codecademy

Attribution Queries

Learn SQL from Scratch

Felipe de Paula

April 2019

Table of Contents

1. Get familiar with CoolTShirts
2. What is the user journey?
3. Optimize the campaign budget

1. Get familiar with CoolTShirts

1.1

How many campaigns and sources does CoolTShirts use?

- 8 Campaigns, using *utm_campaign* column, which lists each specific ad or email blast. Renamed for ease of reading;
- 6 Sources, using *utm_source*, which lists which site sent the user. Renamed for ease of reading.

```
SELECT COUNT(DISTINCT utm_campaign)
AS 'Distinct Campaigns'
FROM page_visits;
```

```
SELECT COUNT(DISTINCT utm_source)
AS 'Distinct Sources'
FROM page_visits;
```

Which source is used for each campaign?

```
SELECT
DISTINCT utm_campaign AS 'Distinct Campaigns',
utm_source AS 'Sources'
FROM page_visits;
```

Two of the sources relate to multiple campaigns: *email* and *google*. List as shown on the left.

Distinct Campaigns	Sources
getting-to-know-cool-tshirts	nytimes
weekly-newsletter	email
ten-crazy-cool-tshirts-facts	buzzfeed
retargetting-campaign	email
retargetting-ad	facebook
interview-with-cool-tshirts-founder	medium
paid-search	google
cool-tshirts-search	google

1.2

What pages are on the CoolTShirts website?

4 pages:

1 - landing_page

2 - shopping_cart

3 - checkout

4 - purchase

```
SELECT
DISTINCT page_name
FROM page_visits;
```

2. What is the user journey

2.1

How many first touches is each campaign responsible for?

Selected minimal timestamp for each ID, made it a temporary table *first_touch* and joined with *page_visits* on equal values, bringing up the campaigns associated with each first touch. Made it a second temporary table *tf2* in order to finally extract from it only the column *ft2.utm_campaigns* and the number of rows *COUNT (*)*, grouped by the former.

```
WITH first_touch AS (
 SELECT user_id,
        MIN(timestamp) as first_touch_at
 FROM page_visits
 GROUP BY user_id),
ft2 AS (
  SELECT ft.user_id,
         ft.first_touch_at,
         pv.utm_campaign
  FROM first_touch AS ft
  JOIN page_visits AS pv
    ON ft.user_id = pv.user_id
    AND ft.first_touch_at = pv.timestamp
)
SELECT ft2.utm_campaign AS 'Campaigns',
COUNT(*) AS 'Number of first touches'
FROM ft2
GROUP BY 1
ORDER BY 2 DESC;
```

Campaigns	Number of first touches
interview-with-cool-tshirts-founder	622
getting-to-know-cool-tshirts	612
ten-crazy-cool-tshirts-facts	576
cool-tshirts-search	169

2.2

How many last touches is each campaign responsible for?

Same as before, but this time around using *MAX(timestamp)* for the last touch and changing nomenclature where necessary.

```
WITH last_touch AS (
 SELECT user_id,
        MAX(timestamp) as last_touch_at
 FROM page_visits
 GROUP BY user_id),
lt2 AS (
  SELECT lt.user_id,
         lt.last_touch_at,
         pv.utm_campaign
  FROM last_touch AS lt
  JOIN page_visits AS pv
    ON lt.user_id = pv.user_id
    AND lt.last_touch_at = pv.timestamp
)
SELECT lt2.utm_campaign AS 'Campaigns',
COUNT(*) AS 'Number of last touches'
FROM lt2
GROUP BY 1
ORDER BY 2 DESC;
```

Campaigns	Number of last touches
weekly-newsletter	447
retargetting-ad	443
retargetting-campaign	245
getting-to-know-cool-tshirts	232
ten-crazy-cool-tshirts-facts	190
interview-with-cool-tshirts-founder	184
paid-search	178
cool-tshirts-search	60

2.3

How many visitors make a purchase?

Selecting the number of rows for each distinct user that reached the purchase page.

```
SELECT COUNT(DISTINCT user_id) AS 'Number of buyers'
FROM page_visits
WHERE page_name = '4 - purchase';
```

Number of buyers
361

2.4

How many last touches *on the purchase page* is each campaign responsible for?

Simply adding a *Where* clause to filter results whose last touch happened to correspond to the purchase page.

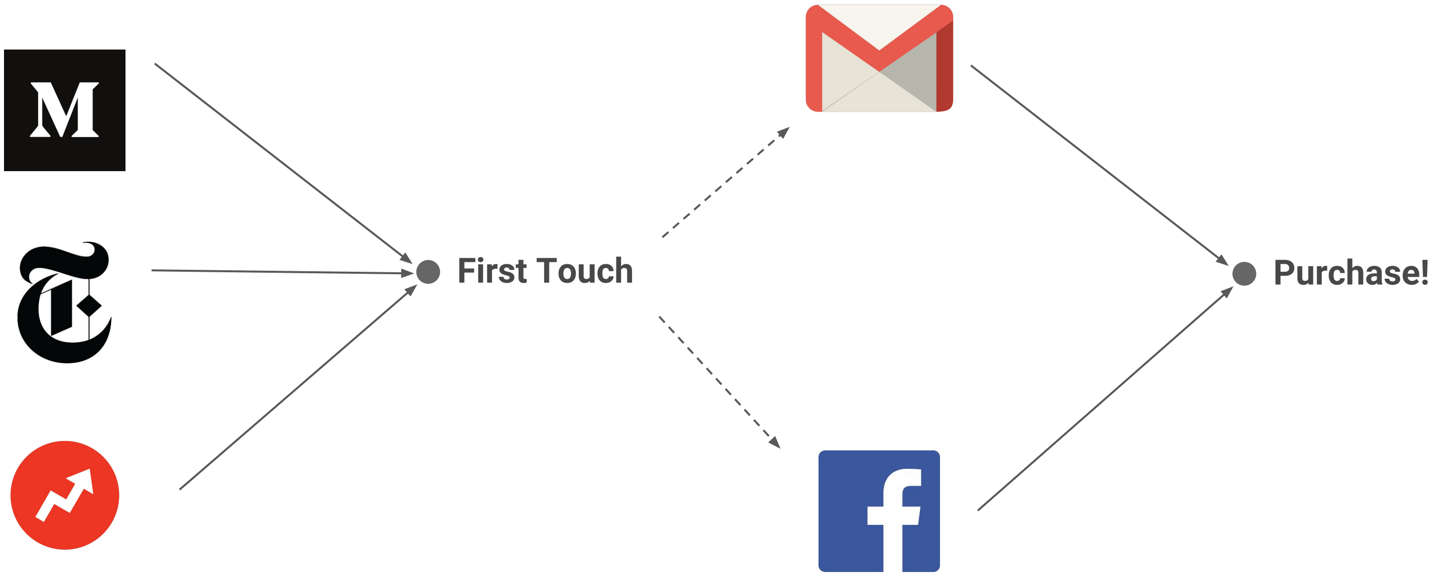
```
WITH last_touch AS (
 SELECT user_id,
        MAX(timestamp) as last_touch_at
 FROM page_visits
 WHERE page_name = '4 - purchase'
 GROUP BY user_id),
lt2 AS (
  SELECT lt.user_id,
         lt.last_touch_at,
         pv.utm_campaign
  FROM last_touch AS lt
  JOIN page_visits AS pv
    ON lt.user_id = pv.user_id
    AND lt.last_touch_at = pv.timestamp
)
SELECT lt2.utm_campaign AS 'Campaigns',
COUNT(*) AS 'Last touches on purchase page'
FROM lt2
GROUP BY 1
ORDER BY 2 DESC;
```

Campaigns	Last touches on purchase page
weekly-newsletter	115
retargetting-ad	113
retargetting-campaign	54
paid-search	52
getting-to-know-cool-tshirts	9
ten-crazy-cool-tshirts-facts	9
interview-with-cool-tshirts-founder	7
cool-tshirts-search	2

2.5

What is the typical user journey?

Users will mostly visit CoolTShirts for the first time coming from campaigns respective to Medium, The New York Times and Buzzfeed. Even though these sources will also account for a considerable amount of last touches, almost none of them results in actual purchase. Effective buyers will be attracted through campaigns run on Facebook and email.



3. Optimize the campaign budget

3.1

CoolTShirts can re-invest in 5 campaigns. Which they should pick and why?

Based on the data, the company should focus on reinforcing first-time visits from the following campaigns:

- interview-with-cool-tshirts-founder
- getting-to-know-cool-tshirts
- ten-crazy-cool-tshirts-facts

These are the ones which already attract the most visitors and get them acquainted with the company and website. Having a positive first impression is critical, and even though users probably won't purchase anything in this first visit, they are more likely to revisit the site when exposed to different campaigns.

For purchase visits, CoolTShirts should re-invest on the following:

- weekly-newsletter
- retargetting-ad

These are the ones which get the majority of user to revisit the website and confirm a purchase, and thus extremely crucial.

codecademy

Attribution Queries

Learn SQL from Scratch

Felipe de Paula

April 2019