元分词的语言得分，计算每个位置的字的语言得分。计算输入文本中每个位置的字的语言得分时，可以求包含该字的二元分词的语言得分的平均值，作为该字的二元得分，并基于包含该字的三元分词的语言得分获得该字的三元得分，然后，求二元得分和三元得分的平均值，作为该位置的字的语言得分。其中，在计算某个位置的字的三元得分时，如果该位置的字仅出现在一个三元分词中，直接将该三元分词的语言得分作为该字的三元得分；如果该位置的字出现在两个三元分词中，则该字位于中间位置的三元分词的权重为 $2/3$ ，该字位于边缘位置的三元分词的权重为 $1/3$ ，按照上述权重对两个三元分词的语言得分进行加权平均，以获得该字的三元得分；如果该位置的字出现在三个三元分词中，则求这三个三元分词的语言得分的平均值，作为该字的三元得分。

[0093] 例如，对于输入文本中的第一个字“我”，包含“我”字的二元分词仅有一个“我爱”，则第一个字“我”的二元得分为 a_1 ，包含“我”字的三元分词仅有一个“我爱中”，则第一个字“我”的三元得分为 b_1 ，第一个字“我”的语言得分即为 $c_1 = (a_1 + b_1) / 2$ ；对输入文本中的第二

个字“爱”，包含“爱”字的二元分词有“我爱”和“爱中”，则第二个字“爱”的二元得分为 $(a_1+a_2)/2$ ，包含“爱”字的三元分词有两个“我爱中”和“爱中国”，“爱”位于“我爱中”的中间位置，而“爱”位于“爱中国”的边缘位置，所以“我爱中”的权重为 $2/3$ ，“爱中国”的权重为 $1/3$ ，则第二个字“爱”的三元得分为 $(2b_1+b_2)/3$ ，因此，第二个字“爱”的语言得分即为 $c_2=[(a_1+a_2)/2+(2b_1+b_2)/3]/2$ ；同样的方法得到“中”的语言得分为 $c_3=[(a_2+a_3)/2+(b_1+b_2+b_2)/3]/2$ ，“国”的语言得分为 $c_4=[a_3+b_2]/2$ 。

[0094] 具体操作时，在求每个字的二元得分时，可按照预设规则，对所有二元分词的语言得分按序组成的第一得分序列进行补全操作，即在第一得分序列第一个位置之前增加一个第一个二元分词的语言得分，在第一得分序列最后一个位置之后增加一个最后一个二元分词的语言得分，例如，第一得分序列为 $[a_1, a_2, a_3]$ ，补全后的第一得分序列为 $[a_1, a_1, a_2, a_3, a_3]$ ，然后，对补全后的第一得分序列，按照长度为2的滑动窗口求平均，依次获得输入文本中每个位置的字的二元得分 $[(a_1+a_1)/2, (a_1+a_2)/2, (a_2+a_3)/2, (a_3+a_3)/2]$ 。在求每个字的三元得分时，可按照预设规则，对所有三元分词的语言得分按序组成的第二得分序列进行补全操作，即在第二得分序列第一个位置之前增加两个第一个三元分词的语言得分，在第二得分序列最后一个位置之后增加两个最后一个三元分词的语言得分，例如，第二得分序列为 $[b_1, b_2]$ ，补全后的第二得分序列为 $[b_1, b_1, b_1, b_2, b_2, b_2]$ ，然后，对补全后的第二得分序列，按照长度为3的滑动窗口求平均，依次获得输入文本中每个位置的字的三元得分 $[(b_1+b_1+b_1)/3, (b_1+b_1+b_2)/3, (b_1+b_2+b_2)/3, (b_2+b_2+b_2)/3]$ 。最后，求每个位置的字的二元得分和三元得分平均值，作为每个位置的字的语言得分。

[0095] 最后，基于每个位置的字的语言得分和预设的得分阈值，确定出输入文本中的易错位置。具体地，可先求序列 $[c_1, c_2, c_3, c_4]$ 中的中位数，记为 med ；然后，获得序列 $[|c_1-med|, |c_2-med|, |c_3-med|, |c_4-md|]$ ，其中符号“ $|$ ”表示绝对值，再求 $[|c_1-med|, |c_2-med|, |c_3-med|, |c_4-md|]$ 的中位数，记为 $med2$ ；接着，根据公式： $final_score=\alpha* [|c_1-med|, |c_2-med|, |c_3-med|, |c_4-md|]/med2$ ，获得每个位置的字的最终得分，若某个位置的最终得分大于预设的得分阈值，则认为该位置的字是易错字。

[0096] 第四种错误检测方式：利用语言模型在词级别上对输入文本进行错误检测。

[0097] 具体实施时，先对输入文本进行分词处理，以获得若干分词。可按照字级别，以2-gram和3-gram的方式对于输入文本进行分词。

[0098] 对于仅包含中文的分词，将分词转换为对应的拼音，再将对应的拼音转换为中文，以获得转换后的分词；若转换后的分词和对应的原始分词不一致、且转换后的分词和原始分词的文本相似度大于相似度阈值，则将转换后的分词确定为易错候选词条；获取由易错候选词条对应的原始分词和位于原始分词前后的若干分词组成的原始文本，用易错候选词条替换原始文本中的原始分词，以获得拼接文本；利用语言模型分别计算原始文本和拼接文本的语言得分，若拼接文本的语言得分大于原始文本的语言得分，则将原始分词确定为易错词条。

[0099] 其中，文本的语言得分越高，说明该文本越合理，出错的概率越低。因此，当拼接文本的语言得分大于原始文本的语言得分时，表明拼接文本相对于原始文本更加合理，即原始文本中被替换的原始分词可能存在错误。相似度阈值可根据实际应用场景确定，本申请实施例不作限定，例如可以是0.5。

[0100] 例如,输入文本切分后获得的分词结果为: W_1 、 W_2 、 W_3 、 W_4 、 W_5 。以分词 W_2 为例,先将分词 W_2 转换为对应的拼音,再将该拼音转换为中文 W_2' ,如果 W_2 和 W_2' 一致,则表明分词 W_2 不是易错词条;如果 W_2 和 W_2' 不一致,则计算 W_2 和 W_2' 的文本相似度,两者相似度越高,表明用户将 W_2' 误输入成 W_2 的概率越高,因此,如果文本相似度大于相似度阈值,则将 W_2' 确定为易错候选词条,否则 W_2' 确定不是易错候选词条。然后,获取由易错候选词条 W_2' 对应的原始分词 W_2 和位于原始分词 W_2 前后的若干分词组成的原始文本,此处可选择位于原始分词 W_2 前后的各一个分词 W_1 和 W_3 ,其组成的原始文本为 $W_1W_2W_3$;用易错候选词条 W_2' 替换原始文本 $W_1W_2W_3$ 中的原始分词 W_2 ,以获得拼接文本 $W_1W_2'W_3$ 。最后,利用语言模型分别计算原始文本 $W_1W_2W_3$ 和拼接文本 $W_1W_2'W_3$ 的语言得分,若拼接文本的语言得分大于原始文本的语言得分,则将原始分词 W_2 确定为易错词条。

[0101] 对于包含拼音的分词,将分词中的拼音转换为中文,并将转换后的分词确定为易错词条。

[0102] 具体实施时,可在开源数据的基础上获取到三个概率分布:初始概率分布、状态转移概率分布和状态-观测值概率分布。其中,初始概率分布记录了每个字处于文本开始位置的概率;状态转移概率分布即各个状态互相转换的概率分布,即一个状态转移到另一个状态的概率分布,在本申请实施例中,状态转移概率分布记录了一个字转移到另外一个字的概率;状态-观测值概率分布即状态到观测值的概率分布,在本申请实施例中,记状态-观测值概率分布录了某个字是某个读音的概率分布。

[0103] 基于预先构建的初始概率分布、状态转移概率分布和状态-观测值概率分布,结合维特比算法,找到将分词中的拼音转换为中文的最优路径,用最优路径对应的中文替换分词中的拼音,以获得转换后的分词。其中,维特比算法的大致过程为:获取拼音所对应的若干个中文字,从首字开始,记录首字的初始概率分布与当前字转移到下一个字的概率分布相乘,最后再乘以当前字是某个拼音的概率,最后对所有路径结果排序,取最大值路径上的文字即可组成转换后的中文。

[0104] 例如,“shebao缴纳”包含拼音,将通过这三个概率分布结合维特比算法,经过隐马尔科夫过程即可转换为“社保缴纳”。

[0105] 申请实施例中使用的语言模型,可以是百度开源的基于字符级别的NgramLM模型,该模型大小的约为2.8G,也可以是利用KenLM统计语言模型工具在开源数据训练的语言模型。

[0106] 基于上述多种错误检测方式,使得本申请的文本自动纠错方法可以覆盖大多数谐音、混淆音、形似字、全拼或包含拼音和中文的混合文本等用户常见的输入易错场景。尤其是在包含拼音的文本场景中,市面上常见的纠错方式是构建索引,在用户输入时对用户作出提示,而本申请提供的方案可以自动将文本中的拼音转换为常见的中文,在错误检测过程中也可以通过将词条转换为拼音再转换为词条,进而发掘出文本中潜在的错误位置,提高错误检测能力。

[0107] 在上述任一实施方式的基础上,可通过如下至少一种方式生成易错词条对应的正确候选词条:

[0108] 方式一:对易错词条中的每个字,用对应的同音字、音似字、形似字、易混淆字进行替换,以获得易错词条对应的若干正确候选词条。

[0109] 具体实施时,可预先构建同音字集合、音似字集合、形似字集合、易混淆字集合。其中,同音字集合中记录了各个字对应的同音字,如同、统和痛为同音字;音似字集合中记录了音似字,如音和硬为音似字;形似字集合中记录了形似字,如熔和溶为形似字;易混淆字集合中记录了其它易混淆的字。

[0110] 对于包含多个字的易错词条,可以对其中的每个字都进行同音字、音似字、形似字、易混淆字替换,通过各个替换字的组合,可获得多个正确候选词条。

[0111] 方式二:通过预测模型对输入文本中易错词条所在位置进行掩码(MASK)操作并生成易错词条所在位置对应的预测结果,将预测结果作为易错词条对应的正确候选词条。

[0112] 其中,预测模型可以是BERT、ERNIE、ELECTRA等。对输入文本中易错词条所在位置做MASK操作,即遮盖掉输入文本中的易错词条,然后基于预测模型对遮盖掉易错词条的输入文本进行预测,预测被遮盖部分的文本,即预测结果,若预测结果为多个,可取前n个预测结果作为易错词条对应的正确候选词条。

[0113] 基于上述方式一和方式二,使得本申请的方案具有高度可扩展性,在丰富若干领域专业词汇后可迅速落地应用场景。方式二采用预测模型的掩码操作对易错词条的纠错结果进行预测,从输入文本整体的语义层面出发,为纠错结果提供了更多可靠的选项,还有助于提高方案的普适性,扩展方案可应用的场景。

[0114] 在上述任一实施方式的基础上,可基于各个候选纠错结果的局部特征和全局特征,对候选结果集合中的候选纠错结果进行排序。具体可选择结合局部特征和全局特征的lightGBM排序模型,对候选结果集合中的候选纠错结果进行排序。通过综合局部特征和全局特征,可提高排序结果的准确度。

[0115] 其中,局部特征主要是统计的是词条的特征,具体可包括:候选纠错结果中各个纠错后的词条的频次特征、切词特征、音形特征和PMI特征中的至少一种。

[0116] 频次是指词条在训练语料中出现的频次,频次越多说明这个词条越常见。针对候选纠错结果中的每个纠错后的词条,统计该纠错后的词条在训练语料中出现的频次,以及该纠错后的词条对应的错误词条在训练语料中的频次,基于上述两个频次获得纠错前后的频次变化,即为该纠错后的词条的频次特征。

[0117] 针对候选纠错结果中的每个纠错后的词条,对该纠错后的词条进行分词处理,基于分词结果统计其包含的短语个数、单字个数、以及含错别字的文本片段的长度,同时对该纠错后的词条所对应的错误词条也进行相同的处理,获得错误词条包含的短语个数、单字个数、以及含错别字的文本片段的长度,基于上述统计的纠错前后的数据获得纠错前后的切词特征的变化,作为该纠错后的词条所对应的切词特征。

[0118] 音形特征:是指错误词条的拼音和纠错后的词条的拼音的Jaccard距离、以及拼音韵母变化。

[0119] PMI(Pointwise Mutual Information,点互信息)特征:是指错误词条和纠错后的词条的NGram的PMI最小值和最大值。

[0120] 其中,全局特征主要整个输入文本的特征,具体可包括至少一种:候选纠错结果的语言模型得分,候选纠错结果和对应的原始文本之间的相似度。可利用百度开源NGram语言模型,计算候选纠错结果的语言模型得分。利用预训练模型分别获取输入文本与候选纠错结果的语言模型向量特征,然后计算两者的相似度。

[0121] 如图4所示,基于与上述文本自动纠错方法相同的发明构思,本申请实施例还提供了一种文本自动纠错装置40,包括:

[0122] 错误检测模块401,用于对输入文本进行错误检测,以确定出所述输入文本中包含的易错词条;

[0123] 第一候选模块402,用于生成每个易错词条对应的正确候选词条,基于每个易错词条对应的正确候选词条以及易错词条在所述输入文本中的位置,构建多个候选纠错结果并添加到候选结果集合中;

[0124] 第二候选模块403,用于基于下游纠错模型获得所述输入文本对应的若干候选纠错结果,并添加到所述候选结果集合中;

[0125] 排序模块404,用于对所述候选结果集合中的候选纠错结果进行排序,基于排序结果获得所述输入文本对应的纠错结果。

[0126] 可选地,错误检测模块401具体用于:

[0127] 对所述输入文本进行分词处理,以获得若干分词;

[0128] 对于包含拼音的分词,将分词中的拼音转换为中文,并将转换后的分词确定为易错词条;

[0129] 对于仅包含中文的分词,将分词转换为对应的拼音,再将对应的拼音转换为中文,以获得转换后的分词,若转换后的分词和对应的原始分词不一致、且转换后的分词和原始分词的文本相似度大于相似度阈值,则将转换后的分词确定为易错候选词条,获取由易错候选词条对应的原始分词和位于原始分词前后的若干分词组成的原始文本,用易错候选词条替换原始文本中的原始分词,以获得拼接文本,利用语言模型分别计算原始文本和拼接文本的语言得分,若拼接文本的语言得分大于原始文本的语言得分,则将原始分词确定为易错词条。

[0130] 可选地,错误检测模块401中还包括拼音转中文单元,具体用于:基于预先构建的初始概率分布、状态转移概率分布和状态-观测值概率分布,结合维特比算法,找到将分词中的拼音转换为中文的最优路径,用最优路径对应的中文替换分词中的拼音,以获得转换后的分词,其中,所述初始概率分布记录了每个字处于文本开始位置的概率,所述状态转移概率分布记录了一个字转移到另外一个字的概率,所述状态-观测值概率分布记录了某个字是某个读音的概率分布。

[0131] 可选地,错误检测模块401中的所述错误检测包括以下至少一种检测方式:

[0132] 检测所述输入文本中是否存在属于易错混淆集合的词语,将属于易错混淆集合的词语确定为易错词条;或

[0133] 对所述输入文本进行分词处理,将不属于预设词典的分词确定为易错词条;或

[0134] 按照2-gram模型将所述输入文本切分成若干二元分词,利用语言模型获得每个二元分词的语言得分;按照3-gram模型将所述输入文本切分成若干三元分词,利用语言模型获得每个三元分词的语言得分;针对所述输入文本中每个位置的字,根据包含该位置的字的二元分词和三元分词的语言得分,确定该位置的字的语言得分;基于每个位置的字的语言得分和预设的得分阈值,确定出所述输入文本中的易错位置,基于易错位置的字获得易错词条。

[0135] 可选地,第一候选模块402,具体用于:

[0136] 对易错词条中的每个字,用对应的同音字、音似字、形似字、易混淆字进行替换,以获得易错词条对应的若干正确候选词条;和/或

[0137] 通过预测模型对所述输入文本中易错词条所在位置进行掩码操作并生成易错词条所在位置对应的预测结果,将预测结果作为易错词条对应的正确候选词条。

[0138] 可选地,所述下游纠错模型包括基于Bi-LSTM的检测网络和基于ALBERT的纠错网络;第二候选模块403,具体用于:

[0139] 将所述输入文本对应的向量表示 $E = (e_1, e_2, e_3, e_4, \dots, e_n)$ 输入所述检测网络,以获得所述输入文本中每个字的错误概率,其中, e_i 表示所述输入文本中第*i*个字对应的向量;

[0140] 基于如下公式获得所述输入文本中每个字的新向量:

[0141]
$$e'_i = p_i * e_{i, \text{mask}} + (1 - p_i) * e_i,$$

[0142] 其中, e'_i 表示第*i*个字的新向量, $e_{i, \text{mask}}$ 表示通过预测模型对向量表示 E 中的 e_i 进行掩码操作后获得的针对 e_i 所在位置的预测向量, p_i 表示第*i*个字的错误概率;

[0143] 将 $E' = (e'_1, e'_2, e'_3, e'_4, \dots, e'_n)$ 输入所述纠错网络,获得若干候选纠错结果。

[0144] 可选地,排序模块404,具体用于:基于各个候选纠错结果的局部特征和全局特征,对所述候选结果集中的候选纠错结果进行排序,其中,所述局部特征包括:候选纠错结果中各个纠错后的词条的频次特征、切词特征、音形特征和PMI特征中的至少一种,所述全局特征包括以下至少一种:候选纠错结果的语言模型得分,候选纠错结果和对应的原始文本之间的相似度。

[0145] 本申请实施例提的文本自动纠错装置与上述文本自动纠错方法采用了相同的发明构思,能够取得相同的有益效果,在此不再赘述。

[0146] 基于与上述文本自动纠错方法相同的发明构思,本申请实施例还提供了一种电子设备,该电子设备具体可以为机器人内部的控制设备或控制系统,也可以是与智能设备通信的外部设备,如桌面计算机、便携式计算机、智能手机、平板电脑、个人数字助理(Personal Digital Assistant, PDA)、服务器等。如图5所示,该电子设备50可以包括处理器501和存储器502。

[0147] 处理器501可以是通用处理器,例如中央处理器(CPU)、数字信号处理器(Digital Signal Processor, DSP)、专用集成电路(Application Specific Integrated Circuit, ASIC)、现场可编程门阵列(Field Programmable Gate Array, FPGA)或者其他可编程逻辑器件、分立门或者晶体管逻辑器件、分立硬件组件,可以实现或者执行本申请实施例中公开的各方法、步骤及逻辑框图。通用处理器可以是微处理器或者任何常规的处理器等。结合本申请实施例所公开的方法的步骤可以直接体现为硬件处理器执行完成,或者用处理器中的硬件及软件模块组合执行完成。

[0148] 存储器502作为一种非易失性计算机可读存储介质,可用于存储非易失性软件程序、非易失性计算机可执行程序以及模块。存储器可以包括至少一种类型的存储介质,例如可以包括闪存、硬盘、多媒体卡、卡型存储器、随机访问存储器(Random Access Memory, RAM)、静态随机访问存储器(Static Random Access Memory, SRAM)、可编程只读存储器(Programmable Read Only Memory, PROM)、只读存储器(Read Only Memory, ROM)、带电可擦除可编程只读存储器(Electrically Erasable Programmable Read-Only Memory,

EEPROM)、磁性存储器、磁盘、光盘等等。存储器是能够用于携带或存储具有指令或数据结构形式的期望的程序代码并能够由计算机存取的任何其他介质,但不限于此。本申请实施例中的存储器502还可以是电路或者其它任意能够实现存储功能的装置,用于存储程序指令和/或数据。

[0149] 本领域普通技术人员可以理解:实现上述方法实施例的全部或部分步骤可以通过程序指令相关的硬件来完成,前述的程序可以存储于一计算机可读取存储介质中,该程序在执行时,执行包括上述方法实施例的步骤;上述计算机存储介质可以是计算机能够存取的任何可用介质或数据存储设备,包括但不限于:移动存储设备、随机存取存储器(RAM, Random Access Memory)、磁性存储器(例如软盘、硬盘、磁带、磁光盘(MO)等)、光学存储器(例如CD、DVD、BD、HVD等)、以及半导体存储器(例如ROM、EPROM、EEPROM、非易失性存储器(NAND FLASH)、固态硬盘(SSD))等各种可以存储程序代码的介质。

[0150] 或者,本申请上述集成的单元如果以软件功能模块的形式实现并作为独立的产品销售或使用,也可以存储在一个计算机可读取存储介质中。基于这样的理解,本申请实施例的技术方案本质上或者说对现有技术做出贡献的部分可以以软件产品的形式体现出来,该计算机软件产品存储在一个存储介质中,包括若干指令用以使得一台计算机设备(可以是个人计算机、服务器、或者网络设备等)执行本申请各个实施例所述方法的全部或部分。而前述的存储介质包括:移动存储设备、随机存取存储器(RAM, Random Access Memory)、磁性存储器(例如软盘、硬盘、磁带、磁光盘(MO)等)、光学存储器(例如CD、DVD、BD、HVD等)、以及半导体存储器(例如ROM、EPROM、EEPROM、非易失性存储器(NAND FLASH)、固态硬盘(SSD))等各种可以存储程序代码的介质。

[0151] 以上所述,仅为本发明的具体实施方式,但本发明的保护范围并不局限于此,任何熟悉本技术领域的技术人员在本发明揭露的技术范围内,可轻易想到的变化或替换,都应涵盖在本发明的保护范围之内。因此,本发明的保护范围应以权利要求的保护范围为准。

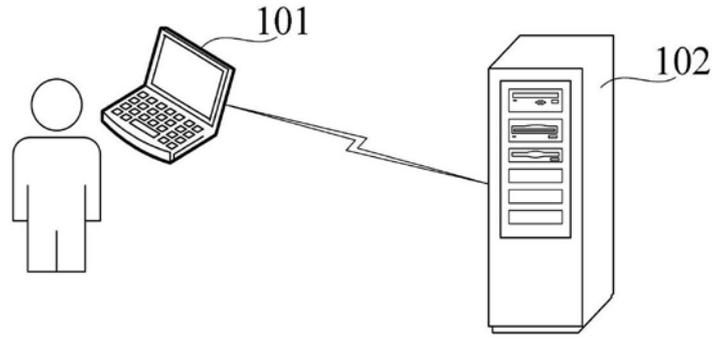


图1

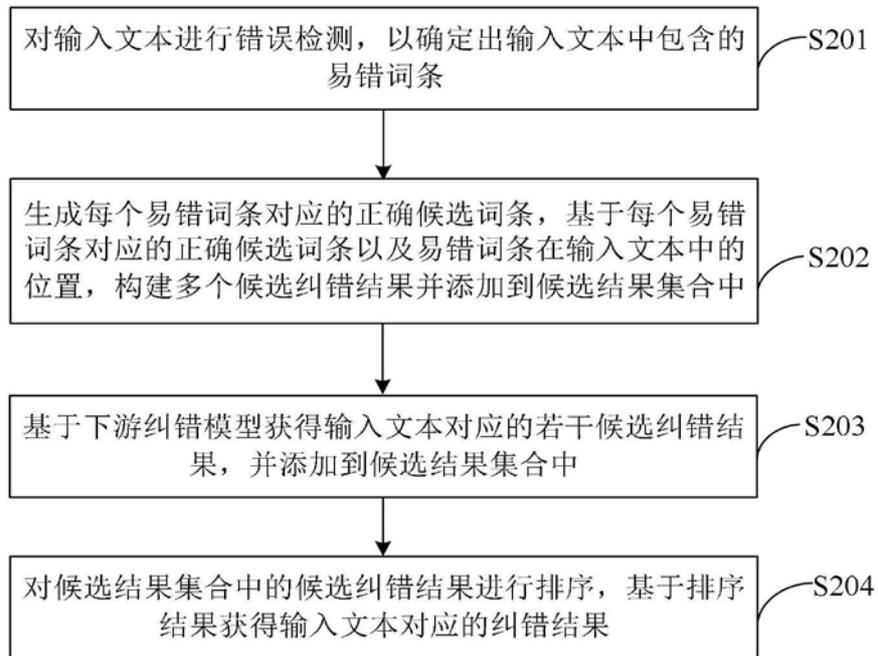


图2

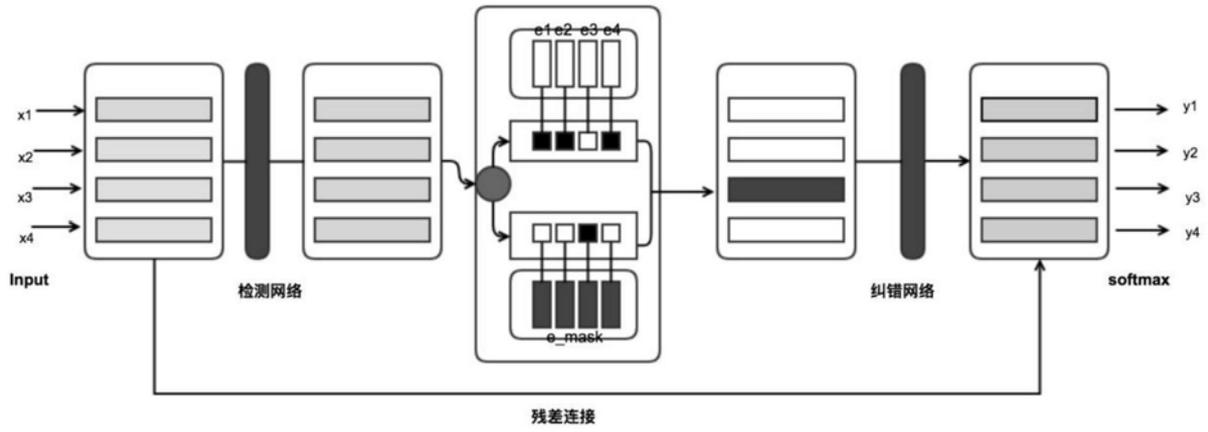


图3

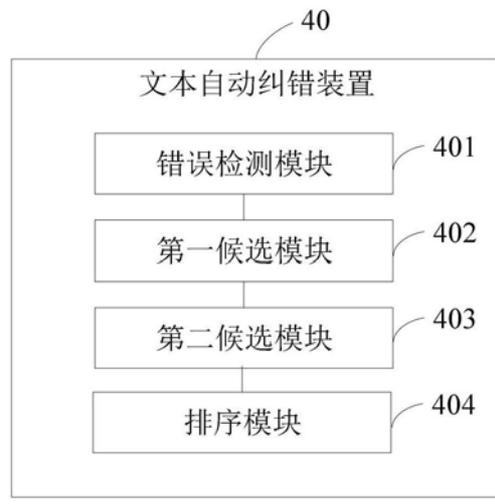


图4

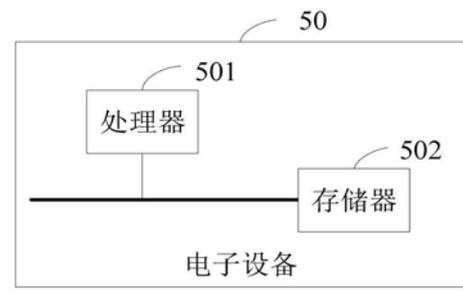


图5