# 安个家推荐系统设计

### 数据部

### 2016.01.11

# 一、产生背景

1、随着公司业务的发展，成交量的增多，对用户的了解愈发重要。在大数据的时代下，我们希望通过对用户数据的探索，发觉用户深层次的需求，从而提升用户购房体验。

2、在互联网+的时代，房源产生到最终交易的时间越来越短，数据的实时性越来越重要，

我们希望通过对用户以及房源实时性数据的分析，进一步提升用户购房体验

# 二、系统功能

系统由两部分组成：

### 1、画像

* 用户画像：

用户画像是对用户基本信息、动态信息、需求模型的描述，可以精准的刻画出一个用户的需求。

* 房源画像:

房源画像是对房源基本信息、点击、带看量等动态信息的描述。

* 经纪人画像：

经纪人画像是对经纪人的基本信息、业绩、服务等的精准刻画。

### 2、推荐

基于公司业务结合业界常用的一些推荐算法，在掌握用户需求的前提下，将用户最喜欢的放在推送给他。

# 三、需求模型

### 1、抽象：

* 用户的需求是多样的，有时连用户自己也不知道自己想买什么样的房子。

例如，某用户的需求：

1、大概买在浦东金桥，张江也会考虑

2、想买三室的，大点的两室也行

3、价格不要超过250万

4、楼层不要太低

5、最好靠地铁的

. . .

* 我们将用户的需求抽象成一个个标签，例如上面的某用户的需求可以抽象成如下标签：

1、板块

2、户型

3、价格

4、楼层

5、地铁

. . .

* 每个标签都对应着用户对这个标签的需求，这需求可能是多个，每个需求对应一个权重，表示用户对此需求的强烈程度。例如对如上用户的需求可以描述为：

1、金桥:90，张江:30

2、三室:80，两室:20

3、200-250:100

4、大于三层:80

5、近地铁:80

. . .

### 2、模型更新

* 用户需求模型的刻画是一个由大概到精准的过程，随着用户每一次的浏览、搜索、筛选、收藏、提交需求单等行为，需求模型在实时更新（即更新用户需求标签里的权重值）。

例如：

1. 用户浏览xx小区两室200万的某套房源，则会将此房源的某些标签加成到用户对应需求标签的权重里。

如：xx小区+10、两室+20、200-250+30等

1. 用户提交需求单，则将此需求加成到用户对应需求标签的权重里
2. 用户筛选某些房屋属性，则将此房屋属性加成到用户对应需求标签的权重里

. . .

# 四、推荐逻辑

有了用户需求模型，再加上现有房源属性，就可以将现有房源里，用户最喜欢的放在推荐给他。

### 1、模糊匹配：

* 根据用户需求模型中的主要标签中用户最喜欢的房屋属性，先模糊匹配出一些房源， 主要是为了避免为所有房源打分。

例如：

1. 选择用户最喜欢的前两个板块
2. 选择用户最喜欢的前两个小区
3. 选择用户最喜欢的那个户型
4. 选择用户最喜欢的那个价格区间

. . .

### 2、打分：

对模糊匹配到的所有房源打分。

* 具体打分逻辑：

观察此套房源的属性中有没有用户需求的属性，如果有，则将用户对此属性需求的权重加成给此套房源的分数。

例如：

1. 如果房源是在金桥，用户对金桥的需求是90
2. 如果房源是xx小区，用户对xx小区的需求是40
3. 如果房源是两室的，用户对两室的需求是80
4. 如果房源是230万，用户对200-250万的需求是70

. . .

则，此套房源得分：

Score=90+40+80+70+...

### 3、结果：

选出得分最高的前10套、100套房源。

# 五、系统实现

系统采用大数据常用软件框架，如kafka、spark、hbase等

### 1、系统架构：



### 2、容错性：

* 系统的每一部分都具有高可用性：

1、数据采集程序如果宕掉，此程序可以记录数据已经采集到的断点，程序再次启动时可以继续从此断点继续采集

2、kafka本身具有分布式特性

3、spark本身也有分布式特性，就算streaming程序宕掉，因为数据是缓存在kafka里，数据没有丢失，streaming程序重启后继续运行

4、hbase本身具有高可用性

### 3、可扩展性：

* 系统每一部分都可以扩展：

1、数据采集程序可以是多个，运行并行将数据写入kafka

2、kafka本身具有可扩展性

3、spark也是分布式框架

4、hbase底层基于hdfs，也有很强的扩展性

### 4、数据流：

