Recommendation System for Advertising Campaigns

Elinor Avraham | Nofet Damri | Lidor Rubi

Requirements Document

Chapter 1 - Introduction

1.1 The Problem Domain

Managing an online campaign can be complex - there is a lot of data to track, such as the number of people who entered the ad versus the number of people who viewed it, and the benefit of advertising on each of the ad-networking platforms, such as Taboola and Outbrain.

OptimusQ helps online advertisers manage their advertising campaigns and streamlines their work with these companies.

As part of the campaign there is the issue of choosing the image that will appear on the front of the ad. Given a landing page provided by the advertiser, there is a difficulty in analyzing the page and understanding the relevant meta data in order to select an image and text for the ad so that it will be optimal in terms of entry to the advertised page.

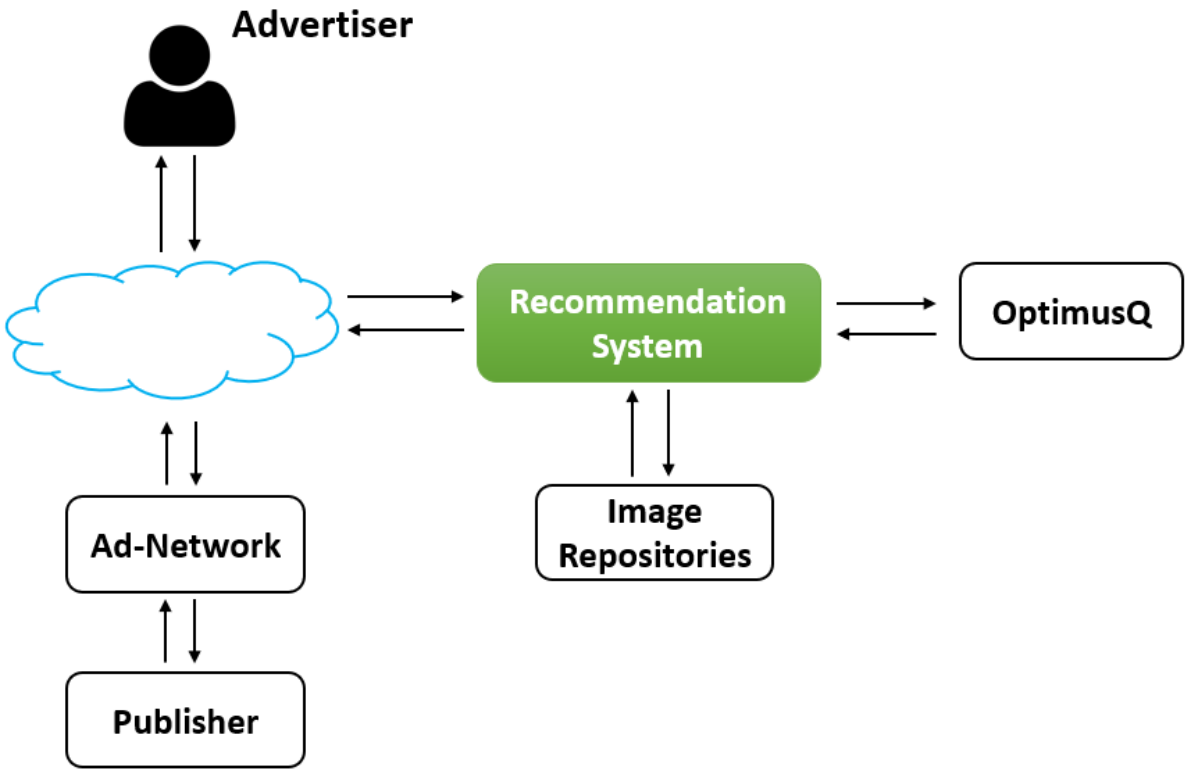
1.2 context

The OptimusQ system enables the management and tracking of advertising campaigns, as opposed to the platforms of the ad-network companies.

The advertiser will enter OptimusQ the landing page URL. Our system will interface with that of OptimusQ, get the page from it, analyze it by web-scraping - on the variety of elements in

it, such as HTML, text, image, video, etc. - and classify the page accordingly. In addition, the system will transfer to OptimusQ the relevant metadata about the page.

The system will also interface with image repositories to select the most appropriate images for the advertising content automatically, and will offer them to the user. The system will transfer the user selection to OptimusQ, which will monitor the campaign.



1.3 Vision

The main goal of the project is to create a system that allows you to analyze landing pages, extract from them the information it needs and classify the landing page in the appropriate category. Next, the system will calculate by machine learning which image, among the image repositories, is most suitable for the benefit of maximizing the landing page entries. The system will check with OptimusQ what the most successful image is, save the data in the cloud and follow up.

1.4 Stakeholders

- Advertisers
- OptimusQ system

1.5 The context in terms of software

The system will scan a landing page received from an advertiser, extract from it the information required to understand the field in which the advertising content deals and store the information in the cloud. The system will connect to image repositories, retrieve images from them and process them. The system will recommend to the advertiser the most suitable images. The system will transfer the data to the OptimusQ system to track the selection.

Schedule for current semester

Week Number	Tasks	Participants	Comments
Week 2 26.10.2021	General project meeting,Work rules	Course staff	According to the syllabus
	Project start-up meeting with a professional guide - Yakir, ML expert and responsible for our group from OptimusQ	Team members + professional guide	
Week 3 31.10.21	Finish Writing an ARD draft + choosing technology		
	Development of a simple cloud application in Azure, which will do basic web-scraping		For study purposes
week 4 7.11.21	Arrival with a draft ARD requirements document and technological selection + personal meeting with a seminar guide		According to the syllabus
Week 7 28.11.21	Submission of requirements document + Continued progress	Group meeting with the seminar guide	According to the syllabus
week 10 23-24.12.21	Presentation to end the marathon + Prototype work The software should be in the cloud or deployed on a server beyond the PC + Personal meetings with the seminar guide	marathon (Joint meeting of all seminar groups Including industry personnel)	According to the syllabus
Week 13 12.01.21	Submission of an ADD document + demo of the system to the customer + proof of Concept of overcoming risks + personal meeting with the seminar guide		According to the syllabus

System Iterations

1. The system will scan a landing page received from an advertiser, extract from it the information required to understand the topic in which the advertising content deals.
2. The system will store the information in the cloud.
3. The system will connect to image repositories.
4. The system will retrieve images from the repositories and process them.
5. The system will recommend to the advertiser the most suitable images.
6. The system will transfer the data to the OptimusQ system to track the selection.

Technologies we are required to learn

- MongoDB
- Kubernetes
- Docker
- Cloud - Azure
- Web-Scraping with Python + HTML basics
- image processing
- ML
- Interface with API - image repositories, for example shutterstock
- Working with the OptimusQ system

Chapter 2 - Use Cases

use case 1: Extract useful information from a landing page

Players: User - Advertiser.

Trigger: User clicks on "scrape a landing page".

preconditions: The user has entered a landing page into the system.

Termination Terms: The systemIntroduced to the user all the essential information regarding the landing page.

normal flow:

1. The user enters a landing page into the system.
2. The system generates the required data from the page.
3. The system stores the data in the cloud.
4. The system shows the user all the data that has been extracted from the page.

use case 2: Recommendation of an image for advertising content

Players: User - Advertiser.

Trigger: The user clicks on "recommend a picture".

preconditions:

a. The user has entered a landing page into the system.
b. The system extracts necessary information from the landing page.

Termination conditions: The system Introduced to the user all the recommended images

normal flow:

1. The system will connect to image repositories.
2. The system will retrieve images from the repositories and process them.
3. The system will recommend to the advertiser the most suitable images.
4. The system will transfer the data to the OptimusQ system to track the selection.

Chapter 3 - Functional Requirements

Id	Requirement	Priority (MH/NTH)	Risk (L/M/H)
1	The system will allow the user to enter a landing page	MH	L
2	The system will allow to scan a landing page according to text, images and other rules that will be defined	MH	L
3	User will be allowed to set rules for scanning a landing page	MH	L
4	The system will store data in Azure cloud	MH	L
5	The system will be connected to image repositories' APIs	MH	L
6	The system will be able to retrieve images from image repositories	MH	L
7	The system will be able to process images taken from image repositories	MH	L
8	The system will be connected to OptimusQ system for receiving and sending information	MH	L
9	The system will recommend the most appropriate image to the user, based on data generated from the landing page	MH	L
10	The system will pass on to OptimusQ the user selection information, for tracking purposes	MH	L

Chapter 4 - Non-Functional Requirements

Id	Requirement	Priority (MH/NTH)	Risk (L/M/H)
1	Capacity and availability: The system must support an unlimited number of users simultaneously	MH	L
2	The system will allow tracking of actions and faults	NTH	L
3	The system will be written in Python	MH	L
4	The system can save and retrieve data from MongoDB	MH	L
5	The system will be in the Azure cloud	MH	L
6	In order to demonstrate the system, a UI interface must be built	NTH	L
7	The system can web-scrape a landing page without limiting the number of words on the page	MH	L
8	The system will complete scraping a landing page scan in less than 5 seconds	MH	L
9	The system will deal gracefully with errors and will display relevant details to the user	MH	L
10	The system will be able to work with landing pages written in English only	MH	L
11	The system should be simple to use, even for inexperienced users	NTH	L
12	The system must be available 24/7, except during maintenance times	MH	L
13	The system must work on Windows operating system	MH	L

Chapter 5 - Risk Assessment & Plan for the proof of concept

In the initial stage, we will learn web scraping with Python, we will get to know relevant HTML commands and we will do experiments on simple HTML pages, imitating landing pages.

After we understand the method of web scraping, we will learn the principles of working with MongoDB-based databases, in order to plan the system implementation. We will also get to know Kubernetes technology and in particular Docker. We will learn about Azure and working with it and read the documentation of communicating with the API of image repositories.

Next, we will start implementing the first iteration of the system, which will include extracting information from a landing page - from pages containing simple elements, such as text only, to pages containing images, to scanning videos.

Later, we will study the OptimusQ system, read its doc and understand how our system should communicate with it. Then, we will implement in our system a module for image processing and ML capabilities, in favor of producing better recommendations for the user.

Plan for the proof of concept:

1. We will make several landing pages, with different content and complexity.
2. The system will contain a predefined configuration file, which will contain the landing pages in for the POC (also can be empty, i.e without landing pages).
3. The system should then scrape each landing page and extract relevant metadata, depending on the rules that were set for it.
4. The system will present the user photos recommendation for each landing page.

Also, we will conduct a user study, in which we are going to let publishers try out our system. Then, we will test its recommendations via OptimusQ tracking ability.

In the next stages:

1. The system will transfer relevant data to OptimusQ for tracking.
2. To test the system, we will compare the system recommendation with other alternatives, and see if in at least half of the cases, the recommendation received the highest Click-Through Rate (CTR).

Appendices

Glossary

Click-Through Rate (CTR) - the ratio of users who click on a specific link to the number of total users who view the advertisement.

OptimusQ - A system that allows tracking of advertising campaigns, and in particular the Click-Through Rate (CTR) of selected images for advertising content.

Web Scraping - data scraping used for extracting data from websites. The web scraping software may directly access the World Wide Web using the Hypertext Transfer Protocol or a web browser.

Landing Page - single web page that appears in response to clicking on a search engine optimized search result, marketing promotion or an online advertisement.

Advertiser User - An advertiser who has a landing page and is interested in suiting a relevant picture for the campaign, that will result in high Click-Through Rate (CTR).

Advertising Network - a company that connects advertisers to websites that want to host advertisements. The key function of an ad network is an aggregation of ad supply from publishers and matching it with advertiser's demand. For example: Taboola, Outbrain.

Publisher - a website that presents advertisements.

configuration files - files used to configure the parameters and initial settings for the system.