

基于淘宝用户行为数据的漏斗分析 与用户增长策略研究

分析师: 刘学贤

日期: 2025 年 9 月 3 日星期三

联系方式: +86 13322909058 / leoliu.tech@qq.com

1 项目概述

本项目旨在通过分析阿里巴巴天池公开的百万级淘宝用户行为数据，运用 SQL 进行数据处理与提取，并结合 Excel 进行深度可视化分析。项目核心在于构建用户行为转化漏斗，量化关键节点的转化效率，最终为平台的精细化运营和用户增长策略提供数据驱动的建议。

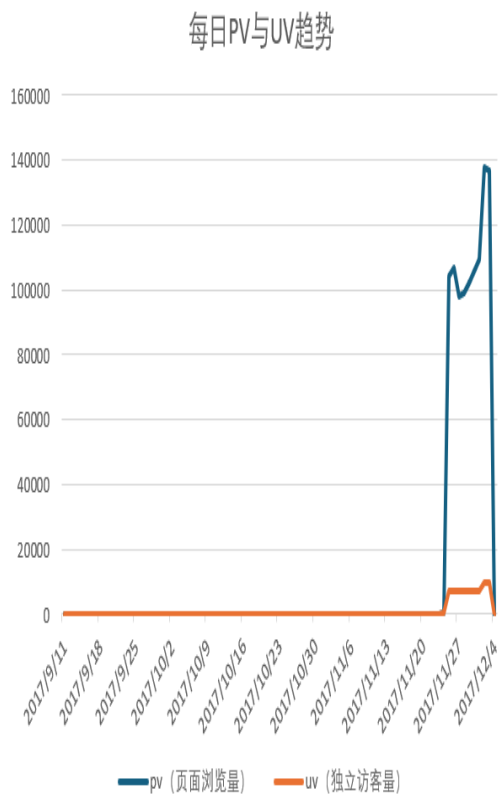
2 技术与工具

- **数据处理：**MySQL 8.0
- **数据查询：**SQL (主要使用聚合函数、WITH 子句等)
- **数据可视化：**Microsoft Excel (主要使用图表、数据透视表)

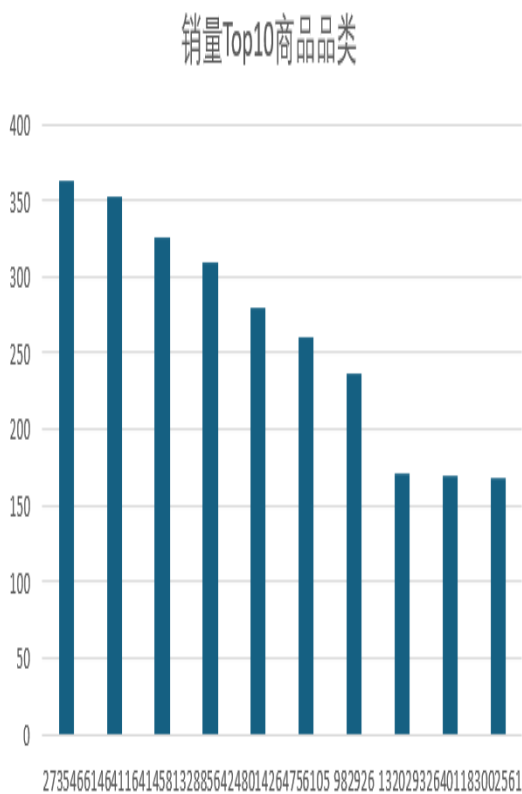
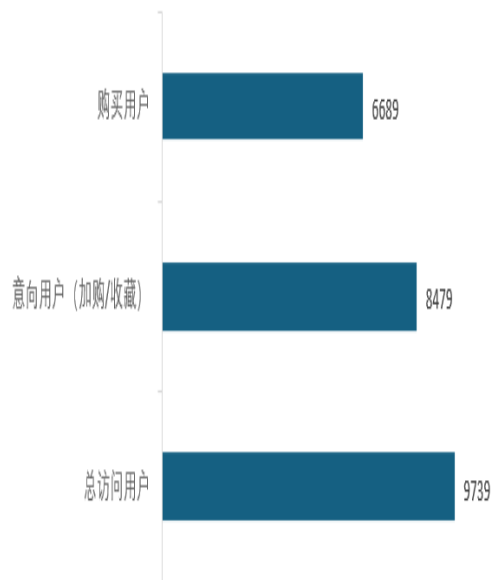
3 分析框架

- **宏观流量分析：**洞察平台整体的每日活跃用户趋势。
- **核心转化分析：**构建从访问到购买的用户转化漏斗，定位流失环节。
- **商品品类分析：**识别销量最高的头部商品品类，分析销售结构。

4 核心数据看板



用户行为转化漏斗 (访问到购买的用户转化情况)



关键指标

总独立访客 (UV)

9,739

最终购买用户

6,689

整体转化率

68.7%

(计算方式: 6689/9739)

5 分析洞察与商业建议

5.1 核心发现

- **发现一：平台流量高度依赖大促活动。**

从“每日核心流量指标”图可以看出，平台整体流量在 11 月 25 日至 12 月 3 日期间出现爆发式增长，PV 与 UV 均达到平日的数十倍。这表明平台的日常流量较低，增长主要由短期营销活动驱动。

- **发现二：用户“临门一脚”的转化是当前增长的关键瓶颈。**

从“用户行为转化漏斗”图可见，用户从“总访问”到“产生意向(加购/收藏)”的转化率高达 87.1%，但从“产生意向”到“最终购买”的转化率下降至 78.9%，这是用户流失最主要的环节。

- **发现三：商品品类销售呈现明显的头部效应。**

从“销量 Top 10 商品品类”图可见，头部品类的销量远高于其他品类，说明平台的销售额主要依赖少数热门品类。

5.2 商业建议

- **针对“流量依赖大促”：**

建议在平日推出小型的、主题性的营销活动（如“美妆周”），并通过内容运营（如直播、短视频）来提升用户的日活，实现更健康的流量结构。

- **针对“临门一脚转化瓶颈”：**

建议将策略重点放在购物车环节的优化上。例如，实施“购物车召回”策略，对加入购物车但未付款的用户进行定向提醒；或发放小额“购物车专属优惠券”，刺激用户尽快完成支付。

- **针对“品类头部效应”：**

建议在巩固头部品类优势的同时，通过交叉推荐、关联销售等方式，利用头部品类的巨大流量为其他有潜力的次级品类引流，以发掘新的增长点。

6 附录：关键技术实现过程

6.1 数据清洗与预处理

在数据导入初期，遇到了原始 CSV 无表头导致列名错误的问题（SQL 报错 1054：未知列）。通过 `SELECT * FROM raw_behavior LIMIT 1` 查询，确认了真实列名并使用反引号进行引用，成功解决了数据转换问题。以下是最终清洗数据的核心 SQL 代码片段：

```
INSERT INTO user_behavior (user_id, item_id, category_id, behavior_type, event_time, event_date)
SELECT
    CAST(`1` AS SIGNED),
    CAST(`2268318` AS SIGNED),
    CAST(`2520377` AS SIGNED),
    `pv`,
    FROM_UNIXTIME(CAST(`1511544070` AS SIGNED)),
    DATE(FROM_UNIXTIME(CAST(`1511544070` AS SIGNED)))
FROM
    raw_behavior;
```

6.2 核心漏斗构建

为构建用户行为转化漏斗，本项目采用了 WITH 子句（CTE）来分步计算各环节的独立用户数（UV），使代码逻辑更清晰、可读性更高。

```
WITH FunnelData AS (
    -- 第1步：计算总的独立访客数
    SELECT '1_total_uv' AS step, COUNT(DISTINCT user_id) AS user_count
    FROM user_behavior

    UNION ALL

    -- 第2步：计算有过“加购”或“收藏”行为的独立用户数
    SELECT '2_cart_or_fav_uv' AS step, COUNT(DISTINCT user_id) AS user_count
    FROM user_behavior
    WHERE behavior_type IN ('cart', 'fav')

    UNION ALL

    -- 第3步：计算有过“购买”行为的独立用户数
    SELECT '3_buy_uv' AS step, COUNT(DISTINCT user_id) AS user_count
    FROM user_behavior
    WHERE behavior_type = 'buy'
)
-- 从临时表中查询最终结果
SELECT
    step,
    user_count
FROM
    FunnelData
ORDER BY
    step;
```