

点赞逻辑分析

关键点

1.因为我们需要一组二维向量，系统自带的button的tag是不够用的，所以，通过自定义的button来获取到当前button所在的cell的indexPath.row

2.cell的重用问题，因为cell不会无限的增加，只会创建固定个数，然后重用，以此来节约内存

3.通过 数组的个数 = indexPath.row 来确定，你点击了第几行的button，再由 button 自带的tag，找到相匹配的button，解决了确定button位置，以及cell重用问题

初始化数组

```
NSMutableArray *dataArray;
```

将0装入数组中

```
/**
 * 控制点赞的，0 -> 为点赞
 * 1 -> 已点赞
 */
dataArray=[[NSMutableArray alloc] init];
for (int i=0; i<20; i++) {
    [dataArray addObject:@"0"];
}
```

创建，点赞的按钮

```
/**
 * 通过for循环创建三个按钮 => 点赞，评论，转发
 */
for (int i=0; i<3; i++) {
    myBtn *goodSendCommentBtn =
    [[myBtn
    alloc] initWithFrame:CGRectMake(i*w/3.1+5, h*3/5-30, w/3.1, 25)];

    goodSendCommentBtn.backgroundColor =
    [UIColor orangeColor];

    goodSendCommentBtn.tag =
    333+i;
    goodSendCommentBtn.index =
    indexPath.row ;;

    [cell
    addSubview:goodSendCommentBtn];

    /**
     * index & tag 共同组成一个二维向量，来确定 button 准确的位置
     * 先由 myBtn 中 index 的属性 确定 按钮所在的 cell 的行数
     * 然后，通过 myBtn 相应的 tag 找到 分别找到 点赞，评论，转发 这三个按钮，添加与之对应的点击事件
     */
    myBtn *goodbtn = [cell viewWithTag:333];
```

通过cell找到button

```
goodbtn.index=indexPath.row;
NSString *str =[dataArray
objectAtIndex:indexPath.row];
[goodbtn setTitle:str
 forState:normal];
[goodbtn addTarget:self
 action:@selector(goodbtnClick:)
 forControlEvents:UIControlEventTouchUpInside];
```

为button设置点击方法 以此来实现点赞的效果

```
- (void)goodbtnClick:(myBtn *)btn{
    [dataArray
    replaceObjectAtIndex:btn.index
    withObject:@"1"];
    [hotTableView reloadData];
}
```