Search Engine Marketing cheat sheet

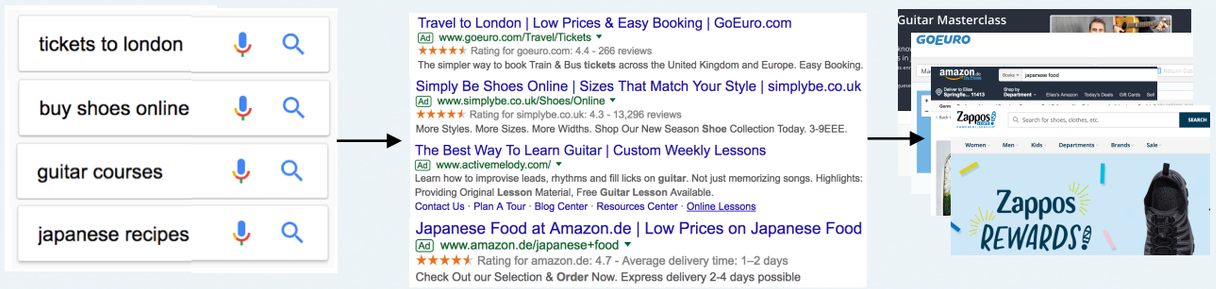
Feedback: @eliasdabbas
Code: GitHub

Main campaign elements

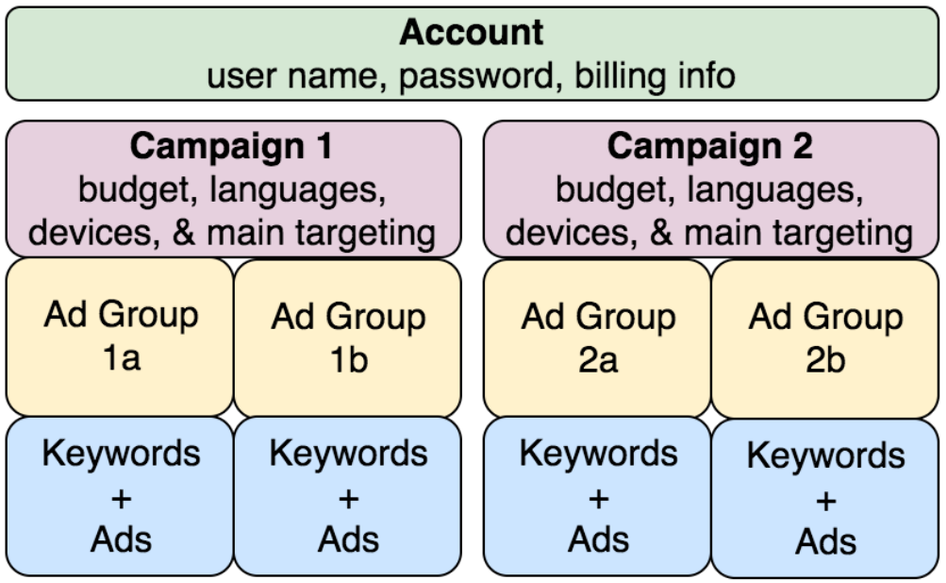
Keywords
users' intentions / desires

Ads
your promise to users

Landing pages
promise fulfillment



Account Structure



Writing effective ads

Element	Explanation	Examples
Product	What it is you are trying to sell, should be clear and prominent	100" TV Screen Leather Shoes
Benefit	The emotional / human benefit, the actual reason why people are motivated to do things	Feel your best by the Summer Achieve a healthy lifestyle
Feature	How your product / service achieves that for them	12 week fitness program Only certified trainers
Call to action	What you want the user to do	Subscribe to our newsletter Buy now Get in touch

Keyword match types

Match Type	How to write	Matches:	Example keyword	Keyword matches:
Broad	keyword example	synonyms and related terms	buy shoes	shop shoes shoes shopping
Broad match modifier	+keyword +example	stemmings and close variants	+buy +shoes	buying shoes buy shoe
Phrase	"keyword example"	exact sequence with anything before and/or after	"buy shoes"	buy shoes online how to buy shoes
Exact	[keyword example]	close variants, order doesn't matter, function words ignored	[buy shoes]	buy shoes buying shoes shoes buy
Negative	-keyword example	do NOT trigger my ads if they include this keyword	-cheap	buy shoes ~~cheap shoes~~

advertools basic functions

```
pip install advertools
>>> import advertools as adv
```

Generate keywords for a campaign:

```
>>> products = ['cars']
>>> words = ['buy', 'price']
>>> adv.kw_generate(products, words).head()
       Campaign Ad Group          Keyword Criterion Type     Labels
0   SEM_Campaign     Cars            cars buy          Exact        Buy
1   SEM_Campaign     Cars            cars buy         Phrase        Buy
2   SEM_Campaign     Cars          +cars +buy          Broad        Buy
3   SEM_Campaign     Cars          cars price          Exact      Price
4   SEM_Campaign     Cars          cars price         Phrase      Price
```

Create ad templates that dynamically fill the blanks with product names and handle long text with a fallback option

```
>>> adv.ad_create('Get the latest {}', ['iphone', 'galaxy', 'sony xperia
xz2 preimum'], fallback='phone', max_len=30)
['Get the latest iphone', 'Get the latest galaxy', 'Get the latest
phone']
```

Convert a long product name into a properly split ad:

```
>>> adv.ad_from_string('Divano Roma Furniture Modern Tufted Linen
Splitback Recliner Sleeper Futon Sofa (Dark Grey)',
slots=(30, 30, 80, 15, 15))
['Divano Roma Furniture Modern', 'Tufted Linen Splitback Recliner',
'Sleeper Futon Sofa (Dark Grey)', '', '']
```