规则碎片的拼接问题

摘要

本文针对两种切割方式以及不同的正反面特点的文件复原问题进行了建模与求解算法设置。首先对只含有纵切的情况，建立了对二值化灰度矩阵匹配度函数的模型，并在求解时进一步转化为优化模型利用贪心算法求解；然后对于横切加纵切的情况，针对中英文字体特点的差异性建立了基于基线匹配的中文拼接模型以及基于四线三格匹配的英文拼接模型，完成了碎纸片复原；最后针对单双面问题，为了提高程序运行效率，我们固定两侧基准碎片减少需要遍历的情况，复原了文件且模型效果较好。

针对问题一：在只考虑纵切的情况下，首先将图像转化为灰度矩阵并进行二值化处理。首先根据页边距特点找出最左侧图片，其次利用最小二乘距离定义了任意两碎片之间的匹配度，并且将此数值定义为碎片之间的权值，将问题简化为问题，最终利用贪心算法将之转化为优化问题求解。由结果得出该模型对中英文均适用，并且无需人工干预。

针对问题二：考虑碎纸机以横切和纵切两种方式切割得到了张碎片，由于黑体文字的规范性且碎纸机为均匀切割，我们考虑将行列排序逐步分开处理。考虑同行文字字体高度一致，所以先将碎片按行分组。由于中英文的差异性我们建立了不同的按行聚类模型，在聚类中文时利用黑体字以及空白间距的基线信息进行分类；同时针对英文四线三格的特点利用累积灰度图确定碎片的文字位置信息进行聚类，然后利用第一问的模型进行行内拼接，最终利用行间匹配算法对行间的碎片进行最终拼接，得到了复原结果。复原过程中需要人工干预的地方出现在部分特征不明显碎片，机器单独将之归为一类，我们利用语义人工干预复原，提高了匹配的成功率。

针对问题三：针对正反面的情况，首先根据页边距确定左右两边的碎片，然后利用贪心算法进行匹配与按行分组，最终利用第二问的模型进行行内匹配和行间匹配得到最终复原结果。其中对复原错误的情况我们进行人工干预，提高了结果的可靠性和准确率。

本文针对文件复原与文字行列特点给出了针对性较强的模型，并且定义了匹配度与基线等因素进行精准复原并给出算法可自动给出碎片编号与复原图，提高了程序执行程度，降低了人工干预率。

关键词：碎片复原、二值化矩阵、、匹配度、聚类、贪心算法

1. 问题重述

破碎文件的拼接复原工作有很大应用，在传统上拼接复原工作需由人工完成，准确率较高，但效率很低。特别是当碎片数量巨大，人工拼接很难在短时间内完成任务。随着计算机技术的发展，人们试图开发碎纸片的自动拼接技术，以提高拼接复原效率。请讨论以下问题：

1. 对于给定的来自同一页印刷文字文件的碎纸机破碎纸片（仅纵切），建立碎纸片拼接复原模型和算法，并针对附件1、附件2给出的中、英文各一页文件的碎片数据进行拼接复原。如果复原过程需要人工干预，请写出干预方式及干预的时间节点。复原结果以图片形式及表格形式表达。
2. 对于碎纸机既纵切又横切的情形，请设计碎纸片拼接复原模型和算法，并针对附件3、附件4给出的中、英文各一页文件的碎片数据进行拼接复原。如果复原过程需要人工干预，请写出干预方式及干预的时间节点。复原结果表达要求同上。

3. 上述所给碎片数据均为单面打印文件，从现实情形出发，还可能有双面打印文件的碎纸片拼接复原问题需要解决。请尝试设计相应的碎纸片拼接复原模型与算法，并就附件5的碎片数据给出拼接复原结果。

1. 问题分析

2.1问题一分析

问题一要求在只考虑纵切的情况下对印刷文件对碎纸片建立复原模型，并给出附件的复原结果。由于颜色在计算机中用之间的数值表示，故将附件所给图片转化为矩阵来表示，由此得到信息灰度矩阵。通过研究每张图片的灰度矩阵来解决问题。图一为一维的复原问题，在只考虑纵切的情况下每张碎片的信息量较大、特征明显，匹配的成功率较高。

首先找出最左端的碎片，该纸片的特点是左侧具有一定的空白（页边距）。然后按照从左往右的顺序依次拼接。由文字特征可以知道若两张碎片相邻，则两张图对应的灰度矩阵边缘像素列灰度值相似度很高，绝大部分对应行的灰度值相等，将这个特点作为衡量任意两个碎片之间匹配度的标准，利用最小二乘法计算出两个碎片的匹配度系数，将此系数赋为这两个碎片之间的权值。此时问题转化为典型的问题，碎纸片为结点，最小二乘值为边的权值，寻找最短路径。本问题中英文字体的差异对结果的影响很小。

考虑到灰度矩阵中代表颜色的数字数值过大且种类较多，在计算处理的时候会存在误差，所以对灰度矩阵进行二值化将之化为只有（黑色像素）和（白色像素）2的矩阵。常用方法有设定全局阈值以及针对特征进行处理，在此考虑到文字特征的重要性、灰色像素其实为文字与空白的边缘的特征，采取针对特征处理，将全白像素视为，灰度以及黑全转化为，即全白才白，有黑就黑。转化后将最小二乘距离计算出解决问题提高算法效率。

2.2问题二的分析

问题二在问题一的基础上加入了横切，碎纸片数量增多为张且碎片上信息量减少，直接拼接误差会很大，我们考虑在第一问模型上进行改进，先按行分组匹配之后再对各行进行拼接。考虑到中文和英文在字体特征和印刷结构的不同，需要定义不同的特征向量去描述中文和英文碎纸片上的信息。在按行分组时，文字在碎片上的位置以及行间距决定分组结果。

对于中文我们对每个碎纸片上文字的字体位置以及行间距空白信息算出基线数值，根据这个数值进行聚类得到按行分组结果。英文因为没有中文字体规整，我们考虑四线三格最中间一格的主体部分，利用累积灰度图验证找到黑色像素值突变点，再利用文字位置等得到基线信息，进行聚类。按行分组结束后利用第一问模型进行行内拼接，之后再把各行拼接起来。

2.3问题三的分析

对于问题三，在问题二的基础上加入了正反面信息，碎纸片的大小形状信息未改变。此时进行匹配情况非常多。需要在第二问的基础上综合考虑正反面匹配的误差，分组方式与拼接方式与第二问基本相同，最后通过语义进行部分人工干预。

综上所述，本问题可以看做对以碎纸片为结点，误差评估函数数值为边的权值的问题寻找最优解。

1. 模型假设

1.假设需要复原的碎片来自同一篇文章且语义通顺。

2.假设文件上文字种类、行间距、文字特征信息一致。

3.假设纸张无黑点等客观影响因素。

4.假设文章两侧空白距离明显。

1. 符号说明

|  |  |
| --- | --- |
|  | 碎片文字文件编号 |
|  | 灰度矩阵 |
|  | 第个碎片文件对应矩阵的第行第列的灰度值 |
|  | 二值化后的灰度矩阵 |
|  | 匹配度 |
|  | 二值化后的灰度矩阵对应灰度值 |
|  | 行高匹配度 |
|  | 碎片集合 |
|  | 问题决策变量 |

1. 模型的建立与求解

5.1 模型准备

5.1.1 图像数据化处理

在进行文字碎片匹配时由于计算机只能处理数字信号，我们将之转化为碎片信息矩阵，问题一中所给的碎片每一个都会被转化为的矩阵。我们通过MATLAB根据像素点的灰度值反应碎片信息，代表不同颜色，“”为黑色，“”为白色。中间各个数值表示由黑到白的过渡色。

在这里每个碎片都可以得到各自的数字矩阵，表示第个碎片所对应的矩阵，表示第个碎片文件对应矩阵第行第列所代表的灰度值。

5.1.2 二值化

由于在定义匹配度以及误差值的时候需要用列边缘像素计算，而边缘像素信息量多且数值较大，我们采取二值化方法对灰度矩阵进行处理，提高算法效率。考虑到此题中文字特征的重要性，为了尽可能保留文字的黑色像素，我们利用如下公式二值化，建立新的灰度矩阵。

 （1）

5.1.3 匹配度的定义

定义任意两个碎片和之间的最小二乘距离称为二者的匹配度：

 （2）

此数值越小，匹配度越高。

5.2 问题一模型的建立及求解

5.2.1 模型的建立

对于碎片的复原问题主要修复方法是像素点匹配法。此题特点性较强，解决问题时主要考虑文字特征以及文字位置。已知原图四周没有文字且假设页边距足够大，根据此特点很容易找出并固定最左侧一列碎片。

之后在对剩余碎片进行匹配时，考虑到图片特征，若两张图片被切割有两种情况：

（1）碎纸机纵切时且在文字上；

（2）正好在两列文字之间的空白部分切割。

针对情况（1）：切在文字上时，可以根据文字笔画的连续性确定碎片是否能拼接，通常情况下左、右碎片黑色像素位置是连续的，像素值相似度很高，计算两个碎片的匹配度。

针对情况（2）：当切在空白部分时，第一问仅考虑纵切，碎片信息较多误差小，针对的矩阵匹配度数值误差影响很小。

根据匹配度的定义我们可以计算得到任意两个碎片之间的匹配度，现在问题本质则变为：已知任意两点之间的权值，寻找一个有序序列使得碎片之间的总匹配度最高，权值之和最小。该问题为典型的求最短路径问题，我们使用贪心算法求解。

如图其中数字节点表示碎片，有向线段长度表示匹配度数值，弧尾为左边碎片，弧头为右碎片。现寻找一条回路使遍历所有节点且不重复达到匹配度最高。

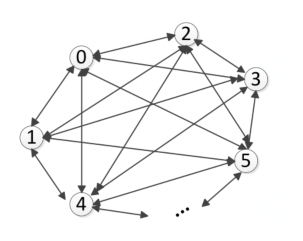


图1 问题示意图

则该问题为以匹配度最高为目标函数建立的寻最优解的问题模型，可表示为：

 （3）

5.2.2 模型的求解

：将所有碎片放入指针存放；

：根据灰度矩阵特点找出最左侧一列，并将之从存放地址中取出；

：依次计算已经固定的最右侧一个碎片与剩余碎片的匹配度，选择匹配度最高的碎片作为当前碎片的右侧碎片，从地址中取出该碎片；

：重复上述步骤直到碎片取完。

该算法根据匹配度求解问题最优解，信息量比较大，对中英文都适用。最终得到附件与附件求解结果如下：

表1 附件1排序后碎片序列表



表2 附件2排序后碎片序列表



由附录还原的文件可知两张图片都具有极高的可读性，所以我们认为复原结果正确。通过结果我们发现贪心算法可以得到全局最优解，原因是匹配度定义准确并且碎片像素信息提供较多，并且对中英文都适用。

5.3 问题二模型的建立与求解

问题二在问题一的问题上加入了横切，碎片数量增多且每个碎片具有的信息量大幅减小，所以我们需要对中英文特征的差异性进一步讨论。此问题是第一问的推广，所以我们考虑将之转化为与第一问相似的问题求解，完整化模型。

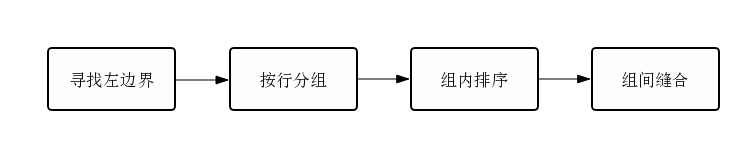


图2 逻辑思路示意图

5.3.1 基于基线匹配的中文拼接模型

5.3.1.1 碎片的按行分组

（1）按行分组的原因

在解决第二问时我们希望整个问题可以通过完整的模型解决，但在此我们采用按行分组而非按列主要原因是此问题是加入了横切，我们应该按横切将碎片归类，且已知行与行之间的文字平行，进行同样横切的碎片文字的行高与字体位置信息是相同的，此特征会极大提高分组效率。

（2）按行分组的标准

由附件我们可以知道，每个碎片都具有三行文字的位置且大小相同，为了找出具有相同横切特征的碎片，我们对每个碎片上具有的文字位置和间距信息进行数据分析，得到每个碎片的一些文字特征信息，按照这类信息进行聚类找出同行碎片。

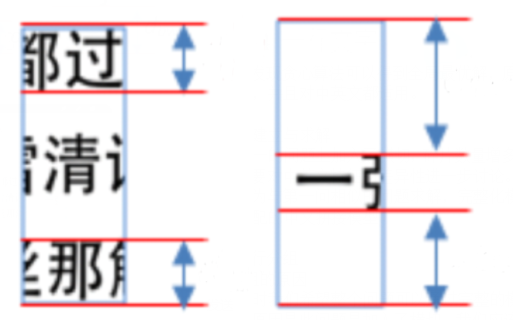


图3 文字参数示意图

如图所示图中含有四个距离，一张碎片文字的行高信息可以由这些信息反映，我们根据这四个距离对每个碎片进行数据分析，由得出的数据进行按行聚类。在计算信息时，为了使特征尽可能准确，我们对每个碎片的灰度矩阵数值进行分析，对矩阵每行进行扫描，遇到有黑色像素即为出现文字，一直到矩阵某一行全为白色像素认为该行文字高度结束。由此确定出每个碎片的文字距离信息，根据此数据信息进行聚类，得到行间分组结果。在扫描行高的时候考虑到系统误差，我们对出现文字的黑色像素矩阵行下方预留行的缓冲带，减小误差。

5.3.1.2 聚类分析

-均值聚类算法是一种划分聚类分割方法。划分的基本原理是：给定一个有个元组或者记录的数据集，分裂法将构造个分组，每一个分组就代表一个聚类，。而这个分组满足下列条件：

（1）每一个分组至少包含一个数据记录；

（2）每一个数据记录属于且仅属于一个分组。

对给定的我们首先给出左首位的一列碎片，通过反复迭代的方法改变分组。工作原理为首先选择个点，每个点初始的代表每个簇的聚类中心，然后计算剩余各个样本到聚类中心的距离，将它赋给最近的簇，接着重新计算这一簇的平均值，不断重复过程直到相邻两次没有明显变化，说明形成的簇已经收敛。具体模型为：

 （4）

利用聚类分析进行按行分组后，会由组内碎片个数等发现明显存在错误，进行人工干预，在给出结果时我们给出人工干预的方式与时间节点。

5.3.1.3 组内碎片匹配

：将按行分组的同组碎片以及聚类不成功未分组碎片存放；

：根据灰度矩阵特点找出最左侧具有全白像素点的碎片，并将之从存放地址中取出；

：依次计算已经固定的最右侧一个碎片与剩余碎片的匹配度，选择匹配度最高的碎片作为当前碎片的右侧碎片，从地址中取出该碎片；

：重复上述步骤直到碎片取完。

在附件中给出按行聚类分组结果，可知横切的十一组分类较准确，分类不明确的碎片被归为其余四类，在人工干预时错误也可以肉眼调整。

5.3.1.4 人工干预

由于碎纸机以横切和纵切方式切割，得到碎片过小，含带矩阵信息过少。在进行最优匹配时若两个碎片较为相似、可能出现排序错误，此时我们加以人工干预。

当行内排序拼接完成后，我们可知文字基线比较相似所以在最优匹配排序时出现乱序现象，但大部分通过语句可读性判断是正确的，针对这种部分片段我们进行人工干预。

5.3.1.5 组间缝合

在第一问纵切的模型上进行改进，利用同样的方法对只进行一种切割的碎片计算匹配度，得到任意两碎片之间的权值后，寻找最上侧存在全白像素行，确定最上侧碎片。此时问题简化为问题，以碎片为结点，匹配度距离值为权值，求解最优解。由于已经进行按行分组之后的组内拼接，所以在组间缝合时碎片像素较多，信息量大，结果较为准确。

5.3.2 基于四线三格基线匹配的英文拼接模型



图4 英文行高示意图

由文字的字形、行分布特征可以知道，中英文文字有明显差别，需要分类讨论。英文在印刷排版有严格的四线三格，根据这个特点我们通过找出黑色像素突变点确定中间一格的文字位置，抹去长笔画影响而寻找主要部分，建立了基于四线三格的英文拼接模型。思路同样为根据行分布信息进行按行聚类，之后与中文解决思路大致相同。

5.3.2.1 基于中文按行分组的模型修正

为了寻找四线三格第二格的主体部分，我们采用累积灰度图方法。由碎片二值化后的灰度矩阵画出文字的累积灰度图，找出数值突变点来确定文字位置。

由于我们在二值化时白色像素值为，并且在此处空白行间距较为容易确定，所以我们对白色像素值进行累积，得到累积灰度图，由图片峰值对文字继续进行分析。

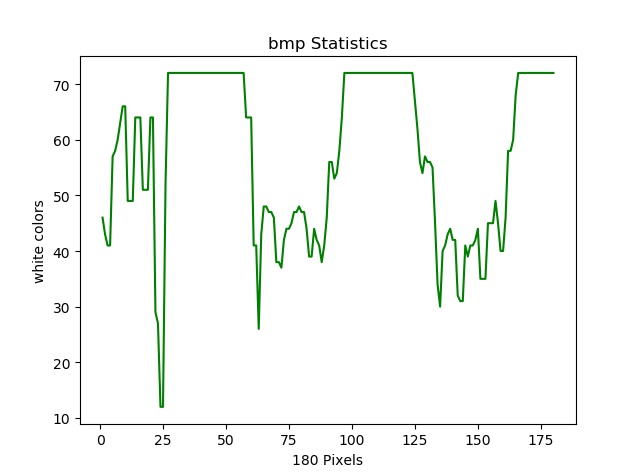










图5 累积灰度示意图

上图为白色像素累计值的图像。按行分组的重要依据是每块碎片文字位置的基准线，因为需要确定每块碎片基准线的位置。由图片基准线是黑白像素点的分水岭，通过黑白像素点个数来确定基准线位置。方法如下：

可知当峰值持续性较高且平稳时，当前灰度矩阵扫描行处于间隔位置，峰值较低为黑色文字部分。通过对比任意碎片之间的累积灰度图峰值位置以及平稳程度我们进行按行聚类。为了进一步准确判断间隔的位置，我们设置阈值来判断灰度累计图是否达到空白部分。

 （5）

5.3.3 模型求解结果

附件求解结果

5.4 问题三模型的建立与求解

对于问题三，在问题二的基础上加入了正反面信息，碎纸片的数据信息均未改变。解决方案基本与第二问相同，但需要在第二问基础上加入针对碎片正反面特点的改进。

5.4.1 模型的建立

（1）寻找最左侧首位碎片

同上述过程相同，我们先找出最左侧的一类碎片，根据假设这类碎片最左侧有一侧空白，我们根据灰度矩阵反应的像素数值信息索引出这类碎片。由文字信息可判断为左侧还是右侧。

（2）行内拼接，组间缝合

在加入正反面情况之后，两个碎片之间理论上存在四种拼接结果，但在固定最左侧基准碎片后，在其余碎片中选择下一个拼接碎片只存在两种情况，如图所示。

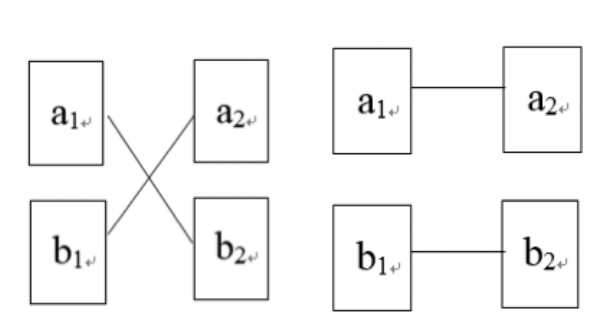


图6 两种碎片匹配示意图

图中为左侧基准图片的正反面。为了将正反面问题简化，我们在固定左侧右侧的基准碎片之后，用剩余的碎片根据第二问思路利用贪心算法进行匹配减少一半的计算量。

思路是利用第二问模型进行固定首位碎片的按行分组。但在此需注意匹配时正反面都需进行，即将所有碎片放入待选区域。若已经确定为正或为反面行的则直接将此碎片对应的背面碎片归为另一面行中排序。在行内拼接完成后进行组间缝合。考虑到碎片信息过小过碎，在过程中加入人工干预。

5.4.2 模型的求解

1. 模型优缺点

6.1模型优点

（1）将一维碎纸片模型转化为排序问题用Tsp模型求解，利用现有模型求得全局最优解，具有拓展性；并且在进行第二问考虑横切与纵切时先以左侧空白部分为搜索特征找出左边界，之后依次进行按行分组，组内拼接，组间缝接这种方法可以很快的将相同特征碎片归类，缩小匹配的次数和数目，极大的减少了运行量提高了效率。

（2）针对中、英文不同的特点寻找各自的基线以及行内文字位置及间距信息，利用文字的这个特征去进行复原匹配。

（3）行内分组时得到了各个碎片的文字基线以及空白间距的位置，利用灰度矩阵进行聚类分析出同行的碎片，针对碎片复原问题误差较小，可以减少人工干预次数。

6.2模型缺点

第二问行内拼接的时候利用两个碎片的灰度信息误差来确定匹配度，若两个碎片信息较为近似则容易出现错误顺序，误差较大导致人工干预次数增多，程序执行效率被降低。

1. 模型的改进与推广

1.对于二维的碎片进行按行聚类时使用的算法是聚类方法，在此我们考虑改进为最大距离聚类法，此方法在聚类时对碎片的文字等信息特征进行放大，可以考虑到原来聚类未细化的数据信息从而使分类结果更精细。

2.在求解问题时原来的求解算法为贪心算法，为了提高算法的适用性和高效性，可以尝试引入模拟退火算法、蚁群算法等智能算法得到全局最优解，得到好一点的结果。

1. 参考文献

[1]卓金武.李必文.魏永生.秦健.《MATLAB在数学建模中的应用》.北京.北京航空航天大学出版社.2014.7页

[2]司守奎.孙兆亮.《数学建模算法与应用》.北京.国防工业出版社.2017.4.7页

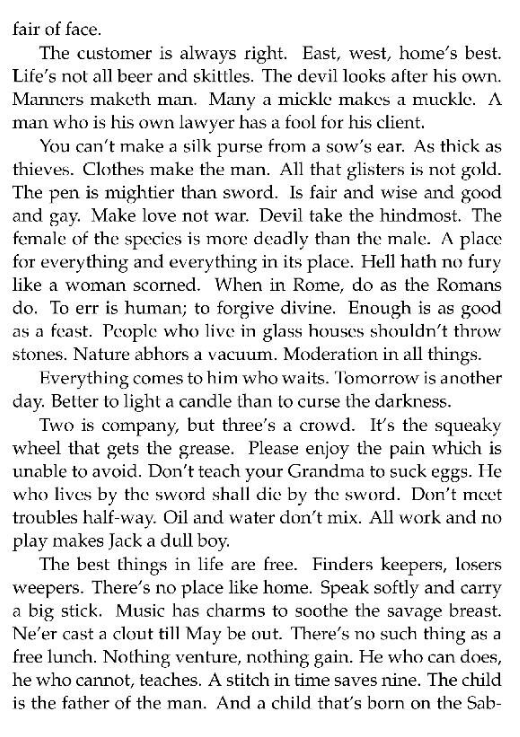
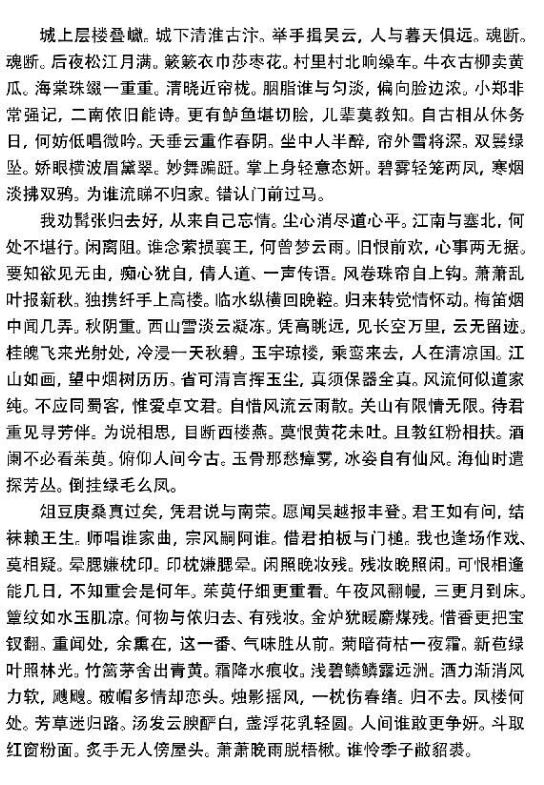
[3]谭忠.碎纸片的拼接复原.<http://special.univs.cn/serv>ice/jianmo/index.shtml

.2019.8.7

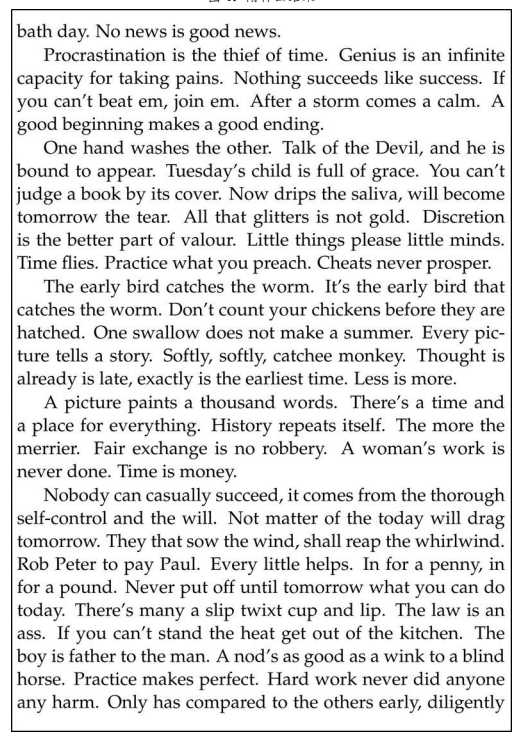
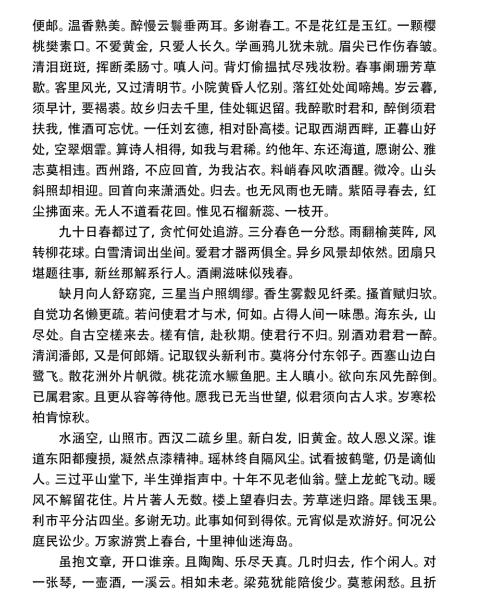
[4]百度百科.贪心算法.https://baike.baidu.com/item/%E8%B4%AA%E5%BF%83%E7%A E%97%E6%B3%95/5411800?fr=aladdin.2019.8.6

附录

问题一原图



问题二原图



问题三原图

