# 第一章 绪论

## 1.1微信小程序开发的背景

## 1.2微信小程序开发的目的和作用

一、成本低

二、有效推广

企业的小程序上线之后，对5公里内的用户进行免费推广，用户离店铺越近，排序就越靠前。而且，用户使用过的微信小程序，会自动的保存在列表中，方便他们下次使用。

从应用生态的角度来看，小程序完全可以和app进行互补，满足用户对应用市场的不同需求。如果一款app不符合市场要求和用户需求，那么他将一定会淘汰，而小程序完成可以弥补这一空缺。对于企业和商家来说，这也是一件利好之事。

三、可直接搜索

而微信又对小程序开放了大量的入口，小程序的流畅度与APP不相上下。另外， HTLM5页面经常会出现加载缓慢、加载超时、卡顿等现象，但微信小程序不会出现这种问题，在运行的流畅性度和用户的体验感上，微信小程序也远超于HTML5页面。

# 

# 第二章 相关技术及其可行性分析

## 2.1前端开发工具

微信web开发者工具是微信官方团队提供的工具，用户能很方便的借助微信web开发者工具编写微信小程序。这个工具集成了小程序的开发、调试、预览和上传功能，并且还带有微信小程序官方的开发文档和开发设计指南。这使得用户能够很方便的对微信小程序进行开发、调试、编辑和发布。

程序调试主要有三大功能区：模拟器、调试工具和小程序操作区。

模拟器能够很方便的模拟微信小程序在客户端真实的表现，一般情况下来说，大部分的API都能够在模拟器上表现出其在客户端真实的表现。

调试工具分为六大功能模块：Wxml、Console、Sources、Network、Appdata、Storage以及WxmlPannel。这些功能模块大都有不同的作用，正常情况下来说，Wxml Pannel主要反应的是微信小程序wxml页面的代码编译情况，通过这个工具，我们可以很方便的获取wxml的编译过程，了解其和传统的html有什么不同。，也很方便我们对wxml文件加以修改。Sources Pannel主要显示脚本的编译情况，而且这个工具还能够主动对代码进行预处理，并主动调用Page页面的require函数。Network Pannel能够监听微信小程序的网络状况，主要是socket包的收发情况。Appdata Pannel主要显示appdata的处理情况，并将其显示在监视器中方便观察。Storage Pannel主要负责数据的处理。而Console Pannel主要方便我们进行代码的调试和错误的显示。

至于小程序操作区，他方便小程序的开发者模拟一些客户端环境。

## 2.2后台开发工具

IDEA全称IntelliJ IDEA，是java编程开发的集成环境，在业界被公认为最好的java开发工具之一，尤其在智能代码助手、代码自动提示、重构、J2EE支持、各类版本工具(git、svn等)、JUnit、CVS整合、代码分析、 创新的GUI设计等方面的功能可以说是超常的。

IDEA所提倡的是智能编码，减少了程序员的工作，他的每个方面的设计都遵循了人体工程学的设计，使程序员能够专注于代码的编写，而不必吧注意力放到其他的地方，比如排版、注释、引用等地方。IDEA的特色功能有以下几点：智能的选取、丰富的导航模式、历史记录功能、对重构的优越支持、编码辅助、 灵活的排版功能、XML的完美支持、动态语法检测、代码检查、列编辑模式、预置模板、完美的自动代码完成、版本控制完美支持、不使用代码的检查、智能代码、正则表达式的查找和替换功能等。

## 2.3开发相关技术介绍

### 2.3.1Java简介

Java是一门面向对象的编程语言，和c++相比，不但吸收了C++的优点，还摒弃了C++中一些过于复杂的概念。因此，Java语言具有功能强大和简单易用两个特征。Java语言作为静态面向对象编程语言的代表，极好地实现了面向对象理论，允许程序员以优雅的思维方式进行复杂的编程。

Java具有简单性、面向对象、分布式、健壮性、安全性、平台独立与可移植性、多线程、动态性等特点。Java可以编写桌面应用程序、Web应用程序、分布式系统和嵌入式系统应用程序等。

Java的编程环境是JDK，常用的编程软件包括Eclipse，NetBeans，IntelliJ IDEA,MyEclipes和EditPlus等。

### 2.3.2JavaScript简介

JavaScript一种轻量级的编程语言，可插入HTML页面，易于学习，是一种动态类型、弱类型、基于原型的语言，内置支持类型。

JavaScript的基本特点如下：

1.不需要对代码进行预编译。

2.可以很方便的和HTML页面进行交互。

3.他既可以直接写入HTML页面，也可以形成自己单独的js文件来方便对结构和行为进行分离。

4.能够跨平台使用，一般情况下，他能够获得绝大多数浏览器的支持，也可以运行于多种平台，包括多种PC端操作系统和移动端操作系统，比如Windows，Linux等。

JavaScript是一种属于网络的脚本语言,常用于各种web开发工作，和HTML、css一起，能够很方便的实现和网络有关的功能。

Javascript作为一种脚本语言，有自身的基本数据类型，表达式和算术运算符及程序的基本程序框架。Javascript既可以通过基本的数据类型来直接处理数据和文字，也可以使用变量来处理比较复杂的数据。

JavaScript的功能包括：插入在动态文本于HTML页面；响应浏览器的基本事件；读写HTML元素并进行数据操作；能够验证数据的正确性，然后再将数据提交到服务器；查询访客的浏览器信息并对不安全的访问发送警告；创建和修改cookies；基于Node.js技术进行服务器端编程。

### 2.3.3HTML简介

HTML全称超文本标记语言，是标准通用标记语言下的一个应用。不同于我吗通常学习的编程语言，脚本语言，HTML 是一种标记语言，是网页制作所必备的。

1

3.平台无关性：超文本标记的语言可以在大多平台上使用， HTML可以忽视后台环境的差异，在不同的平台上几乎可以表现出相同的效果。

### 2.3.4CSS简介

CSS全称层叠样式表，他是一种用来表现HTML或XML等文件样式的计算机语言。CSS不仅可以静态地修饰网页，还可以配合各种脚本语言动态地对网页各元素进行格式化。 CSS 能够对网页中元素位置的排版进行像素级精确控制，支持几乎所有的字体字号样式，拥有对网页对象和模型样式编辑的能力。

CSS的主要特点包括：丰富的样式定义、易于使用和修改、多页面应用、层叠、页面压缩。

# 

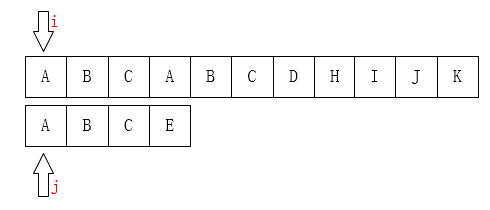
# 第三章 微信联系人分组后台设计

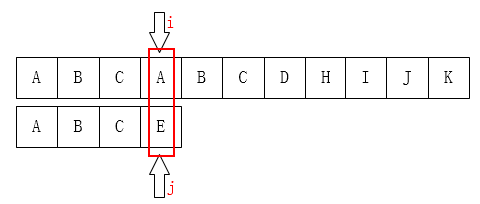
## 3.1对单个微信联系人分组后台设计

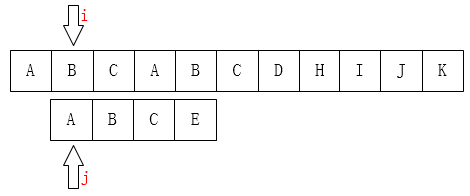
对单个微信联系人进行数据挖掘首先要解决的就是分词提取问题，之后才是具体数据挖掘方法和种类的问题。

### 3.1.1微信联系人聊天记录的分词提取

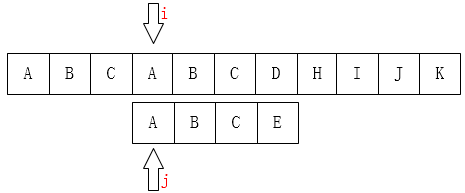
首先，最简单的就是贪心算法，对于主串（T）和子串（P），设主串的长度为M,子串长度为N，从左到右一个一个进行匹配比较，如果这个过程中，发现有字符不匹配，就跳回去，将主串的开始比较位向右移动一位。这样循环下去，一直到发现完全匹配或者说是不能匹配，所需的时间复杂度为O(M\*N)。整个过程可以用下图进行演示：







基于这个思路，我们可以得到以下的算法：

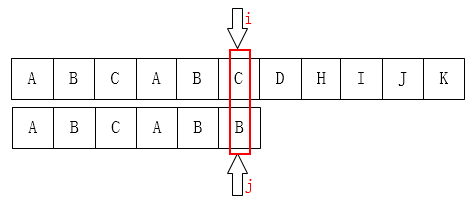


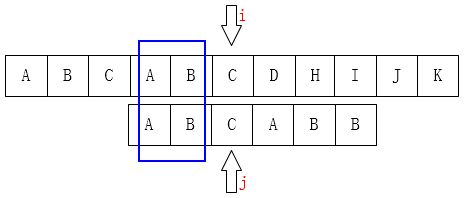
上面的情况还是简单的，可如果遇见主串为”SSSSSSSSSSSSSSS”而子串为”SSSSA”的情况，那么我们也会一直比较到最后一位才能发现他的不匹配情况。然后回溯，显然，这样的效率是非常低下的。

那么kmp算法的核心就是：在部分匹配这个有效信息的情况下，保持主串的i指针不动，而修改子串的j指针，让子串尽量移动到有效的位置，从而降低回溯的花费，进而减小算法的时间复杂度。

所以，整个kmp算法的重点是，当出现子串的某一个字符和主串不匹配的情况时，我们应该把子串的j指针移动到何处，即找到j指针的最优移动解。

如下图所示，我们可以得到这样的结论：当匹配失败时，子串的j指针移动的下一个位置k一定满足这样的条件是，最前面的k个字符和j之前的k个字符是一样的。即P[0——k-1] = P[j-k——j-1]





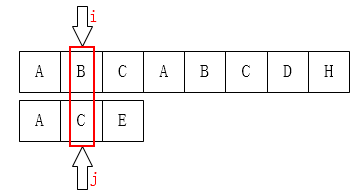
接下来，我们就要想办法求出k值，来确定主串和子串的前多少个字符相同。由于主串T的每个位置和子串P前k个字符相同的k值是不同的，因此我们需要一个数组来记录每个位置的相同个数。

因此，我们可以得到如下的算法：

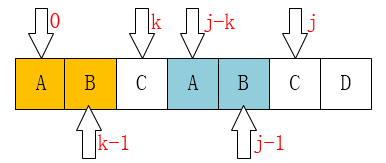
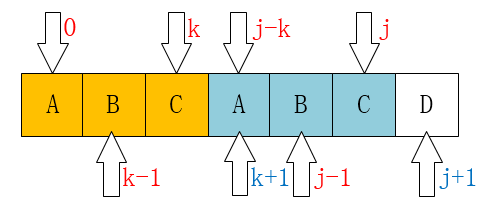
这个算法的关键就是，P[j]和T[i]不相同时，指针j的下一步位置。这存在以下几种情况：

首先，若j为0，即P和T的首字符都不相同的话，直接将i移动到下一个位置，j指针不变。因此对应与next[0] = -1这种情况、

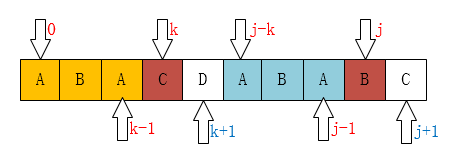
其次，如果j为1时，如下图。j指针一定会前移到0位置，因为他前面也只有0这个位置且T[0]和T[1]不相同。



当j在一个不为0的位置时，就可以观察到以下的情况：

如果P[j] = P[k],那么next[j+1] = next[j]



而如果P[k] != P[j],则k=next[k]。比如上图的例子，我们已经不能找到ABAB这样的后缀串了，但我们仍然可以找到类似于AB、B这样的前缀串，那么如果子串和主串对应指针位置不一样了的话，就会将指针移动到next[k]的位置。

接下来我们就可以手写KMP算法了：

### 3.1.2微信联系人聊天记录的数据挖掘

#### 3.1.2.1微信联系人聊天记录数据挖掘的分词选择

就通常情况下来说，我们可以从微信聊天记录中提取出很多相关信息。首先是聊天记录中某个特定词语出现的次数或者频率。

比如，就一般情况而言，在两人的聊天记录中常常出现“学校”，“作业”，“上课”等词，那么则说明两人的关系有很大的可能是同学关系。

如果在两人的聊天记录中常常出现“办公室”，“家长”等词，那么则说明两人的关系有很大的可能是师生关系。

如果在两人的聊天记录中常常出现“公司”，“项目”，“老板”等词，那么则说明两人的关系有很大的可能是同事关系。

如果在两人的聊天记录中常常出现“回家”或者出现一些和亲属关系有关的词语时，那么则说明两人的关系有很大的可能是亲属关系。

如果在两人的聊天记录中常常出现“玩”，“聚会”等词，那么则说明两人的关系有很大的可能是朋友关系。

除此之外，我们在聊天记录中确定的聊天时间和频率也有助于我们对人物关系加以分析。这其中可以获得的信息包括：

首次交流时间：在两人的首次交流过程中，首条信息的接收者回应首条信息发送者发送的信息所花费的时间。

一般交流时间：在两人的交流过程中，发送者和接受者对互相所发送信息进行响应的总时间除以两人的交流次数。

通信段数：将两位联系人之间的聊天记录以某一个特定的间隔时长为划分，统计两位联系人的交流总段数。

通信时长：按照通信段数的划分方法，出去两人交流的间隔时间，统计两位联系人交流的总时长。

平均通信时长：通信时长除以通信段数。

首次联系时间均值：在两人联系人的每段通信之中，接受者首次回应发送者的平均时长。

一般响应时间均值：在两人联系人的每段通信之中，接受者回应发送者的平均时长。

未响应次数：聊天记录仅为1的通信段数。

以上的聊天记录的时间信息和人物的具体关系还要在实验中加以测试和分析。

#### 3.1.2.2微信联系人聊天记录数据挖掘简介

数据挖掘，就是从大量数据中提取出我们认为的潜在的、有价值的过程。也就是根据预先设定的目标，不断的处理分析大量的数据信息，挖掘数据之间潜在的规律和联系，并逐渐将其模型化、商业化，最终以期能够给予决策者方便的一个过程。数据挖掘是一门交叉学科，他也包含了很多其他学科的内容，运用了其他很多成熟学科所包含的技术和工具，包括数据库技术、统计学、机器学习、模式识别、人工智能和神经网络等等。

数据挖掘也是一种对数据进行深入分析的方法。由于各个商业公司所需求的大量数据流通和数据自动存储分析，因此这些数据的收集也不只在于分析，而且在于商业运作。对数据的分析也不只是在于实验研究的需要，更是为了给商业决策提供必要的信息，从而带来更大的利润。但最关键的问题是，几乎所有企业公司都有庞大的数据，而其中的有效数据却相对而言较少，且难以和普通数据加以区分。因此，数据挖掘也可以说是一种从庞大的数据中高效提取出有效数据，进而带来大量商业利益的技术，就像从矿石中提取出金子一样，数据挖掘技术也因此而得名。

因此，数据挖掘也可以如此描述：对公司和企业的大量数据进行反复的处理和分析，并进一步提取出有关信息，发掘信息数据之间潜在的联系，然后建立相关数据模型并将其商业化，最终获得符合公司和企业要求的潜在信息，并给决策者提供方便。

正常情况下来说，数据挖掘的任务可以有一下集中分类：

分类：分类就是构造一个分类函数或者分类模型，这个分类模型或者分类函数能够将数据分类映射到不同的集合之中。这个分类函数和分类模型既方便使用者对已有数据进行分析处理，又方便使用者对未来的数据进行预测分析。

聚集：聚集就是对已经处理的数据进行分组，将具有相同或者相似属性的数据放入一个集合中。聚集和分类的区别是聚集不需要依靠于预先定义好的类，也就是说不需要训练集。

数据可视化：严格来说，数据可视化不是一个单独的数据挖掘任务。他也被用来支持其他数据挖掘任务。因为数据挖掘的结果还需要使用图表，图形，图片等易于观察的结果来进行展示。

关联规则：关联规则就是寻找数据的相关性，主要是在同一个事物中不同属性的相关性或者说是同一个属性在不同事物中的相关性。比如在一次选课过程中某一位同学所选择的不同的课程的相关性。

序列分析：序列分析也是为了找到数据之间的相关性。他的侧重点在于数据间的关系，时间属性是必不可少的。对于具有循环性，重复性的数据，序列分析是必不可少的。

偏差分析：一般来说，进行初次数据挖掘之后，数据挖掘的结果和实际结果会存在一定的偏差，我们需要观察这种偏差和正常情况的不同之处，来分析这是有意的诈骗行为还是正常的变化行为。如果这是异常行为，那么我们就需要在数据挖掘之中加入预防措施；如果这是正常行为，那么我们就需要更新数据库。

就一般情况下来说，数据挖掘的方法有一下几种分类：机器学习，可视化，统计方法，神经元网络方法，决策树，最近邻技术，而我们主要考虑的就是最近邻技术。

#### 3.1.2.3微信联系人聊天记录数据挖掘方法分析

首先确定测试集的属性，主要包括以下属性：

朋友：“聚会”、“玩”

家人：“回家”、“爸”、“妈”、“儿”、“女”

同事：“工作”、“项目”、“老板”、“经理”

同学/师生：“作业”、“老师”、“同学”、“上课”、“学校”

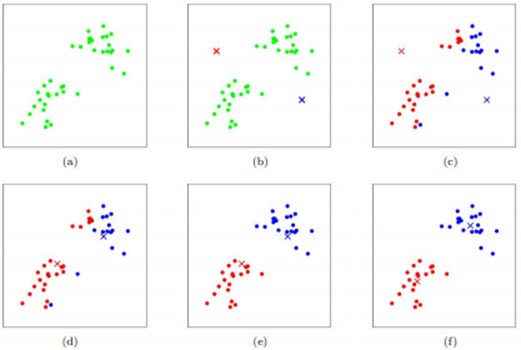
在用KMP算法统计这些词语的出现次数后，每个对应的联系人可以得到这4个方向属性分词的出现次数，这些就作为测试集，而训练集就是我设定的根据微信联系人的备注信息的分组属性，接下来就是具体的数据挖掘算法。

通常来说，最简单的数据挖掘算法就是根据属性的曼哈顿距离或者属性组成的向量角度的差来确定分组，但这种方法太过简单，而且误差和偶然性太大，因此用在这里不合适。

在数据挖掘的一系列算法中，这个项目比较适合的就是KNN和Kmeans算法。

KNN算法是一种分类算法，分类算法要求训练集，然后再将测试级和训练集进行匹配，测试集将按照训练集提供的模板匹配。Kmeans算法是聚类算法，聚类算法和分类算法最大的区别是聚类算法不需要训练集，因为聚类算法是依靠测试集自己的相关性进行分类。

Kmeans算法：

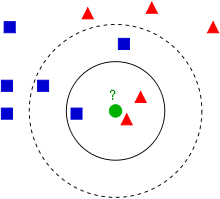


如图所示，数据样本点用圆点表示，每个簇的中心使用叉号表示。最开始数据是杂乱无章的，假设数据可以分为2类，每个点划分到离自己最近的中心点所在的分组。划分完成之后，更新每个簇的中心，既求出到该簇中所有点距离之后最短的点。就这样不断的进行划分更新划分更新，直到簇的中心点不再变动为止。

Kmeans算法还有一些缺陷。比如最开始很难确定k的值即分组个数，在本实验中也许我们所设定的k值也不会是最合适的。另一个方面，在初始中心点的确认上，也会有一些不确定的地方，因为不同的初始中心点也会得到不同的结果，相对而言，尽量较远的出书中心点所得到的结果较为准确。就本实验而言，我们也很难确定最好的初始点在哪。此外，针对本实验的结果，难以划分也是一个缺陷。因为在分为4个簇得到结果之后，很有可能出现一种情况：某个簇的测试集2个或者多个属性的出现次数和频率都很高，这会使得实验结果很难判断测试集的分组。

而针对Kmeans算法的改进可以用Kmeans++实现。Kmeans++算法的基本思想是尽可能的使初始的聚类中心的点相聚较远，这样就会避免初始聚类中心点太过接近而造成的误差。然而，这个算法还是不能解决k值的确定问题和分组结果的划分问题。

KNN算法：



如图所示，对于测试集中的某个样本，如果他在k个最相似的样本中大多属于某个类别，则样本也属于这个类别，这就是KNN算法的大概用法。

KNN的算法构造过程如下，图中的其他数据集都是训练集，一类是蓝色的正方形，一类是红色的三角形，那个绿色的圆形是我们待分类的数据。如果K为3，则离样本最近的3个训练集数据中，有2个属于红色三角形，那么这个样本就应该属于红色三角形簇。同样的情况，如果K为5，那么由于其中有3个蓝色正方形训练集，则样本应该属于蓝色正方形簇。就本质而言，KNN是一种数据统计方法，没有明显的前期训练过程。

在本实验中，我们选用KNN算法来对微信聊天记录进行数据挖掘。首先，根据微信聊天人的备注，将确定4种最基本的人物分类作为训练集。之后，把剩余的微信聊天人当做测试集，每个微信聊天人进行分析时，所取的k值为四种人物分组的最小组人数，该样本分组之后，将会被加入该组作为训练集而存在。如果数据挖掘过程中，某一次最近k个训练集中包含2个或者多个分组人数为最多，那么将k不停的增加，直到能够对样本进行分组为止。

## 3.2对微信群组聊天记录进行数据挖掘

微信群组聊天记录的数据挖掘主要是使用图论的相关技术。在对微信群组进行数据挖掘时，很难根据聊天记录确定两人之间的人物关系，然而我们可以用构造图的方法来将人物之间的关系确定下来。而确定群组成员之间的群组图论关系就需要使用邻接表来存储。

### 3.2.1邻接表和邻接矩阵

邻接表和邻接矩阵是图最主要的2种存储结构。

邻接矩阵这种存储方式就是用两个数组来存储数据，其中一个一维数组用来存储所有的顶点信息，一个二维数组用来存储两点相连，也就是边的信息。无向图的邻接矩阵，是一个对称矩阵，从这个矩阵中我们可以很容易获得以下信息，根据二维数组的某个点是否为0判断两人之间是否有交流；观察某一行的数据来判断群组中的某个成员在群组中的发言频率和次数；观察某一行的数据来确定某位成员和其他成员交流的紧密程度。邻接矩阵最大的缺点就是记录的数据太岁或者说是对于顶点太多、边数较少的情况，邻接矩阵会造成很大的浪费。

图的另一种存储方法就是邻接表，他将数组和链表结合在一起，很好的避免了邻接矩阵的缺点。邻接表使用一个一维数组或者单链表来存储所有的顶点信息，相比较而言，数组存储的顶点信息读取更为方便，通常情况下，我们也会使用数组来存储顶点信息。某个顶点的所有邻接表将和这个顶点使用线性表的方式相连接，由于邻接点个数不定，所以一般使用单链表进行存储。

如果边具有权值，那么单链表中需要加入权值这个数据。

邻接表和邻接矩阵如下图：

