

Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»
Кафедра математики факультету інформатики

Курсова робота
за спеціальністю „Прикладна математика” 113

Дослідження моделі торгів в Apple Search Ads

Керівник курсової роботи

Дрінь Світлана Сергіївна

Старший викладач, кандидат фізико-
математичних наук.

(підпис)

Виконала студентка 3-го курсу
спеціальності «Прикладна математика»

Журавльова Анастасія Дмитрівна

«14» червня 2022 р.

Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»
Кафедра інформатики факультету інформатики

ЗАТВЕРДЖУЮ
Зав.кафедри математики,
проф., д.ф.-м.н.
_____ Б. В. Олійник
(підпис)
«__»_____2022 р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ
на курсову роботу

студенту Журавльовій Анастасії Дмитрівні факультету інформатики 3 курсу

ТЕМА Дослідження моделі торгів в Apple Search Ads

Зміст ТЧ до курсової роботи:

Індивідуальне завдання

Вступ

1 Загальна структура Apple Search Ads

2 Ключові слова

3 Модель торгів Apple Search Ads

4 Оптимізаційні моделі ROAS та ROI

5 Хід роботи

Висновки

Список літератури

Додатки

Дата видачі „11” жовтня 2021 р. Керівник _____

(підпис)

Завдання отримав _____

(підпис)

Тема: Дослідження моделі торгів в Apple Search Ads

Календарний план виконання роботи:

№ п/п	Назва етапу курсової роботи	Термін виконання етапу	Примітка
1.	Отримання завдання на курсову роботу.	11.10.2021	
2.	Обговорення деталей теми на розробка плану виконання	15.12.2021	
3.	Огляд технічної літератури за темою роботи та опрацювання матеріалів	11.02.2022	
4.	Написання теоретичної частини роботи	08.04.2022	
5.	Розробка практичної частини та фіналізація практичної	31.05.2022	
6.	Аналіз отриманих результатів, обговорення остаточної доповіді з керівником	01.06.2022	
7.	Здача курсової роботи	06.06.2022	

Студент: Журавльова А. Д.

Керівник: Дрінь С.С.

“11” жовтня 2021

ЗМІСТ

Перелік прийнятих скорочень.....	6
Вступ.....	7
РОЗДІЛ 1: Загальна структура Apple Search Ads.....	8
1.1 Загальні зведення про ASA та чому обирати саме її:	8
1.2 Основні стратегії в використанні ASA:	9
1.3 Загальна структура. Як влаштована сама система:.....	9
1.4 Як влаштований рекламний кабінет?	10
1.5 Розходження даних:	11
РОЗДІЛ 2 Ключові слова	12
2.1 Ключові слова (KW) та як правильно їх використовувати:.....	12
2.1.1 Поділ KW за мотивацією аудиторії:.....	12
2.1.2 Типи ключових слів:	12
2.2 Ідекс популярності ключового слова	14
РОЗДІЛ 3: Модель торгів в ASA:	16
РОЗДІЛ 4: Оптимізаційні моделі ROI та ROAS:.....	17
4.1 Найважливіші показники оптимальної рекламної кампанії	17
4.1.1 Ціна за дію (CPA)	18
4.1.2 Рентабельність витрат на рекламу (ROAS) і рентабельність інвестицій (ROI).....	18
4.1.3 Ціна за клік (CPT).....	19
4.1.4 Обсяг конверсій (CV).....	20
4.1.5 Коефіцієнт конверсії (CR)	21
4.2 Оптимізація кожного з показників на прикладах	21
4.2.1 Ціна за дію (CPA)	21
4.2.2 Рентабельність витрат на рекламу (ROAS) і рентабельність інвестицій (ROI).....	23
4.2.3 Ціна за клік (CPT).....	25
4.2.4 Обсяг конверсій (CV).....	26

РОЗДІЛ 5: Хід роботи	27
Висновок.....	36
Список використаної літератури:	37
Додатки.....	38

Перелік прийнятих скорочень

ASA	– Apple Search Ads;
VS	– в порівнянні з;
KW	– ключові слова, keyword;
CV	– conversion volume, обсяг конверсій;
CPA	– cost per action, ціна за дію;
CPT	– cost per tap, ціна за клік;
ROAS	– return on ad spend, рентабельність інвестицій на рекламу;
ROI	– return on investments, рентабельність інвестицій;
CR	– conversion rate, рівень конверсій;
MMP	– Mobile Measurement Partner, допоміжний інструмент для трекінгу реклами;
LAT	– limit ad traffic, обмеження на рекламу;
IDFA	– ідентифікатор рекламодавця;
API	– Application programming interface;
ARPU	– average revenue per user, середні витрати на користувача;
LVT	– lifetime value, ціна за період підписки користувача;
LT	– lifetime, період підписки користувача;
ACoS	– відсоток бюджету, що був отриманий від продажу товару, який був витрачений на рекламу;

Вступ

Для кожного сучасного бізнесу, в яких є свої застосунки, постає проблема із залучення більшого кола зацікавлених людей та споживачів. Один з методів для популяризації серед населення – це рекламна кампанія, зокрема в інтернеті. На сьогоднішній день сфера digital маркетингу нам пропонує всілякі ресурси та інструменти для запуску рекламних кампаній. Деякі працюють за прямими домовленостями між клієнтом і платформою розміщення. Інші ж – на основі аукціону другого біда, наприклад, Google Ads, Business Facebook, Apple Search ads. Як саме працює ця модель торгів розглянемо на прикладі моделі торгів в рекламній системі Apple Search Ads (ASA) та чим вона краща за інші аналоги на ринку. Отже, **мета даної роботи** – дослідження моделі торгів в Apple Search Ads. **Бізнес задача роботи:** максимізація обсягів закупки користувачів при ROI-позитивній моделі, тобто рентабельність бюджету вище 0.

Курсова робота складається з 4 розділів. В першому розділі розглянемо загальну структуру системи, які її принципи роботи, її переваги та недоліки. В наступному заглибимось в тему ключових слів, які існують типи, як правильно ними керувати і оптимізувати. Третій розділ полягає в розгляді моделі торгів та як вона функціонує. Остання частина – найважливіша: розглядається оптимізація основних інструментів в рекламній кампанії.

РОЗДІЛ 1: Загальна структура Apple Search Ads

1.1 Загальні зведення про ASA та чому обирати саме її:

Apple Search Ads вперше вийшов на ринок в 2016 році, але й досі є новим та не повністю зрозумілим інструментом. Чому ж треба обирати ASA для розміщення реклами вашого застосунку?

- Високий рівень конверсії – Conversion Rate (CR). Для ASA це близько 50% для банерної реклами VS Google Ads (1.35% - 3.40%). Це через те, що в ASA максимально зацікавлена аудиторія. 65% загальних завантажень застосунків в App Store надходить із запиту в пошуку, до того ж більше 50% шукають певний застосунок.
- Преміум розміщення – лише на першому місці на сторінці результатів пошуку. Якщо поєднати 65% завантажень, які надходять із пошуку із преміальним місцем розташування, то в результаті ми отримаємо найефективніше розміщення реклами мобільного додатка (рис. 1.1).

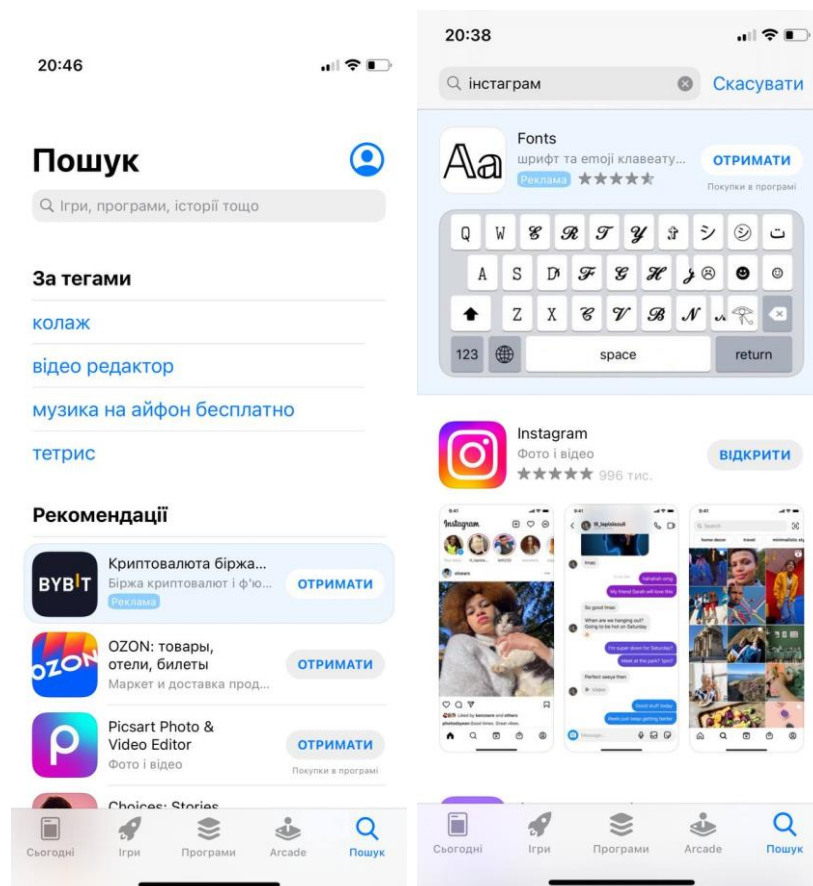


Рисунок 1.1 – Скріншот розміщення реклами в App Store

- Високе органічне охоплення. Коли оголошення ASA відображаються за певним ключовим словом, ваш звичайний список додатків автоматично відображатиметься на 8–10 позиціях у результатах пошуку, навіть якщо у вашому списку ключових слів немає даного ключового слова
- Легко почати. Apple не вимагає мінімальних зборів. Крім того, існує ряд попередньо створених інструментів для швидкого запуску (Apple Search Ads Basic, Search Match в ASA Advanced тощо). Хоча ви можете почати швидко, оптимізація та масштабування ваших кампаній – це те, де Apple Search Ads стає справді складним.

1.2 Основні стратегії в використанні ASA:

1. Захист бренду – підтримка застосунку та захист від конкурентів, які ставлять ставки на ваші ключові слова
2. Конкурентні торги – цілеспрямовані ставки на ключові слова ваших конкурентів, щоб залучити частку потенційних користувачів ваших конкурентів.
3. Пошук – пошук нових користувачів для вашого застосунку, які ще не зробили вибір на користь певного бренду та вибирають серед різних варіантів.
4. Дослідження – використання Apple Search Ads як джерела пошуку найбільш ефективних ключових слів для вашої кампанії в Apple Search Ads.

1.3 Загальна структура. Як влаштована сама система:

На сьогодні Google Ads, Business Facebook та інші великі рекламні системи пропонують дуже широкий і різноманітний вибір форматів для розміщення реклами. Через це розробникам складно обирати правильні формати для кориснішої реклами, тому і втрачають потенційне охоплення та бюджет. ASA пропонує лише два формати для реклами в App Store:

- а) На вкладці пошуку
- б) Після запиту на першій позиції

Рекламне оголошення складається з назви застосунку і розробника, іконки, 100-470 символів опису, рейтинг застосунку та скріншоти з самого застосунку (рис. 1.2)

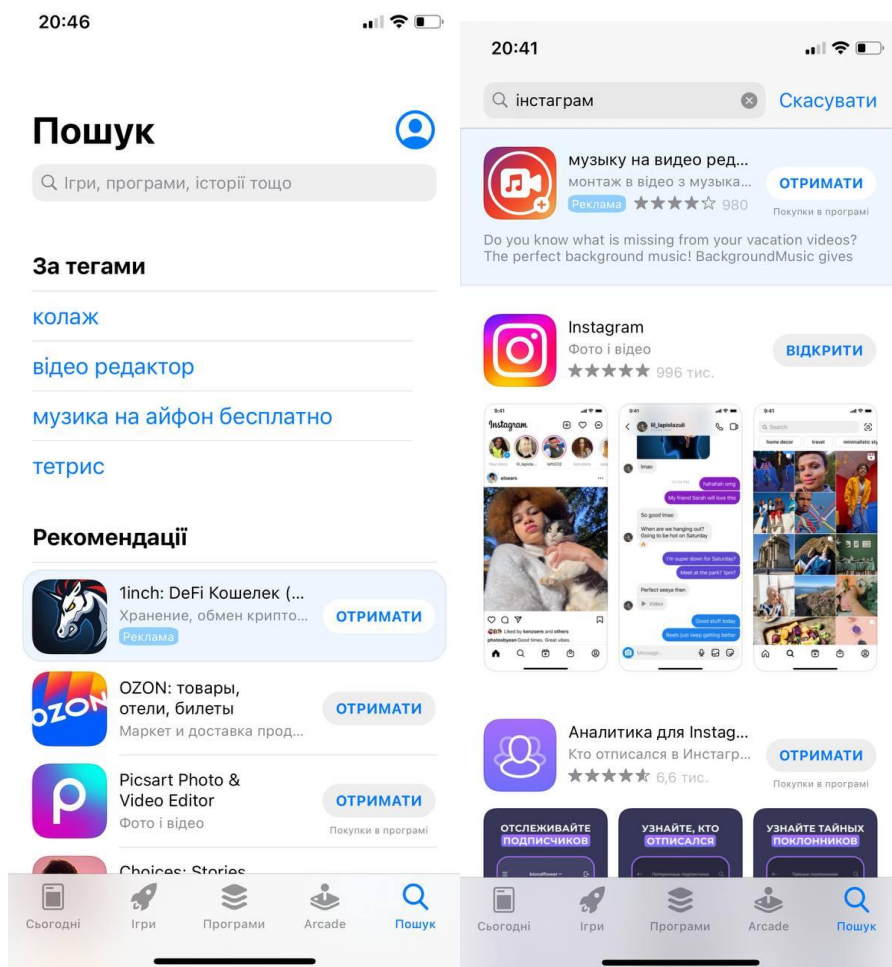


Рисунок 1.2 – Скріншот розміщення реклами в App Store

1.4 Як влаштований рекламний кабінет?

Створення бренду – початковий етап потребує більше часу, більше інформації, пошуку груп відповідних користувачів і вкладення коштів для дослідження трафіку і купівлі першої позиції на вкладці пошуку. Для кожної рекламної кампанії є групи оголошень, тобто де зберігаються схожі ключові слова, на які таргетується ваш застосунок. Крім того, саме на цьому рівні відбувається керування ставками, які і є основним об'єктом дослідження далі. Також можна керувати вашою аудиторією, наприклад, задавати вік, стать, країна, інтереси аудиторії, що переглядає вашу рекламу. І останній найширший рівень – рівень ключових слів. Ключові слова – це

слова, за якими буде показуватись реклама вашого застосунку під час пошуку. Створюються цілі списки слів, за яким відображується оголошення.

1.5 Розходження даних:

У Apple Search Ads розбіжності даних із MMP коливаються від 20 до 70%, що означає, що до 70% доходу від каналу можна приховати, тобто віднести до органічного трафіку або інших маркетингових каналів.

Чому це відбувається? Ось кілька причин:

1. Повторне завантаження. Повторне завантаження – це користувачі, які завантажують програму кілька разів. Залежно від своїх налаштувань, MMP можуть розпізнати, що це перезавантаження наявним користувачем, і віднести його до медіа-джерела, з якого цей користувач спочатку прийшов. Деякі категорії додатків, наприклад, знайомства, мають понад 60% повторних завантажень.
2. Обмеження відстеження реклами (LAT) – файли cookies. LAT допомагає користувачам Apple запобігти відстеженню їхніх пристроїв медіа-джерелами та MMP. Коли LAT увімкнено, ідентифікатор для рекламодавців (IDFA), пов'язаний з пристроєм, замінюється на нулі. Як наслідок, MMP не може збирати дані користувачів, включаючи встановлення додатків, тому Apple Search Ads не відображається в статистиці вашого мобільного трекера, і ви не отримуєте повного уявлення про атрибуцію. У середньому розбіжності на основі LAT становлять 15-20%.
3. Збій API, відкриті затримки, налаштування атрибуції. Ці причини розбіжностей зазвичай становлять 5-10%. Щоб глибше зануритися в причини невідповідності даних, прочитайте цю статтю.

РОЗДІЛ 2: Ключові слова

2.1 Ключові слова (KW) та як правильно їх використовувати:

2.1.1 Поділ KW за мотивацією аудиторії:

1. **Пізнавальний** («кулінарні ігри», «фітнес» тощо). Основна мета пізнавального пошуку це знайти новий застосунок

2. **Конкретна зацікавленість** («Amazon» , «Instagram» тощо). Мета даного пошуку – завантаження конкретного застосунку, в якому зацікавлений користувач.

2.1.2 Типи ключових слів:

1. **Широка відповідність (Broad match)** – тип відповідності за замовчанням для ключового слова. Він відображатиме ваше оголошення за пошуковим запитом з близькими словами, як-от синоніми, рідкісні орфографічні помилки, та пов'язаний пошук і фрази, які містять цей термін частково або повністю.

Для того, щоб правильно налаштувати кампанію треба поділити ключові слова на 4 групи оголошень:

- Ключові слова, пов'язані з функціями застосунку
- Ключові слова, пов'язані з конкурентами бренду
- Ключові слова, пов'язані з самим брендом
- Синонімічні конструкції

Крім того треба перевірити результати рекламної кампанії. Якщо є слова, які мають мало трафіку або нерелевантні, треба перемістити їх в мінус слова. Слова з найбільшим попитом навпаки треба перенести в групу точної відповідності (Exact match)

2. **Точна відповідність (Exact match)** – для точних ключових слів ваше оголошення буде показано користувачеві лише тоді, коли його пошуковий запит точно відповідає вашому ключовому слову, а також для найпопулярніших орфографічних помилок і множин відмінювання слів. Для брендovаних слів краще використовувати тип ключових слів, тобто тих, що містять назву конкурентів або вашого застосунку.

3. Автоматичний підбір слів (Auto, Search Match) – вбудований інструмент за допомогою якого ASA автоматично підбирає ключові слова для вашої рекламної кампанії з опису та аналізу вашого застосунку. Автоматичний тип використовується на подальших етапах для поповнення груп ключових слів та визначення їх типу (точна або широка відповідність).

4. Мінус-слова (Negative words) - це означає, що ваше оголошення не відображатиметься, коли користувач використовує точний пошуковий термін або пошукові терміни, які містять мінус-слова. Їх найкраще використовувати для наступних груп слів:

- Слова з низьким трафіком
- Нерелевантні слова
- Слово «безкоштовно» для застосунків, де застосунок платний

Узагальнення вище сказаного можна подивитись у таблиці 2.1

Таблиця 2.1 – Розподіл типів ключових слів в залежності від типу кампанії



Розглянемо на прикладі пошукового ключового слова – photo edit free різні варіанти, яким типам ключових слів підходить відповідний пошуковий запит, як наведено в таблиці 2.2:

Таблиця 2.2 – Відповідність пошукових запитів (Джерело: Apple Search Ads)

		Bidded keywords: photo edit free		Negative keywords: photo edit free	
Search query scenarios	Customer's search term	Exact match	Broad match	Exact match	Broad match
Words in order	photo edit free	✓	✓	✗	✗
Close variants	photo editor free	✓	✓		
	photo edit free	✓	✓		
Not in order	edit photo free		✓		✗
Partial words	photo edit		✓		
	edit free		✓		
Partial words	photo edit collage free		✓		✗
	free crop photo edit		✓		✗
	resize edit photo free		✓		✗
Synonyms/related words	picture edit		✓		
	picture edit text		✓		

2.2 Індекс популярності ключового слова

Вибір ключових слів є життєво важливим для видимості ваших додатків у конкурентному середовищі. Щоб отримати максимальну частку голосу/залучень, має сенс розглянути обсяг пошуку за ключовими словами в App Store. Лише якщо користувачі вже шукають ключові слова, вони потенційно можуть залучити більше трафіку.

Одним із способів оцінити обсяг пошуку в App Store є перевірка показників популярності в ASA. Цей показник показує частку користувачів, які ввели пошуковий запит. У Apple Search Ads оцінка позначена синіми крапками, де чим більше точок, тим більша популярність. Однак за допомогою деяких інструментів, індекс популярності пошуку Apple візуалізується з реальними числами від 5 до 100 (рис. 2.3)

Average Search Popularity: 27.5

Median: 27

Percentage of keywords with

- Search Popularity <= 10 : 15%
- Search Popularity <= 20 : 37%
- **Search Popularity <= 30 : 61%**
- Search Popularity <= 40 : 80%
- Search Popularity <= 50 : 91%
- Search Popularity <= 60 : 97%

Source: Apptweak

Рисунок 2.3 – Індекс популярності слів та відсоток відповідних ключових слів
Справді, цей діапазон може допомогти приблизно оцінити популярність.
Популярність пошуку та покази в App Store мають експоненційну залежність (рис. 2.4)

$$\text{Max Impressions} = 254.4443 * \exp(0.0615 * \text{Search Popularity})$$

 SearchAdsHQ |  Phiture

Search Popularity	Estimated impressions per day
10	500
20	900
30	1,600
40	3,000
50	5,500
60	10,000
70	19,000
80	35,000

Рисунок 2.4 – Формула обсягу показів для певного ключового слова
(Джерело: SearchAdsHQ)

Наприклад, для популярності пошуку 60 максимальна кількість показів становить щонайменше 10 000, тоді як для популярності пошуку 82 – щонайменше 40 000 показів.

РОЗДІЛ 3: Модель торгів в ASA:

Apple Search Ads працює як аукціон другого біда. Наприклад, розглянемо два ідентичні застосунки. Тобто, якщо дві компанії хочуть просувати свої програми для однієї аудиторії за допомогою однакових ключових слів і призначають ставки 2,00\$ і 3,00\$ відповідно, виграє програма з вищою ставкою. У цьому випадку остаточна ціна, яку переможець заплатить за натискання, становить не 3\$, а $2,00\$ + 1 \text{ цент}$

Якщо перша компанія хоче перевищити ставку CPT конкурента і почати показ реклами з певним ключовим словом, їй потрібно збільшити максимальну ставку CPT до $3,00\$ + 1 \text{ цент}$. Зміна ставки відбувається після старту рекламної кампанії з причин дуже низького трафіку або оцінка більш оптимальної ціни. На старті рекламної кампанії варто використовувати індикатор ставок, який пропонує Apple Search Ads. Не так давно ця, здавалося б, простота збентежила рекламних менеджерів, оскільки Apple Search Ads не давала підказки щодо ставок конкурентів і не пропонувала ставки на основі середньої CPT для ключового слова. Тому пошукова реклама покращила умови призначення ставок для своїх користувачів. Статистика ставок була розроблена як рекомендації, щоб підвищити шанси на перегляд оголошення.

РОЗДІЛ 4: Оптимізаційні моделі ROI та ROAS:

4.1 Найважливіші показники оптимальної рекламної кампанії

Після детального розгляду функціонування ASA треба визначити, які показники дають вам уявлення про те, як заробляти більше, витрачаючи менше.

Причина того, що ці показники настільки важливі, полягає в тому, що в Apple Search Ads, хоча ви платите за натискання користувачів на банер, ви заробляєте гроші, лише коли користувач витрачає гроші у вашому додатку та при завантаженні додатку. Є рекламна «воронка» (рис. 4.1), кожен етап якої має бути оптимізований, щоб гарантувати, що ваші витрати на рекламу повертаються зі оптимальною маржею в кінець.

Можна відстежувати багато показників, щоб забезпечити ефективність «воронки». Розглянемо найважливіші з них.

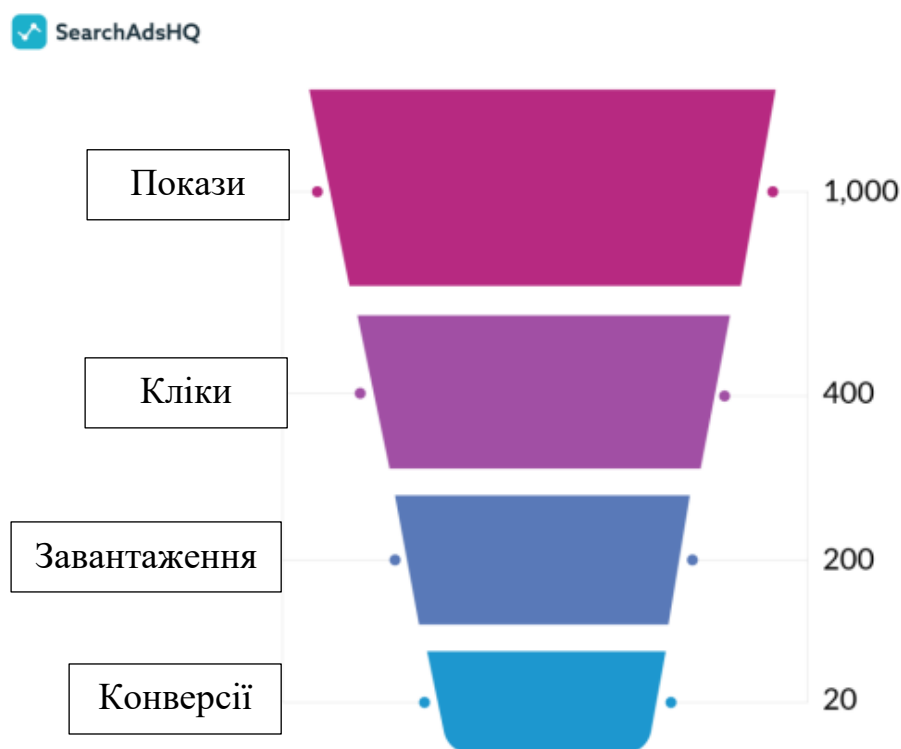


Рисунок 4.1 – «Воронка» ключового слова в рекламній кампанії (Джерело: Search Ads HQ)

4.1.1 Ціна за дію (CPA)

Маркетологи, отримують дохід за оптимальне налаштування CPA. CPA – це середня вартість дій конверсії (завантаження, підписка, купівля в додатку), які виконують користувачі. CPA показує вам, скільки ви витрачаєте на користувача, який підписався на ваш додаток. Якщо ви хочете, щоб ваші кампанії з пошуковою рекламою були прибутковими, ці витрати мають бути нижчими за дохід, отриманий від дії

$$CPA = \frac{\text{витрати}}{\# \text{ дій}}$$

Цільові дії після встановлення є специфічними для кожної програми та залежать від прийнятої стратегії монетизації. Цільовими діями є:

- Завантаження платної програми
- Покупки в програмі;
- Підписки;
- Початковий безкоштовний період з подальшою підпискою тощо.

Приклад: Скажімо, ви витратили 100 доларів на рекламу, що призвело до 10 покупок у програмі. $\$100/10 = \10 – це середня сума, яку ви платите за цільову дію.

4.1.2 Рентабельність витрат на рекламу (ROAS) і рентабельність інвестицій (ROI)

Існують покупки в програмі, коли користувачі платять за різними цінами (наприклад, користувач, який купує монети в грі, може внести від 1 до 1000 доларів США). Ця модель монетизації не дозволяє розрахувати цільову CPA. У таких ситуаціях маркетологи відстежують свій ROAS.

ROAS показує кількість доларів, які компанія заробляє на рекламі на кожен долар, витрачений на неї

$$ROAS = \frac{\text{дохід}}{\text{витрати на рекламу}}$$

Приклад: Припустимо, ви виділяєте 100\$ на рекламу, що принесе вам 120\$ доходу. Порахуємо ROAS:

$$ROAS = \frac{\text{дохід}}{\text{витрати на рекламу}} * 100\% = \frac{120\$}{100\$} * 100\% = 120\% = 1,2$$

Інтерпретація результату: ви маєте кращу сторону балансу витрат і прибутку, оскільки за 1\$, вкладений в рекламу, ви заробляєте 1,2\$. У ідеальному світі дохід від рекламних кампаній має перевищувати витрати на рекламу, тому здорова рентабельність інвестицій у рекламу перевищує 100%.

І навпаки, рентабельність інвестицій у рекламу нижче 100% є попереджувальним сигналом про необхідність оптимізації рекламних витрат.

Розглянемо ще один важливий індикатор – рентабельність інвестицій (ROI):

$$ROI = \frac{\text{дохід} - \text{витрати}}{\text{витрати}} * 100\%$$

Приклад:

$$ROI = \frac{120\$ - 100\$}{100\$} * 100\% = 20\%$$

Хоча ROAS вимірює ефективність одного долара, вкладеного в рекламу, ROI враховує маркетингові зусилля в цілому, тобто накладні витрати, зарплату, навчання співробітників та інші фактори, включаючи витрачені долари.

У той час як ROAS вимірює ефективність певного маркетингового каналу, ROI вимірює загальну маркетингову ефективність. Ще одна відмінність між цими двома показниками полягає в тому, що ROAS завжди показує додатне число, тоді як ROI може бути негативною.

Можна показати залежність між ROI та ROAS:

$$ROI = ROAS - 100\% = 120\% - 100\% = 20\%$$

4.1.3 Ціна за клік (CPT)

Ціна за клік (CPT) – це базова модель ціноутворення пошукових оголошень. У ASA ви платите лише тоді, коли користувачі торкаються вашого банера, а CPT — це показник, який показує, скільки вам коштує цей клік.

$$CPT = \frac{\text{витрати на рекламу}}{\text{\# кліків}}$$

Наприклад, ви витратили 1000\$ на рекламу та отримали 500 натискань на банер, тобто:

$$CPT = \frac{1000\$}{500} = 2\$$$

За допомогою ставки CPT можна брати участь в аукціоні другого біда (як саме працює аукціон – РОЗДІЛ 3). Правильно встановлюючи CPT, ви зможете отримувати високий трафік за оптимальну ціну. Чим вища CPT, тим швидше зменшується рекламний бюджет, і в якийсь момент вам буде важко дотримуватися бажаної ROAS. З іншого боку, низький CPT зменшує шанси потрапити на аукціон. Як наслідок, ваша оголошення не буде мати суттєвого трафіку.

Подібно до CPA та ROAS, CPT відстежує загальний стан вашої маркетингової кампанії.

4.1.4 Обсяг конверсій (CV)

Досягнення цільових значень CPA, ROAS і CPT є життєво важливим, але це лише половина роботи.

В ідеалі хороші показники мають підкріплюватися розумною кількістю конверсій. Конверсія – це відношення числа користувачів додатку, які виконали на ньому певні цільові дії (придбання, реєстрацію, підписку тощо), до загального числа користувачів, виражене у відсотках.

Приклад: Ваш додаток завантажило 500 людей протягом тижня, за цей період було здійснено 7 цільових дій. Тож відсоток конверсії буде:

$$\frac{7}{500} * 100\% = 1,4\%$$

CV – це обсяг конверсій, створених за певний період, напр. тиждень або місяць.

Показник не має нічого, що говорить про те, наскільки ефективно ви рекламуєте, але ви повинні підтримувати його на певному рівні для високої ефективності реклами. Якщо ваші рекламні кампанії в пошуковій мережі добре оптимізовані, кількість конверсій з часом має зростати, а вартість реклами має залишатися на тому ж рівні або знижуватися.

4.1.5 Коефіцієнт конверсії (CR)

Коефіцієнт конверсії (CR) є ідеальним показником, щоб оцінити, чи «воронка» є ефективною. CR показує загальну кількість конверсій:

$$CR = \frac{\# \text{ конверсій}}{\# \text{ кліків}} * 100\%$$

Коефіцієнт конверсій є оптимальним, якщо CR високий, оскільки це означає, що ви платите менше за певну цільову дію, а бюджет дозволяє залучити більше користувачів. Але якщо ваш коефіцієнт конверсії знижується, це означає, що потрібна оптимізація.

Приклад.

$$CR = \frac{\# \text{ конверсій}}{\# \text{ кліків}} * 100\% = \frac{20}{400} * 100\% = 5\%$$

Ви можете оптимізувати конверсію на будь-якому етапі рекламної кампанії, включаючи події після початкового.

4.2 Оптимізація кожного з показників на прикладах

4.2.1 Ціна за дію (CPA)

Наприклад, ви продаєте фітнес-додаток на основі підписки. Підписка – це ваша цільова дія. Припустимо, ви знаєте середню дохідність за весь час (LTV) від ваших користувачів (тобто прибуток від взаємодії з користувачем ви отримуєте), і ця LTV залишається незмінною.

Скажімо, середній користувач платить 40\$ на рік (ARPU) і залишається з вами протягом 2 років (LT).

Знаючи ARPU і LT, ви можете розрахувати LTV:

$$LTV = LT * ARPU = 2 * 40\$ = 80\$$$

Отже, в цьому випадку вартість цільової дії не повинна перевищувати LTV = 80\$. Але ви не можете заплатити 80\$, оскільки потрібно отримати ще й прибуток.

Справа в тому, що LTV – це валовий прибуток. Крім того, існують фіксовані витрати на придбання та підтримку, які вам, ймовірно, доведеться заплатити – 30% комісія за

підписку користувача на App Store і близько 20% у вигляді додаткових податків, підтримки клієнтів або витрат на виробництво контенту.

Повернемося до фітнес-додатка. Враховуючи 50% (30% + 20%) знижки від LTV користувача, ваш прибуток на одного абонента складе:

$$\text{Прибуток} = LTV * 50\% = 80\$ * 50\% = 40\$$$

Якщо CPA = 40\$, то це буде точкою беззбитковості. Але якщо ви хочете, щоб ваш прибуток був більшим за витрати, спробуйте ще більше знизити CPA

$$\text{Цільова CPA} = 75\% * \text{Прибуток} = 75\% * 40\$ = 30\$$$

І ось, 30\$ — це мінімальна CPA, яка принесе 10\$ прибутку на вашого абонента фітнес-додатка за 2 роки LT. Приклад наведено в таблиці 4.2:

Таблиця 4.2 – Вирахування оптимальної CPA

LTV	80\$	ARPU * LT
Віднімемо 30% як вартість в App Store Віднімемо 20% на поточні витрати (податки тощо)	40\$	Фіксовані вартості для підписки, податків тощо
Максимальний CPA	40\$	Якщо ви поставите цю CPA, то ваш прибуток буде дорівнювати нулю. Якщо більше, то буде збиткова кампанія
75% * CPA (щоб отримати 25% маржі)	30\$	Якщо використати цю CPA, то отримаєте прибуток 10\$ з кожного користувача, який заплатить протягом LT

Тепер, коли ви знаєте свою цільову CPA, ви можете налаштувати ставку CPT для ключового слова відповідно до формули:

$$\text{Макс. ставка CPT} = \text{коефіцієнт конверсії} * \text{Цільова CPA} * 1,2$$

Ви можете задатися питанням, звідки взявся 1,2. Насправді CPT зазвичай приблизно на 10-20% нижчий за встановлену максимальну ставку CPT через другий аукціон ціни. Наприклад, якщо ви встановили максимальну ставку CPT на 1,00\$, ви, ймовірно, витратите приблизно на 20% менше – в середньому 0,80\$ за клік. Отже, щоб компенсувати різницю, помножте ставку на 1,2.

$$\text{Коефіцієнт конверсії} = \frac{\text{конверсії}}{\text{кліки}} = \frac{20}{400} = 5\%$$

$$\text{Макс. ставка CPT} = 5\% * 30\$ * 1,2 = 1,8\$$$

Цей метод оптимізації ставки CPA слід застосовувати до кожного ключового слова відповідно до його рекламної «воронки» (рисунки 4.1)

4.2.2 Рентабельність витрат на рекламу (ROAS) і рентабельність інвестицій (ROI)

Ідея оптимізації ROAS полягає в тому, щоб зберегти дохід на певному рівні. Тут важливі не наші витрати на рекламу, а сума прибутку, яку ви отримуєте від своїх інвестицій.

Уявімо, що ви продаєте ігровий додаток, де користувачі можуть купувати золото за реальні гроші. Одна покупка в додатку може принести вам від 1\$ до 1000\$, що означає, що ви не можете розрахувати середню CPA. Отже, тут використовуємо ROAS.

Подібно до оптимізації CPA, доведеться відрізати 50% отриманого доходу – 30% як комісію App Store і 20% як поточні витрати.

Розрахуємо маржу на 1000 доларів прибутку (табл. 4.3):

Таблиця 4.3 – Розрахунок оптимального ROAS

Дохід	1000\$	
Віднімемо 30% як вартість в App Store Віднімемо 20% на поточні витрати (податки тощо)	500\$	Фіксовані вартості для підписки, податків тощо
Прибуток від продажу	500\$	
Максимальні витрати на рекламу	500\$	
Максимальний ACoS	50%	ACoS -- відсоток бюджету, що був отриманий від продажу товару, який був витрачений на рекламу. 50% максимальний відсоток для не збиткової рекламної кампанії
Мінімальний ROAS	2.0x (200%)	$ROAS = 1 / ACoS = 1 / 0.5 = 2.0x$ Рентабельність витрат на рекламу становить 200%
Цільовий ROAS	2.5x (250%)	Якщо хочете отримати прибуток, слідкуйте, щоб рентабельність була вища за мінімальний ROAS, іншими словами, реінвестуйте меншу частку прибутку і рекламу. Таким чином цільовий ACoS = $1 / 2.5 = 40\%$ (менше за 50%)

Деякі компанії хочуть досягти цільової ROAS в 2,5 рази, інші – у 4 рази. Ви вирішуєте, яку частину прибутку від продажу ви хочете витратити на рекламу.

Переходимо до оптимізації ставок. Нижче наведена формула для підрахунку максимальної ставки CPT для цієї послідовності:

$$\text{Макс. ставка CPT} = \left(\frac{\text{дохід}}{\text{кліки}} \right) * \left(\frac{1}{\text{цільовий ROAS}} \right) * 1,2$$

$$\text{Макс. ставка CPT} = \frac{1000\$}{400} * \left(\frac{1}{2,5} \right) * 1,2 = 1,2\$$$

Це спосіб оптимізації ставки для моделі на основі ROAS. Ви повинні застосувати цей розрахунок до кожного ключового слова на основі його конкретної «воронка».

В чому є розбіжності ROAS та ROI? Оскільки невідповідність даних може досягати 70%, ROI у рекламу розраховується, ігноруючи, що невідповідність спотворюється.

Приклад: Припустимо, що ви рекламуєте програму, і ваша маркетингова діяльність дала такі результати (таблиця 4.4):

Витрати на рекламу в ASA	200\$
Завантаження	100
Повторні завантаження	27 (27%)
Конверсії з увімкненою функцією LAT	8 (8%)
Збій API, затримки кліків, налаштування атрибуції	5 (5%)
MMP	50
ARPU	5\$

Таблиця 4.4 – Дані програми, що рекламуємо

Перш за все, давайте обчислимо ROAS, ігноруючи розбіжності даних:

$$\text{Справжній ROAS} = \frac{(\text{встановлення MMP} * \text{ARPU} - \text{витрати})}{\text{витрати}} * 100\%$$

$$\text{Справжній ROAS} = \frac{(50 * 5\$ - 200\$)}{200\$} * 100\% = 25\%$$

ROAS, що дорівнює 25%, означає, що ви витрачаєте більше, ніж отримуєте, і або потрібна оптимізація ключового слова, або рекламний канал для вас не є оптимальним.

Тепер врахуємо приховані інсталяції. Для простоти давайте разом додамо повторні завантаження та LAT On Conversions (за умови, що вони не перекриваються).

$$\text{Скориговані завантаження MMP} = 50 + 27 + 8 + 5 = 90$$

$$\text{Скоригована ROAS} = \frac{(90 * 5\$ - 200\$)}{200\$} * 100\% = 125\%$$

Подивіться на суттєву різницю: скоригована ROAS в 5 разів вища за справжню ROAS. Розраховуючи таким чином ROAS, ви можете побачити справжню картину ефективності ваших Apple Search Ads і таким чином призначати ставки більш конкурентоспроможними та ефективніше масштабувати свої кампанії.

4.2.3 Ціна за клік (CPT)

Висока або низька ставка CPT не працюватиме добре в пошуковій рекламі. Наприклад:

- Відкоригуйте ставки до цільових показників. Це найпростіший метод оптимізації. Зосередьтеся на цільових показниках – CPA та ROAS і відповідно збільшуйте або зменшуйте ставку CPT (див. попередні розділи).

- Робота з сегментами. Наприклад, за статистикою жіноча аудиторія має більше шансів на конверсію порівняно з чоловіками. Отже, якщо ви зменшите ставку CPT для чоловічого сегмента, середня CPT також знизиться. Ось ще один поширений приклад. Аудиторія може вести себе по-різному протягом тижня – вони активні з понеділка по п'ятницю і неактивні у вихідні дні. Вимкніть рекламу на вихідних, і

середня CPT зменшиться. Зважайте на підтримку розумного балансу між ціною за клік та обсягом конверсій.

4.2.4 Обсяг конверсій (CV)

Додайте більше ключових слів. Шукаючи та додаючи релевантні ключові слова, ви збільшуєте свої шанси отримати додаткові покази, а отже, більше конверсій. Збільште/перерозподіліть бюджет. Дуже обмежений щоденний бюджет може значно обмежити ваші покази. Розумно іноді переглядати та оптимізувати свої розподіли та визначати сфери, у які варто інвестувати. Зосередьтеся на локалізаціях, групах оголошень і ключових словах з сильною продуктивністю.

Збільште ставку. Низькі ставки можуть бути причиною відмови від участі в аукціоні, а отже, і втрати трафіку. Таким чином, підвищення ставок є розумним способом підвищення конкурентоспроможності.

РОЗДІЛ 5: Хід роботи

1.

```
asa <- read.csv("asa_dataset_final_11112021.csv")
clear_data = asa[, c(1, 4, 5, 7, 9, 10, 11, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25)]
str(clear_data)
dim(clear_data)
summary(clear_data)
sum(is.na(clear_data))
attach(clear_data)
```

Імпортуємо датасет (репорт з Apple Search Ads). Трохи чистимо датасет від стовпців, як не впливають на аналіз. Отже, остаточний опис вигляд виграшки такий: 253840 рядків і 19 стовпців

campaign_name – назва рекламної кампанії

country_or_region – країни чи регіони

ad_group_name – назва груп оголошень

keyword – ключове слово

cpt_bid – ставка на ціну за клік на аукціоні

keyword_match_type – тип ключового слова

spend – витрати

taps – кліки

installs – завантаження

impressions – покази

ttr – коефіцієнт клікабельності, розраховується як :

$$\frac{\text{кількість людей, що зробили клік}}{\text{кількість людей, що побачили рекламу}}$$

cr – коефіцієнт конверсій

avg_cpa – середня ціна за дію

avg_cpt – середня ціна за клік

new_downloads – нові завантаження

redownloads – повторні завантаження

lat_on_installs – скриті завантаження

lat_off_installs – відкриті завантаження

Результат коду:

```
> asa <- read.csv("asa_dataset_final_11112021.csv")
> clear_data = asa[, c(1, 4, 5, 7, 9, 10, 11, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25)]
> str(clear_data)
'data.frame': 253840 obs. of 19 variables:
 $ date : chr "31.10.2021" "31.10.2021" "31.10.2021" "31.10.2021" ...
 $ campaign_name : chr "ww29" "ww29" "ww29" "ww29" ...
 $ country_or_region : chr "X27" "X27" "X27" "X27" ...
 $ ad_group_name : chr "YY70" "YY70" "YY72" "YY70" ...
 $ keyword : chr "Zz6816" "Zz8973" "Zz10531" "Zz8646" ...
 $ cpt_bid : num 1 1 0.5 0.5 0.5 0.7 0.5 2.2 1 0.5 ...
 $ keyword_match_type: chr "BROAD" "BROAD" "EXACT" "EXACT" ...
 $ spend : num 0 0 0 0 0 ...
 $ taps : int 0 0 0 0 1 0 152 0 0 ...
 $ installs : int 0 0 0 0 1 0 111 0 0 ...
 $ impressions : int 1 19 5 1 1 47 1 1798 1 1 ...
 $ ttr : num 0 0 0 0 0 0.0213 0 0.0845 0 0 ...
 $ cr : num 0 0 0 0 0 ...
 $ avg_cpa : num 0 0 0 0 0 ...
 $ avg_cpt : num 0 0 0 0 0 ...
 $ new_downloads : int 0 0 0 0 1 0 31 0 0 ...
 $ redownloads : int 0 0 0 0 0 0 80 0 0 ...
 $ lat_on_installs : int 0 0 0 0 1 0 10 0 0 ...
 $ lat_off_installs : int 0 0 0 0 0 0 2 0 0 ...
> dim(clear_data)
[1] 253840 19
> summary(clear_data)
      date      campaign_name      country_or_region      ad_group_name      keyword      cpt_bid      keyword_match_type      spend
Length:253840 Length:253840 Length:253840 Length:253840 Length:253840 Min. : 0.0500 Length:253840 Min. : 0.000
Class :character Class :character Class :character Class :character Class :character 1st Qu.: 0.5000 Class :character 1st Qu.: 0.000
Mode :character Mode :character Mode :character Mode :character Mode :character Median : 0.5000 Mode :character Median : 0.000
Mean : 0.8689 Mean : 2.133
3rd Qu.: 1.0000 3rd Qu.: 0.000
Max. :13.2200 Max. :3562.072

      taps      installs      impressions      ttr      cr      avg_cpa      avg_cpt      new_downloads
Min. : 0.000 Min. : 0.0 Min. : 0.00 Min. :0.000000 Min. : 0.0000 Min. : 0.0000 Min. :0.000000 Min. : 0.0000
1st Qu.: 0.000 1st Qu.: 0.0 1st Qu.: 1.00 1st Qu.:0.00000 1st Qu.: 0.0000 1st Qu.: 0.0000 1st Qu.:0.000000 1st Qu.: 0.0000
Median : 0.000 Median : 0.0 Median : 1.00 Median :0.00000 Median : 0.0000 Median : 0.0000 Median :0.000000 Median : 0.0000
Mean : 2.928 Mean : 1.8 Mean : 86.42 Mean :0.07087 Mean : 0.1118 Mean : 0.1113 Mean :0.09974 Mean : 0.4539
3rd Qu.: 0.000 3rd Qu.: 0.0 3rd Qu.: 4.00 3rd Qu.:0.00000 3rd Qu.: 0.0000 3rd Qu.: 0.0000 3rd Qu.:0.000000 3rd Qu.: 0.0000
Max. :3620.000 Max. :1334.0 Max. :237959.0 Max. :3.00000 Max. :57.0000 Max. :70.7700 Max. :6.63000 Max. :480.0000

      redownloads      lat_on_installs      lat_off_installs
Min. : 0.000 Min. : 0.0000 Min. : 0.0000
1st Qu.: 0.000 1st Qu.: 0.0000 1st Qu.: 0.0000
Median : 0.000 Median : 0.0000 Median : 0.0000
Mean : 1.346 Mean : 0.0609 Mean : 0.0958
3rd Qu.: 0.000 3rd Qu.: 0.0000 3rd Qu.: 0.0000
Max. :1087.000 Max. :582.0000 Max. :541.0000
> sum(is.na(clear_data))
[1] 0
```

2.

```
data <- subset(clear_data, impressions != 0 & spend != 0)
summary(data)
data$keyword_match_type <- as.factor(data$keyword_match_type)
data$date <- as.factor(data$date)
attach(data)
```

Створюємо ще один датасет, куди записуємо дані, де показів і витрат більше нуля.

Тобто ті кампанії, на яких був відгук. Перетворюємо тип ключового слова і дату на фактори

Результат коду:

```
> data <- subset(clear_data, impressions != 0 & spend != 0)
> summary(data)
      date      campaign_name      country_or_region      ad_group_name      keyword      cpt_bid      keyword_match_type      spend
Length:50906 Length:50906 Length:50906 Length:50906 Length:50906 Min. : 0.050 Length:50906 Min. : 0.05
Class :character Class :character Class :character Class :character Class :character 1st Qu.: 0.500 Class :character 1st Qu.: 0.34
Mode :character Mode :character Mode :character Mode :character Mode :character Median : 1.000 Mode :character Median : 0.75
Mean : 1.273 Mean : 10.63
3rd Qu.: 1.900 3rd Qu.: 2.12
Max. :11.000 Max. :3562.07

      taps      installs      impressions      ttr      cr      avg_cpa      avg_cpt      new_downloads
Min. : 1.0 Min. : 0.000 Min. : 1.0 Min. :0.0010 Min. : 0.000 Min. : 0.0000 Min. :0.0500 Min. : 0.000
1st Qu.: 1.0 1st Qu.: 0.000 1st Qu.: 2.0 1st Qu.:0.0500 1st Qu.: 0.000 1st Qu.: 0.0000 1st Qu.:0.2470 1st Qu.: 0.000
Median : 1.0 Median : 1.000 Median : 13.0 Median :0.2000 Median : 0.500 Median : 0.3400 Median :0.4300 Median : 0.000
Mean : 14.6 Mean : 8.845 Mean : 415.3 Mean :0.3534 Mean : 0.556 Mean : 0.5542 Mean :0.4960 Mean : 2.253
3rd Qu.: 4.0 3rd Qu.: 3.000 3rd Qu.: 62.0 3rd Qu.:0.5000 3rd Qu.: 1.000 3rd Qu.: 0.7400 3rd Qu.:0.6325 3rd Qu.: 1.000
Max. :3620.0 Max. :1334.000 Max. :237959.0 Max. :3.0000 Max. :57.000 Max. :70.7700 Max. :6.6300 Max. :480.000

      redownloads      lat_on_installs      lat_off_installs
Min. : 0.000 Min. : 0.000 Min. : 0.0000
1st Qu.: 0.000 1st Qu.: 0.000 1st Qu.: 0.0000
Median : 0.000 Median : 0.000 Median : 0.0000
Mean : 6.592 Mean : 0.301 Mean : 0.4726
3rd Qu.: 2.000 3rd Qu.: 0.000 3rd Qu.: 0.0000
Max. :1087.000 Max. :582.000 Max. :541.0000
> data$keyword_match_type <- as.factor(data$keyword_match_type)
> data$date <- as.factor(data$date)
> attach(data)
```

3.

```
campaign_ww1 <- subset(data, campaign_name == 'ww1')
attach(campaign_ww1)

total_spend = sum(campaign_ww1$spend)
total_installs = sum(campaign_ww1$installs)
total_taps = sum(campaign_ww1$taps)
average_cpa = total_spend/total_installs
average_cpa
average_cpt = total_spend/total_taps
average_cpt
to_negative <- subset(campaign_ww1, (spend>average_cpa & installs<1 & avg_cpa != 0)||
                        (spend>average_cpt & taps<1 & avg_cpt !=0))
```

Тренуємо програму на одній кампанії та рахуємо загальні витрати, завантаження та кліки, щоб розрахувати загальну ціну за дію і клік. На основі цих даних за найгіршими показниками фільтруємо та відносимо до мінус слів. В нашому випадку таких випадків не виявилось

Результат коду:

```
> campaign_ww1 <- subset(data, campaign_name == 'ww1')
> total_spend = sum(campaign_ww1$spend)
> total_installs = sum(campaign_ww1$installs)
> total_taps = sum(campaign_ww1$taps)
> average_cpa = total_spend/total_installs
> average_cpa
[1] 2.63816
> average_cpt = total_spend/total_taps
> average_cpt
[1] 0.9390319
> to_negative <- subset(campaign_ww1, (spend>average_cpa & installs<1 & avg_cpa != 0)||
+                               (spend>average_cpt & taps<1 & avg_cpt !=0))
```

to_negative	0 obs. of 19 variables
-------------	------------------------

4.

```
total_spend_data = sum(data$spend)
total_installs_data = sum(data$installs)
total_taps_data = sum(data$taps)
average_cpa_data = total_spend_data/total_installs_data
average_cpa_data
average_cpt_data = total_spend_data/total_taps_data
average_cpt_data
average_cr_data = total_installs/total_taps_data
average_cr_data

to_negative <- subset(data, (spend>average_cpa_data & installs<1 & avg_cpa != 0)||
                        (spend>average_cpt_data & taps<1 & avg_cpt !=0))
to_negative <- subset(data, (installs>0 & avg_cpa > average_cpa_data)||
                        (taps>0 & avg_cpt > average_cpt_data))
```

Пророблюємо ті самі дії з усіма кампаніями. Результат – аналогічний.

Результат коду:

```
> total_spend_data = sum(data$spend)
> total_installs_data = sum(data$installs)
> total_taps_data = sum(data$taps)
> average_cpa_data = total_spend_data/total_installs_data
> average_cpa_data
[1] 1.20227
> average_cpt_data = total_spend_data/total_taps_data
> average_cpt_data
[1] 0.7286132
> average_cr_data = total_installs/total_taps_data
> average_cr_data
[1] 0.001652808
> to_negative <- subset(data, (spend>average_cpa_data & installs<1 & avg_cpa != 0)|| (spend>average_cpt_data & taps<1 & avg_cpt !=0))
```

to_negative	0 obs. of 19 variables
-------------	------------------------

```
> to_negative <- subset(data, (installs>0 & avg_cpa > average_cpa_data)|| (taps>0 & avg_cpt > average_cpt_data))
```

to_negative	0 obs. of 19 variables
-------------	------------------------

5.

```
to_exact <- subset(data, (avg_cpa < average_cpa_data & avg_cpa !=0 & cr > average_cr_data) &
  (avg_cpt < average_cpt_data & avg_cpt!=0 & cr > average_cr_data) & keyword_match_type == "BROAD")
```

Кампанії з найкращими показниками відносимо до точного типу ключового слова.

Вийшло 190 таких слів.

Результат коду:

to_exact	190 obs. of 19 variables
----------	--------------------------

6.

```
keyword_ZZ1098 <- subset(to_exact, keyword == "ZZ1098")

linear_model <- lm(avg_cpt ~ cpt_bid + spend + taps + installs, data = keyword_ZZ1098)
summary(linear_model)

plot(x = spend, y = avg_cpt)
```

Робимо новий датасет для одного ключового слова – ZZ1098. Будуємо лінійну модель як залежить ціна за дію від ставки, витрат, кліків і завантажень. Аналізуючи результати, бачимо, що результати не оптимальні (погана F-статистика). Тому для подальших дій будуюмо графіки залежності від кожної змінної.

Результат коду:

```
> keyword_ZZ1098 <- subset(to_exact, keyword == "ZZ1098")
> linear_model <- lm(avg_cpt ~ cpt_bid + spend + taps + installs, data = keyword_ZZ1098)
> summary(linear_model)
```

Call:

```
lm(formula = avg_cpt ~ cpt_bid + spend + taps + installs, data = keyword_ZZ1098)
```

Residuals:

Min	1Q	Median	3Q	Max
-0.202497	-0.034994	0.007051	0.032310	0.232106

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	0.438921	0.083443	5.260	1.90e-05 ***
cpt_bid	-0.024344	0.090404	-0.269	0.79
spend	0.075314	0.015127	4.979	3.94e-05 ***
taps	-0.036189	0.006882	-5.259	1.91e-05 ***
installs	0.009045	0.015317	0.591	0.56

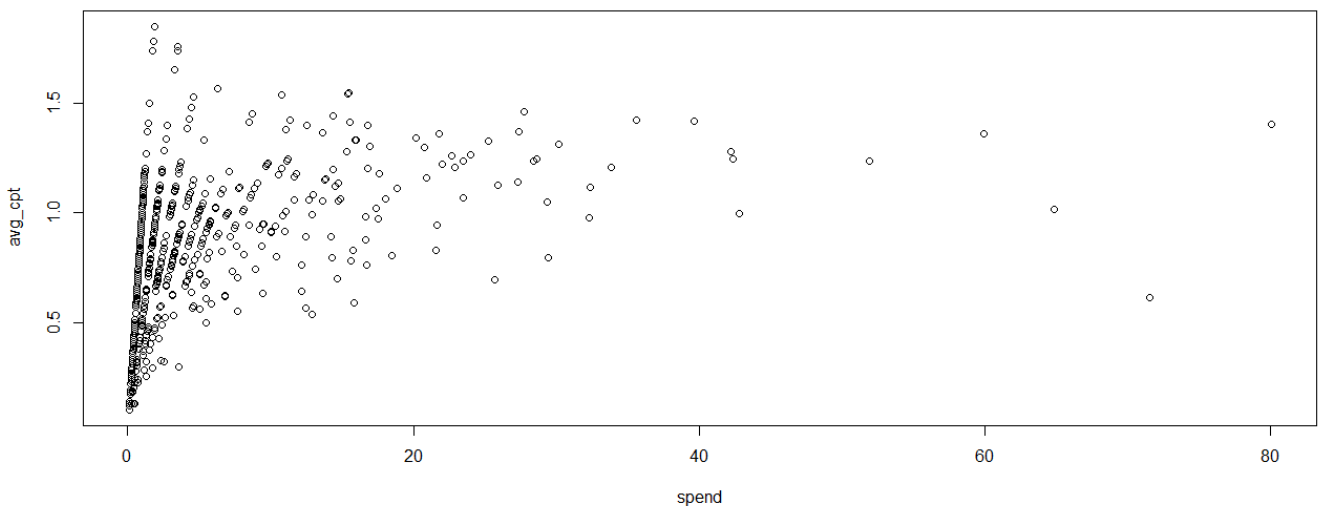
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

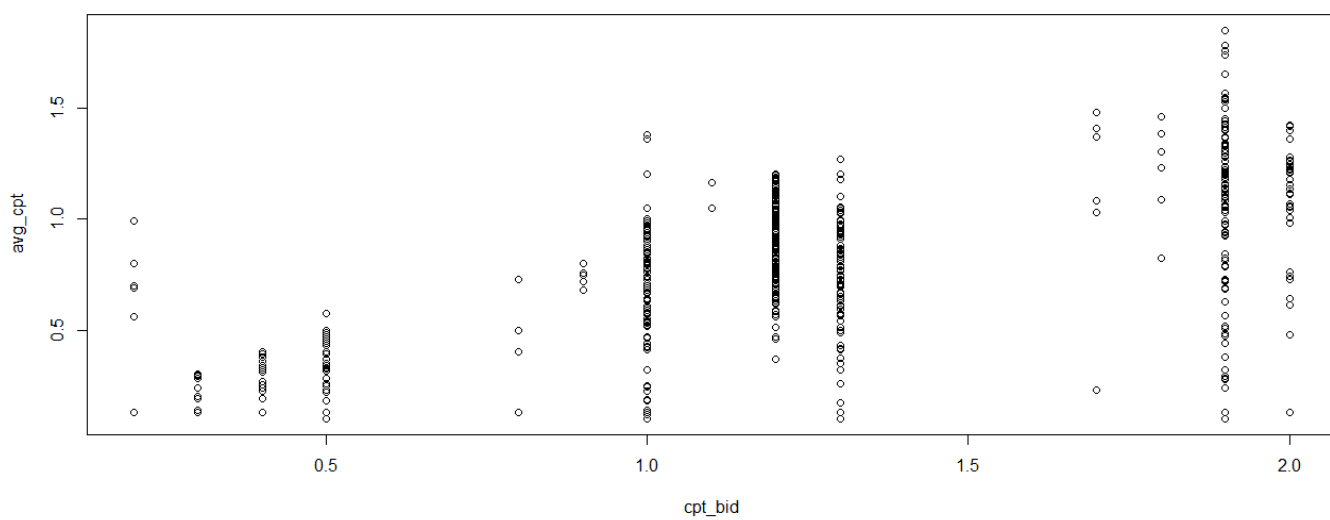
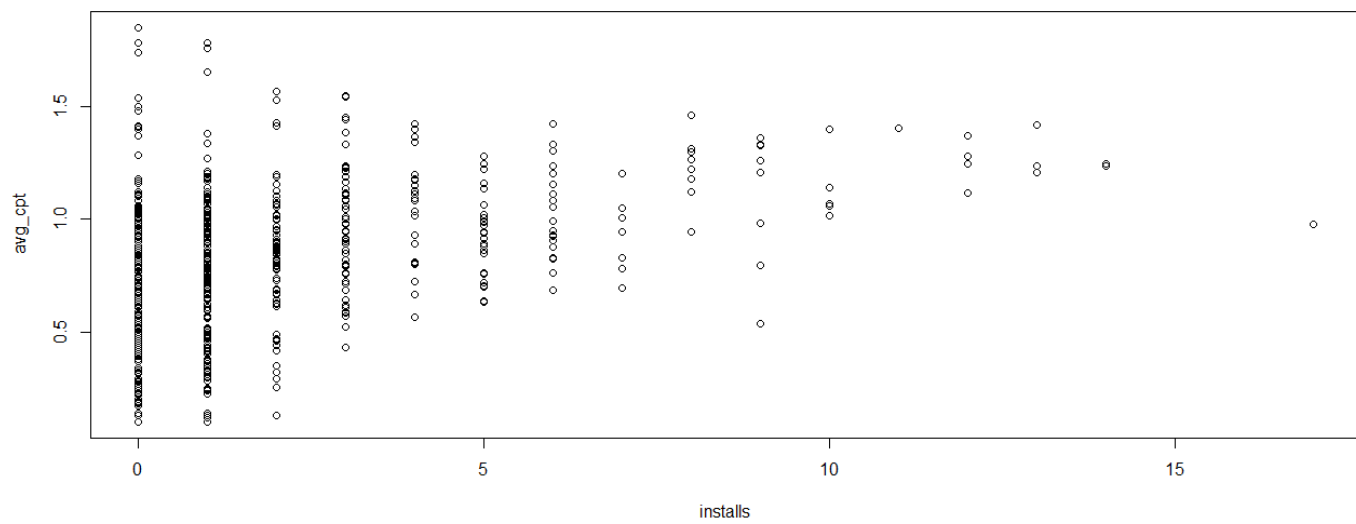
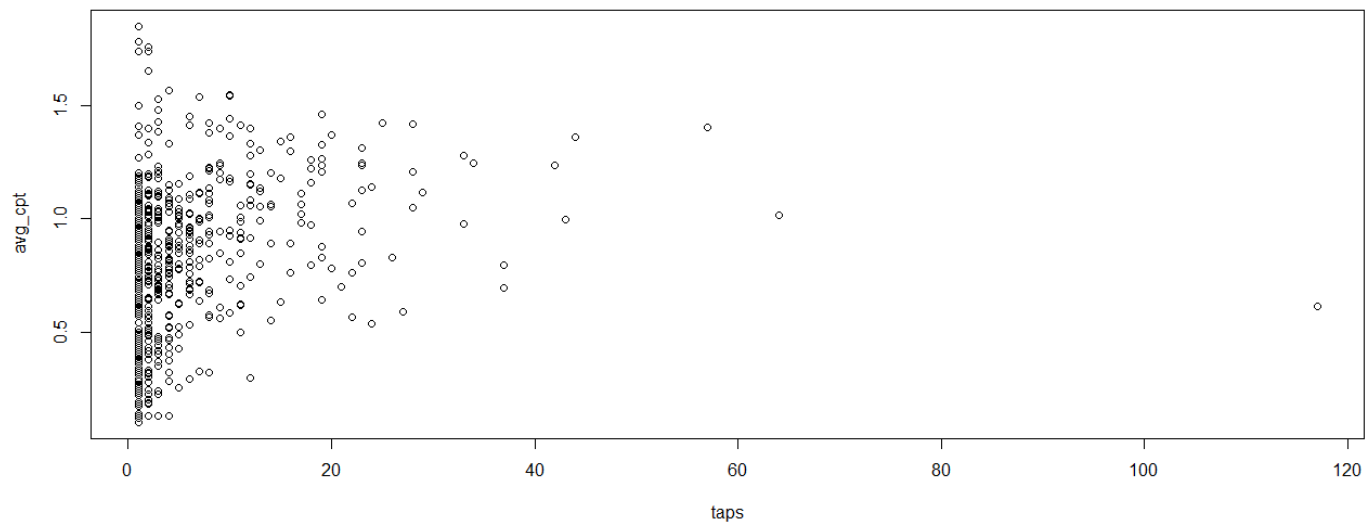
Residual standard error: 0.08519 on 25 degrees of freedom
Multiple R-squared: 0.612, Adjusted R-squared: 0.5499
F-statistic: 9.858 on 4 and 25 DF, p-value: 6.273e-05

7.

```
plot(x = spend, y = avg_cpt)
plot(x = taps, y = avg_cpt)
plot(x = installs, y = avg_cpt)
plot(x = cpt_bid, y = avg_cpt)
```

Такий вигляд мають відповідні графіки:





8.

```
model1 <- lm(avg_cpt ~ cpt_bid + I(sqrt(spend)) + taps + installs, data = keyword_ZZ1098)
summary(model1)

model2 <- lm(avg_cpt ~ cpt_bid + I(sqrt(taps)) + I(sqrt(spend)) + installs, data = keyword_ZZ1098)
summary(model2)

model3 <- lm(avg_cpt ~ I(sqrt(cpt_bid)) + I(sqrt(taps)) + I(sqrt(spend)) + installs, data = keyword_ZZ1098)
summary(model3)

model4 <- lm(avg_cpt ~ I(sqrt(cpt_bid)) + I(sqrt(taps)) + I(sqrt(spend)), data = keyword_ZZ1098)
summary(model4)
```

З графіків видно, що для більшості показників підходить модель з коренями (нелінійна), проте переконаємось в цьому, будуючи відповідні моделі. Аналізуючи їх, доходимо висновку, що найкраща модель – model4.

Результат коду:

```
> model1 <- lm(avg_cpt ~ cpt_bid + I(sqrt(spend)) + taps + installs, data = keyword_ZZ1098)
> summary(model1)

Call:
lm(formula = avg_cpt ~ cpt_bid + I(sqrt(spend)) + taps + installs,
    data = keyword_ZZ1098)

Residuals:
    Min       1Q   Median       3Q      Max
-0.096385 -0.051228 -0.007747  0.033047  0.272106

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.259161   0.081163   3.193  0.00378 **
cpt_bid      -0.119670   0.088125  -1.358  0.18661
I(sqrt(spend)) 0.349827   0.063350   5.522  9.72e-06 ***
taps         -0.044990   0.007277  -6.183  1.82e-06 ***
installs      0.017047   0.013656   1.248  0.22349
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.08069 on 25 degrees of freedom
Multiple R-squared:  0.6519,    Adjusted R-squared:  0.5962
F-statistic: 11.7 on 4 and 25 DF,  p-value: 1.71e-05

> model2 <- lm(avg_cpt ~ cpt_bid + I(sqrt(taps)) + I(sqrt(spend)) + installs, data = keyword_ZZ1098)
> summary(model2)

Call:
lm(formula = avg_cpt ~ cpt_bid + I(sqrt(taps)) + I(sqrt(spend)) +
    installs, data = keyword_ZZ1098)

Residuals:
    Min       1Q   Median       3Q      Max
-0.135946 -0.024502 -0.001512  0.013188  0.173757

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.446034   0.058441   7.632 5.48e-08 ***
cpt_bid      -0.032810   0.063992  -0.513   0.613
I(sqrt(taps)) -0.265073   0.027236  -9.732 5.53e-10 ***
I(sqrt(spend)) 0.431695   0.049153   8.783 4.11e-09 ***
installs     -0.005266   0.009176  -0.574   0.571
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.05864 on 25 degrees of freedom
Multiple R-squared:  0.8162,    Adjusted R-squared:  0.7867
F-statistic: 27.75 on 4 and 25 DF,  p-value: 7.161e-09
```

```
> model3 <- lm(avg_cpt ~ I(sqrt(cpt_bid)) + I(sqrt(taps)) + I(sqrt(spend)) + installs, data = keyword_ZZ1098)
> summary(model3)

Call:
lm(formula = avg_cpt ~ I(sqrt(cpt_bid)) + I(sqrt(taps)) + I(sqrt(spend)) +
    installs, data = keyword_ZZ1098)

Residuals:
    Min       1Q   Median       3Q      Max
-0.13560 -0.02415 -0.00151  0.01295  0.17436

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   0.473973   0.104840   4.521 0.000129 ***
I(sqrt(cpt_bid)) -0.061239   0.115333  -0.531 0.600122
I(sqrt(taps))   -0.264516   0.027312  -9.685 6.09e-10 ***
I(sqrt(spend))   0.431003   0.048950   8.805 3.92e-09 ***
installs        -0.005368   0.009161  -0.586 0.563183
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.05862 on 25 degrees of freedom
Multiple R-squared:  0.8163,    Adjusted R-squared:  0.7869
F-statistic: 27.77 on 4 and 25 DF,  p-value: 7.095e-09
```

```
> model4 <- lm(avg_cpt ~ I(sqrt(cpt_bid)) + I(sqrt(taps)) + I(sqrt(spend)), data = keyword_ZZ1098)
> summary(model4)

Call:
lm(formula = avg_cpt ~ I(sqrt(cpt_bid)) + I(sqrt(taps)) + I(sqrt(spend)),
    data = keyword_ZZ1098)

Residuals:
    Min       1Q   Median       3Q      Max
-0.141807 -0.022465 -0.004412  0.016486  0.174417

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   0.48724   0.10107   4.821 5.39e-05 ***
I(sqrt(cpt_bid)) -0.06409   0.11377  -0.563  0.578
I(sqrt(taps))   -0.26654   0.02675  -9.965 2.28e-10 ***
I(sqrt(spend))   0.41413   0.03908  10.596 6.26e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.05787 on 26 degrees of freedom
Multiple R-squared:  0.8138,    Adjusted R-squared:  0.7923
F-statistic: 37.87 on 3 and 26 DF,  p-value: 1.233e-09
```

9.

```
forecast <- data.frame(cpt_bid = mean(cpt_bid)-0.01, taps = mean(taps), spend = mean(spend))
forecast
new_avg_cpt <- predict(model4, newdata = forecast)
new_avg_cpt
diff <- avg_cpt - new_avg_cpt
diff
```

За допомогою четвертої моделі з 8 пункту побудуємо прогнозування як і наскільки зміниться ціна за клік, якщо зменшити ставку.

Результат коду:

```
> forecast <- data.frame(cpt_bid = mean(cpt_bid)-0.01, taps = mean(taps), spend = mean(spend))
> forecast
  cpt_bid    taps    spend
1 1.215772 5.324074 4.999475
> new_avg_cpt <- predict(model4, newdata = forecast)
> new_avg_cpt
1
0.727527
> diff <- avg_cpt - new_avg_cpt
```

Різниця:

[1]	0.132472952	-0.597527048	-0.587527048	-0.587527048	0.152472952	-0.032527048	-0.447527048	-0.007527048	-0.627527048	-0.093227048	-0.597527048
[12]	0.372472952	-0.587527048	0.172472952	-0.030427048	0.118072952	-0.427527048	0.012472952	-0.044227048	0.072472952	0.002472952	0.031972952
[23]	-0.217527048	0.161072952	-0.327527048	-0.267527048	0.482472952	-0.176827048	0.172472952	-0.147527048	0.057472952	0.332472952	-0.597527048
[34]	0.669172952	0.182472952	-0.627527048	0.389672952	-0.004227048	-0.191227048	0.129972952	0.518072952	0.696872952	-0.597527048	0.452472952
[45]	0.262472952	-0.597527048	-0.267527048	0.481372952	0.169172952	-0.087527048	-0.537527048	-0.327527048	0.219472952	-0.472472952	0.382472952
[56]	0.252472952	-0.487527048	0.278772952	-0.487527048	-0.117527048	0.530272952	0.178172952	0.152472952	-0.507527048	0.327472952	0.302472952
[67]	0.117472952	-0.327527048	-0.112527048	0.509372952	0.493772952	0.452472952	-0.597527048	-0.597527048	0.015772952	0.392472952	-0.187527048
[78]	-0.597527048	0.550772952	0.052472952	0.262472952	0.059972952	0.152472952	0.222472952	0.222472952	-0.397527048	-0.597527048	0.272472952
[89]	-0.195827048	-0.407527048	0.583772952	-0.024227048	0.258172952	0.118472952	-0.597527048	0.256972952	-0.147527048	0.277472952	0.042472952
[100]	-0.057527048	0.507172952	0.422472952	-0.101527048	0.221472952	0.082472952	-0.327527048	0.312472952	0.315772952	0.672472952	-0.627527048
[111]	0.002472952	0.049172952	-0.117527048	-0.597527048	0.027472952	0.448472952	0.184272952	1.122472952	0.422472952	0.327472952	0.772472952
[122]	-0.212527048	0.192472952	0.034972952	-0.157527048	0.217472952	0.072472952	-0.337527048	0.061072952	0.005472952	-0.077527048	-0.380827048
[133]	0.433572952	-0.500827048	0.330772952	0.142472952	-0.086427048	0.272472952	-0.040827048	0.037472952	0.112472952	-0.212527048	0.603272952
[144]	0.184172952	0.452472952	1.032472952	-0.312527048	-0.325027048	-0.257527048	0.002472952	-0.168627048	-0.597527048	-0.227527048	-0.597527048
[155]	-0.037527048	-0.487527048	-0.430027048	0.042472952	0.012472952	0.328872952	-0.487527048	0.172472952	-0.547527048	0.094972952	0.238472952
[166]	0.148772952	0.042472952	-0.467527048	0.245772952	0.161072952	0.252472952	-0.002527048	0.212472952	0.6127527048	0.185172952	0.477772952
[177]	-0.222527048	0.472472952	0.557472952	0.121572952	0.232472952	0.132472952	0.192472952	-0.017527048	1.012472952	0.271072952	0.162472952
[188]	-0.597527048	0.362472952	0.392472952	-0.407527048	0.599272952	0.467472952	0.052472952	0.073272952	0.265772952	0.615172952	-0.627527048
[199]	-0.327527048	0.084972952	0.182472952	-0.597527048	0.266772952	0.604972952	0.457472952	0.217472952	0.412472952	0.202472952	0.469172952
[210]	0.132472952	-0.502527048	0.502472952	-0.132472952	-0.067527048	0.097472952	0.070472952	0.007472952	0.102472952	0.575572952	0.272472952
[221]	0.334272952	0.137472952	0.817472952	0.434472952	0.353272952	0.067872952	0.232472952	-0.337527048	0.212472952	-0.597527048	-0.030827048
[232]	0.542472952	-0.062527048	0.262472952	0.072472952	-0.267527048	0.272472952	-0.387527048	0.208872952	0.092472952	0.397472952	0.406272952
[243]	-0.287527048	0.147472952	0.473872952	0.142472952	0.012472952	0.002472952	-0.597527048	0.459172952	0.162472952	-0.137527048	0.248472952
[254]	0.012472952	-0.397527048	-0.157527048	0.485772952	0.277472952	0.927472952	0.362472952	0.109172952	-0.627527048	0.336072952	-0.047527048
[265]	0.242472952	0.007472952	0.212472952	0.569372952	0.302472952	0.302472952	-0.597527048	0.082472952	0.383072952	-0.230227048	-0.597527048
[276]	0.062472952	0.354972952	0.029972952	0.117472952	-0.058727048	-0.077527048	-0.452527048	-0.400427048	-0.287527048	-0.357527048	-0.497527048
[287]	-0.597527048	-0.287527048	-0.597527048	-0.380827048	-0.435827048	-0.237527048	-0.597527048	-0.227527048	-0.597527048	-0.597527048	0.318472952
[298]	0.122472952	-0.110227048	0.017472952	0.811072952	0.342472952	0.383872952	-0.115927048	0.122472952	0.102472952	-0.107527048	-0.57527048
[309]	-0.627527048	-0.387527048	0.250672952	0.262472952	-0.143527048	-0.227527048	0.201272952	0.082472952	0.234172952	1.052472952	-0.117527048
[320]	0.391672952	0.052472952	0.222472952	0.202472952	0.312472952	1.052472952	-0.106627048	0.072472952	-0.597527048	-0.030827048	-0.152527048
[331]	-0.299527048	-0.397527048	-0.547527048	0.065772952	-0.317527048	-0.217527048	-0.262527048	0.083472952	-0.287527048	-0.047527048	0.299172952
[342]	0.030772952	0.017472952	-0.295027048	-0.257527048	-0.477527048	0.072472952	-0.132472952	-0.597527048	-0.097527048	0.262472952	0.339972952
[353]	0.222472952	0.693772952	-0.042527048	0.091072952	-0.473527048	-0.132527048	0.272472952	0.633772952	0.002472952	0.152472952	0.536172952
[364]	0.132472952	-0.267527048	0.227472952	-0.607527048	-0.627527048	0.507672952	0.424472952	0.082527048	0.222472952	-0.387527048	0.229172952
[375]	0.289172952	0.282472952	0.039172952	-0.417527048	0.398572952	0.132472952	0.032472952	0.442472952	0.152472952	-0.487527048	0.372472952
[386]	0.162472952	-0.020027048	-0.352527048	0.352472952	-0.227527048	0.077472952	0.222472952	-0.241527048	-0.467527048	-0.447527048	0.216272952
[397]	0.002472952	0.673272952	0.197472952	-0.627527048	-0.407527048	0.220772952	0.372472952	-0.157527048	0.272472952	0.385772952	-0.033227048
[408]	0.097472952	-0.140527048	-0.157527048	-0.477527048	0.724172952	0.275772952	0.002472952	-0.627527048	-0.127527048	0.194472952	0.120772952
[419]	0.138472952	-0.242527048	-0.096827048	0.212472952	0.427472952	0.154972952	0.095772952	0.642472952	-0.097527048	-0.017527048	-0.017527048
[430]	0.062472952	0.382472952	0.032472952	-0.482527048	0.009172952	-0.203527048	-0.012527048	-0.297527048	-0.317527048	0.472472952	0.002472952
[441]	0.607472952	0.092472952	0.014172952	-0.597527048	-0.161227048	0.182472952	-0.152527048	-0.407527048	-0.027527048	0.289972952	-0.597527048
[452]	0.432472952	-0.367527048	0.262472952	0.462472952	0.272472952	0.493572952	-0.252527048	0.054972952	-0.412527048	-0.445027048	-0.153727048
[463]	-0.277527048	-0.250827048	0.062472952	-0.597527048	0.498772952	0.062472952	0.252472952	0.408672952	-0.597527048	0.517672952	-0.597527048
[474]	0.276972952	0.252472952	0.169172952	0.052472952	0.170472952	0.262472952	0.312472952	-0.327527048	0.134172952	0.152472952	0.518372952
[485]	0.164972952	-0.060827048	-0.457527048	0.341072952	0.688572952	0.120772952	-0.627527048	-0.084227048	-0.327527048	0.294172952	-0.267527048
[496]	0.450472952	0.332472952	0.552172952	-0.167527048	0.354972952	0.272472952	-0.597527048	0.199172952	0.132472952	0.360772952	0.272472952
[507]	-0.627527048	0.409972952	0.102472952	0.815472952	0.287472952	0.354972952	0.272472952	0.342472952	0.202472952	0.462472952	-0.597527048
[518]	-0.007527048	0.214672952	-0.237527048	0.323972952	0.189172952	0.252472952	-0.037527048	-0.347527048	0.132472952	-0.327527048	0.442472952
[529]	0.212472952	-0.337527048	0.369172952	0.317472952	-0.597527048	0.217472952	0.117472952	-0.172472952	0.210472952	0.062472952	0.292472952
[540]	0.533572952	0.302472952	0.202472952	0.101672952	0.655772952	0.238472952	-0.187527048	0.358472952	0.172472952	-0.347527048	0.377472952
[551]	0.734072952	0.002472952	0.066872952	0.352472952	0.445772952	0.076772952	0.642472952	-0.497527048	0.322472952	0.304472952	0.676472952
[562]	0.712472952	0.682472952	0.653772952	0.752472952	0.472472952	0.132472952	0.634072952	-0.527527048	-0.597527048	-0.627527048	-0.237527048
[573]	-0.537527048	-0.247527048	-0.597527048	-0.297527048	-0.297527048	-0.277527048	0.639472952	0.307472952	0.152472952	0.207472952	0.837472952
[584]	0.265572952	-0.407527048	0.214972952	-0.050827048	0.685772952	0.272472952	0.397472952	0.381272952	0.372472952	0.389572952	0.286472952
[595]	-0.087527048	0.469172952	-0.627527048	0.152472952	0.252472952	0.601672952	1.012472952	0.322472952	-0.160027048	0.305772952	0.002472952
[606]	0.152472952	0.149972952	-0.037527048	-0.597527048	0.799172952	0.172472952	-0.064227048	0.452472952	0.372472952	-0.025727048	-0.310827048
[617]	0.685172952	-0.003227048	-0.387527048	-0.107527048	-0.467527048	0.699172952	-0.161627048	0.048472952	-0.597527048	0.294872952	0.047472952
[628]	0.242472952	0.100872952	-0.307527048	0.253072952	-0.257527048	0.285772952	-0.542527048	0.321072952	-0.017527048	0.002472952	-0.357527048

Висновок

Отже, в курсовій роботі була розглянута рекламна система Apple Search Ads для розміщення реклами ваших застосунків. Розібрали її структуру, її переваги і недоліки та взагалі чому саме Apple Search Ads є привабливою системою для маркетологів.

Крім того, розглянули основний таргетинг на ключові слова. Які типи існують та як з ними працювати. У ході виконання роботи були знайдені методи оптимізацій ставок, цін та інвестицій.

Результат роботи був викладений додатково в кодї мовою R (Додатки), де була проаналізована робота з ключовими словами та ставками

Список використаної літератури:

1. Вікіпедія – Конверсія (в інтернет-маркетингу) [Електронний ресурс]
[https://uk.wikipedia.org/wiki/Конверсія_\(в_інтернет-маркетингу\)](https://uk.wikipedia.org/wiki/Конверсія_(в_інтернет-маркетингу))
2. Apple Search Ads довідка [Електронний ресурс]
<https://searchads.apple.com/help>
3. Apple_Search_Ads_Handbook. Splitmetrics
4. ASA-Email-Course-1-Apple-Search-Ads-Essentials. Курс від маркетингової компанії Splitmetrics
5. ASA-Email-Course-2-Apple-Search-Ads-Dashboard-Overview. Курс від маркетингової компанії Splitmetrics
6. ASA-Email-Course-3-Apple-Search-Ads-Metrics-KPIs-to-Track. Курс від маркетингової компанії Splitmetrics
7. ASA-Email-Course-4-Keyword-Expansion-Apple-Search-Ads. Курс від маркетингової компанії Splitmetrics
8. ASA-Email-Course-5-Adding-Keywords-With-Discovery-Campaigns. Курс від маркетингової компанії Splitmetrics
9. ASA-Course-6-Apple-Search-Ads-Semantics-Based-Account Structure. Курс від маркетингової компанії Splitmetrics
10. ASA-Course-7-Apple-Search-Ads-Value-Based-Account-Structure. Курс від маркетингової компанії Splitmetrics
11. ASA-Email-Course-8-How-to-Optimize-Performance. Курс від маркетингової компанії Splitmetrics
12. ASA-Course-9-Managing-Apple-Search-Ads-with-SearchAdsHQ. Курс від маркетингової компанії Splitmetrics

Додатки

```
asa <- read.csv("asa_dataset_final_11112021.csv")
clear_data = asa[, c(1, 4, 5, 7, 9, 10, 11, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25)]
str(clear_data)
dim(clear_data)
summary(clear_data)
sum(is.na(clear_data))
attach(clear_data)
data <- subset(clear_data, impressions != 0 & spend != 0)
summary(data)
data$keyword_match_type <- as.factor(data$keyword_match_type)
data$date <- as.factor(data$date)
attach(data)

campaign_WW1 <- subset(data, campaign_name == 'WW1')
attach(campaign_WW1)

total_spend = sum(campaign_WW1$spend)
total_installs = sum(campaign_WW1$installs)
total_taps = sum(campaign_WW1$taps)
average_cpa = total_spend/total_installs
average_cpa
average_cpt = total_spend/total_taps
average_cpt
to_negative <- subset(campaign_WW1, (spend>average_cpa & installs<1 & avg_cpa !=
0)||
                        (spend>average_cpt & taps<1 & avg_cpt !=0))

total_spend_data = sum(data$spend)
total_installs_data = sum(data$installs)
total_taps_data = sum(data$taps)
average_cpa_data = total_spend_data/total_installs_data
average_cpa_data
average_cpt_data = total_spend_data/total_taps_data
average_cpt_data
average_cr_data = total_installs/total_taps_data
average_cr_data
```

```

to_negative <- subset(data, (spend>average_cpa_data & installs<1 & avg_cpa !=
0)|| (spend>average_cpt_data & taps<1 & avg_cpt !=0))
to_negative <- subset(data, (installs>0 & avg_cpa > average_cpa_data)|| (taps>0 &
avg_cpt > average_cpt_data))
to_exact <- subset(data, (avg_cpa < average_cpa_data & avg_cpa !=0 & cr >
average_cr_data) &
      (avg_cpt < average_cpt_data & avg_cpt!=0 & cr > average_cr_data) &
keyword_match_type == "BROAD")

keyword_ZZ1098 <- subset(to_exact, keyword == "ZZ1098")
linear_model <- lm(avg_cpt ~ cpt_bid + spend + taps + installs, data = keyword_ZZ1098)
summary(linear_model)

plot(x = spend, y = avg_cpt)
plot(x = taps, y = avg_cpt)
plot(x = installs, y = avg_cpt)
plot(x = cpt_bid, y = avg_cpt)

model1 <- lm(avg_cpt ~ cpt_bid + I(sqrt(spend)) + taps + installs, data =
keyword_ZZ1098)
summary(model1)
model2 <- lm(avg_cpt ~ cpt_bid + I(sqrt(taps)) + I(sqrt(spend)) + installs, data =
keyword_ZZ1098)
summary(model2)
model3 <- lm(avg_cpt ~ I(sqrt(cpt_bid)) + I(sqrt(taps)) + I(sqrt(spend)) + installs, data =
keyword_ZZ1098)
summary(model3)
model4 <- lm(avg_cpt ~ I(sqrt(cpt_bid)) + I(sqrt(taps)) + I(sqrt(spend)), data =
keyword_ZZ1098)
summary(model4)

forecast <- data.frame(cpt_bid = mean(cpt_bid)-0.01, taps = mean(taps), spend =
mean(spend))
forecast
new_avg_cpt <- predict(model4, newdata = forecast)
new_avg_cpt
diff <- avg_cpt - new_avg_cpt
diff

```

```

asa <- read.csv("asa_dataset_final_11112021.csv")
clear_data = asa[, c(1, 4, 5, 7, 9, 10, 11, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25)]
str(clear_data)
dim(clear_data)
summary(clear_data)
sum(is.na(clear_data))
attach(clear_data)
data <- subset(clear_data, impressions != 0 & spend != 0)
summary(data)
data$keyword_match_type <- as.factor(data$keyword_match_type)
data$date <- as.factor(data$date)
attach(data)

campaign_ww1 <- subset(data, campaign_name == 'ww1')
attach(campaign_ww1)

total_spend = sum(campaign_ww1$spend)
total_installs = sum(campaign_ww1$installs)
total_taps = sum(campaign_ww1$taps)
average_cpa = total_spend/total_installs
average_cpa
average_cpt = total_spend/total_taps
average_cpt
to_negative <- subset(campaign_ww1, (spend>average_cpa & installs<1 & avg_cpa != 0)||
                      (spend>average_cpt & taps<1 & avg_cpt !=0))

total_spend_data = sum(data$spend)
total_installs_data = sum(data$installs)
total_taps_data = sum(data$taps)
average_cpa_data = total_spend_data/total_installs_data
average_cpa_data
average_cpt_data = total_spend_data/total_taps_data
average_cpt_data
average_cr_data = total_installs/total_taps_data
average_cr_data

to_negative <- subset(data, (spend>average_cpa_data & installs<1 & avg_cpa != 0)||
                      (spend>average_cpt_data & taps<1 & avg_cpt !=0))
to_negative <- subset(data, (installs>0 & avg_cpa > average_cpa_data)||
                      (taps>0 & avg_cpt > average_cpt_data))

to_exact <- subset(data, (avg_cpa < average_cpa_data & avg_cpa !=0 & cr > average_cr_data) &
                      (avg_cpt < average_cpt_data & avg_cpt!=0 & cr > average_cr_data) & keyword_match_type == "BROAD")

keyword_ZZ1098 <- subset(to_exact, keyword == "ZZ1098")

linear_model <- lm(avg_cpt ~ cpt_bid + spend + taps + installs, data = keyword_ZZ1098)
summary(linear_model)

plot(x = spend, y = avg_cpt)
plot(x = taps, y = avg_cpt)
plot(x = installs, y = avg_cpt)
plot(x = cpt_bid, y = avg_cpt)

model1 <- lm(avg_cpt ~ cpt_bid + I(sqrt(spend)) + taps + installs, data = keyword_ZZ1098)
summary(model1)

model2 <- lm(avg_cpt ~ cpt_bid + I(sqrt(taps)) + I(sqrt(spend)) + installs, data = keyword_ZZ1098)
summary(model2)

model3 <- lm(avg_cpt ~ I(sqrt(cpt_bid)) + I(sqrt(taps)) + I(sqrt(spend)) + installs, data = keyword_ZZ1098)
summary(model3)

model4 <- lm(avg_cpt ~ I(sqrt(cpt_bid)) + I(sqrt(taps)) + I(sqrt(spend)), data = keyword_ZZ1098)
summary(model4)

forecast <- data.frame(cpt_bid = mean(cpt_bid)-0.01, taps = mean(taps), spend = mean(spend))
forecast
new_avg_cpt <- predict(model4, newdata = forecast)
new_avg_cpt
diff <- avg_cpt - new_avg_cpt
diff

```