**拼字游戏中的优化问题及解答**

**前言**

拼字游戏是在西方的青少年中非常流行的游戏形式，游戏开始前通常会指定一本参考词典，所有在该词典中的词汇及其变形均被接纳，由主持人选中一个单词并给出特定信息，参赛者根据信息回答出主持人选中的单词。很多人为了在拼字游戏中取得成绩，甚至每天花费七八个小时去记忆那些冷门的英语单词，其中甚至包括绝大多数成年都闻所未闻的偏词[1]。其积极意义在于增进处于学语期间的青少年对单词的熟悉程度，帮助其理解单词构造，单就这一点而言，许多英语教育者已经将其运用到了课题上，在温习所学的同时活跃课堂气氛。

广义上讲的拼字游戏包含很多类型，包括广为人知的吊死鬼、拼字数独甚至传统的单词听写。本报告将讨论其中的一个特殊形式，并给出一个优秀的游戏策略及从优化方法角度讨论其数学原理。

**问题叙述**

本报告讨论的拼字游戏规则如下：

（1）进行游戏需要1位主持人与1位参加者。

（2）游戏开始前会公开一系列参考单词，称为词典，主持人选取其中一个单词，称为秘密，并告知参加者单词长度。

（3）参加者从词典中选取一个单词提交给主持人，该单词称为一个猜测，主持人在收到参加者的猜测后需要回答该猜测对秘密的命中程度，即猜测与秘密在几个位置上具有相同的字母。

（4）重复进行步骤（3）十次。

（5）参加者选取的十个猜测中，如果包含秘密，则参加者胜，反之，则主持人胜。

本报告将站在参加者的立场上，希望能够找到一个有尽可能大胜率的游戏策略。

**问题公式化**

本节内容会介绍我们对该问题的建模过程、在本报告中将使用到的数学符号及其含义。首先对于给定的单词长度，令表示词典中包含的长度为的单词数量，则可以将词典建模为的由整数构成的矩阵，即，猜测与秘密被建模为在维整数空间上的向量，即，同时显然的，存在使得，表示词典的第个行向量，存在使得，表示词典的第个行向量。此时，根据定义，问题叙述节的步骤（3）中所涉及到的命中程度可以被建模为猜测与秘密之差的零范数相关的函数，即：

|  |  |  |
| --- | --- | --- |
|  | 。 | (1) |

在实际工程中我们对单词的编码方法是逐字母独立进行的，每个字母被唯一编码为中的整数。但为图叙述简便，在本报告的其他内容中不会针对此实现进行规约。

我们的目标则可以认为是利用尽可能少的猜测求解出秘密，即联立尽可能少的方程，使得关于秘密的方程组

|  |  |  |
| --- | --- | --- |
|  | 。 | (2) |

中表示第个猜测，为其对应的命中程度。

除此之外我们还定义了关于给定词典的函数

|  |  |  |
| --- | --- | --- |
|  |  | (3) |

我们认为上式可以用来在一定程度上衡量词典内单词的数量与单词之间的相似程度，而通常的，基于一个单词数量较少，单词之间相似程度较高的词典进行的拼字游戏会具有更低的难度。

**算法流程与关键难点**

**算法的基础框架及可行性分析**

为了达成上述目的，我们分析了拼字游戏的一般思路，总结了参加者方的思维逻辑，并基于此搭建了我们算法设计的基础框架，如图1所示。我们设计每个猜测会基于当时的词典，并从其中选取，在得到的命中程度后，对中的所有行向量进行测试，所有不满足

|  |  |  |
| --- | --- | --- |
|  |  | (4) |

的行向量被从中删除，这样设计的好处在于可以使得程序能在有限步内停止并输出，我们称满足这个特性的算法是可行的。由于至少时式（4）不被满足（否则即有，输出即可），故在每个迭代过程中词典的规模都得到了缩小且对于所有词典，都成立，故显然程序能在有限步内停止并输出，即算法可行，关键点在于如何设计猜测使得迭代的步数尽可能少。

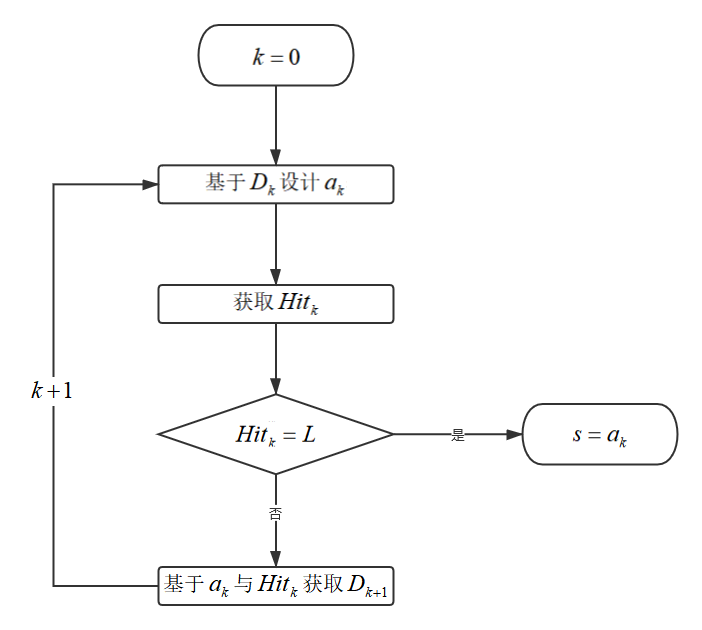


图1. 算法设计的基础框架

**基于词典的问题难度分析及猜测向量的设计**

除了上述内容所述的有限步内停机约束，即可行性约束外，我们还希望经过每次迭代后都能获得一个更容易求解的词典，换句话说每次迭代都将原始问题转化为了一个更易解答的问题，我们在问题公式化部分提出了基于词典衡量问题难度的方法（式（3）），即设计的应该是使得成立的最小值点。从而把设计转化成了一个优化问题，然而由于式（3）中包含零范数，难以优化求解，只能通过的约束枚举尝试。由于式（1）中未知，所以对于给定，的值只能基于期望给出，不妨记该期望为，其具体计算方法如下。

|  |  |  |
| --- | --- | --- |
|  |  | (5) |

其中是对应的命中程度的概率函数，是反映输入向量是否需要从当前词典中删除的门函数，二者的具体计算方法如下：

|  |  |  |
| --- | --- | --- |
|  |  | (6) |

|  |  |  |
| --- | --- | --- |
|  |  | (7) |

需要说明的是，在实际工程中，可以通过预先根据分类并缓存的方法规避式（5）中的门函数，从而提高计算性能。我们设计的算法会在每次设计时遍历，从而找到使得式（5）的最小值点，即。算法的具体伪代码见附录。

**结果与分析**

为简单起见，我们在构造测试集时选择忽略了英语构词规则，设计了一套随机生成单词的程序生成了一系列不同单词长度、不同规模的词典，并利用其检验了我们设计的算法性能。

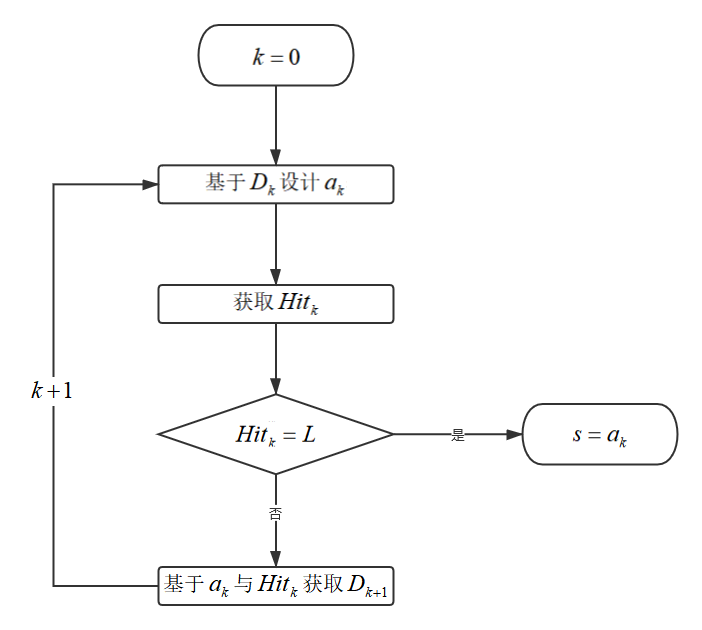


图2. 还没做好的图框架

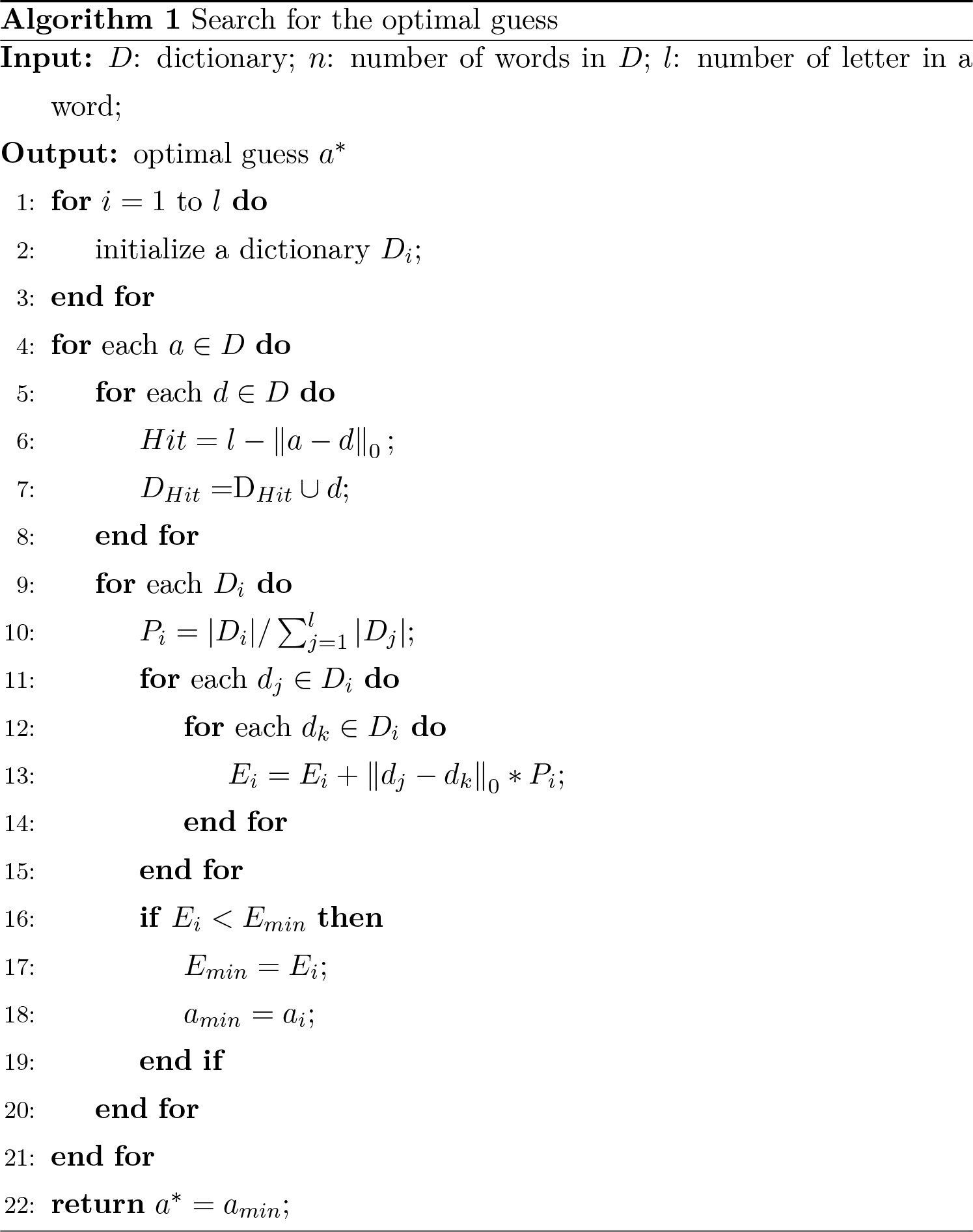
注意到猜测次数并不随字典规模单调增长，我们认为这可能表明了一个优秀的猜测对决策具有的积极作用，更大的字典规模使得我们具有更大的选择范围以确定一个优秀的猜测。

**结论**

**参考文献**

[1] 王盈.拼字游戏备受美国青少年青睐[J].基础教育参考,2007(04):29.

**附录**



设计最优猜测向量的伪代码

除此之外，可以在https://github.com/South-Walker/OptimizationMethod中访问到本文中所涉及的具体代码。