Был проведен первичный анализ на основе первичного датасета "pets\_dataset\_2025.csv" (в папке "Test\_Dataset") по следующим поставленным аналитическим задачам:

1. Анализ ситуации в регионах.

1.1. Списки регионов, ранжированные по общему числу поступивших заявок, по числу найденных животных/хозяев, процентному соотношению находок к общему числу заявок (коэффициент успеха региона).

1.2. Построение прогнозной модели нагрузки (количества заявок) по регионам на основе временных рядов (исторические данные + сезонность + праздничные дни). Это поможет волонтерским организациям готовиться к периодам пиковой активности.

2. Анализ поведения людей на сайте.

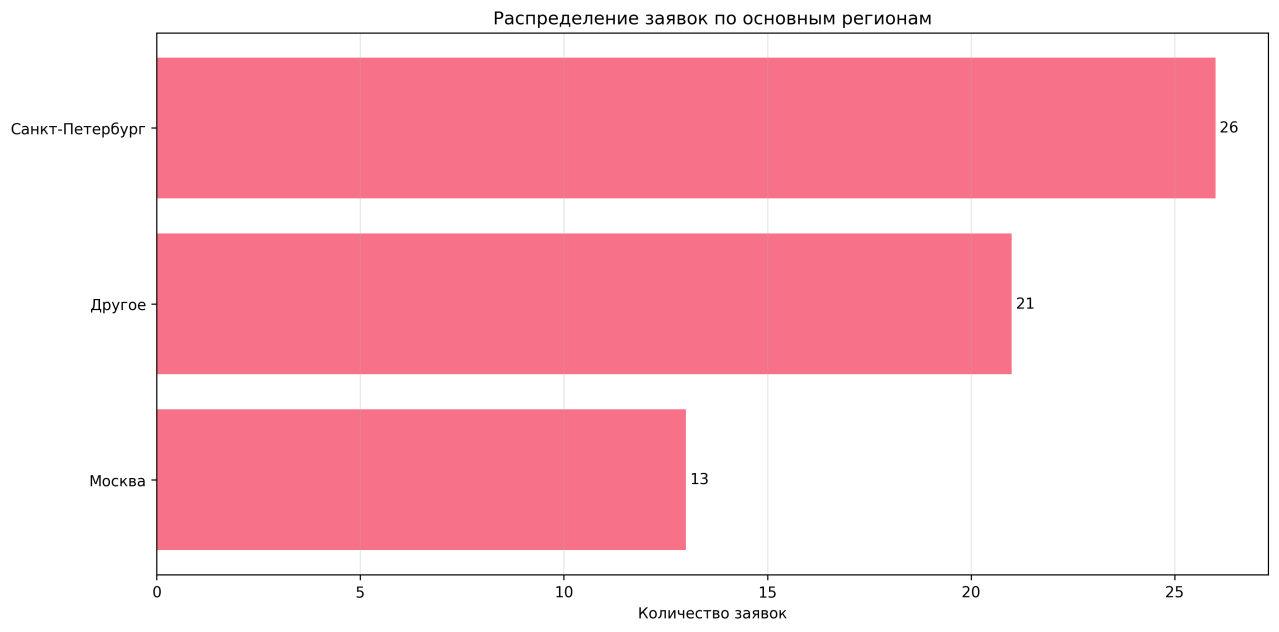
2.1. Анализ количества\_комментариев (активность) пользователей, корреляции между их количеством и вероятностью успешного поиска (статус\_поиска - найден). Представить как пнг файл с таблице или диаграммой.

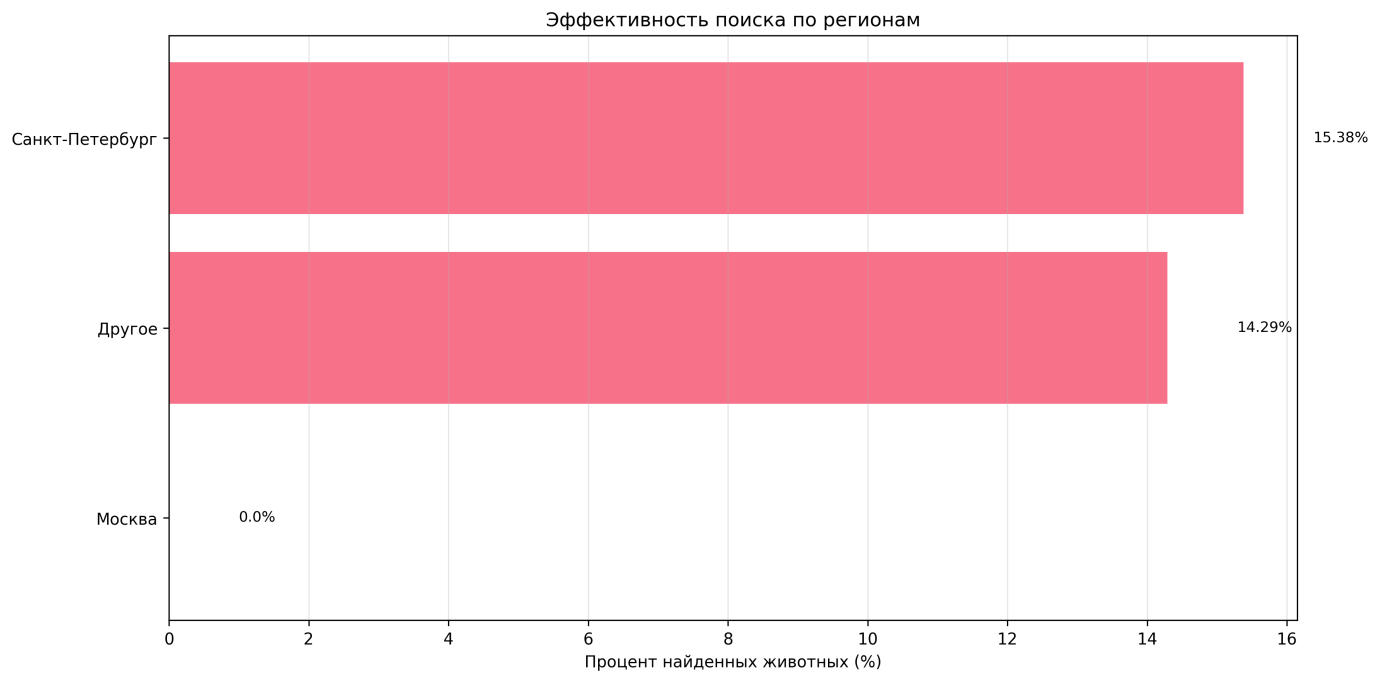
2.2. Анализ того, какие типы публикаций (количество\_фото, длина\_описания) имеют наибольший успех.

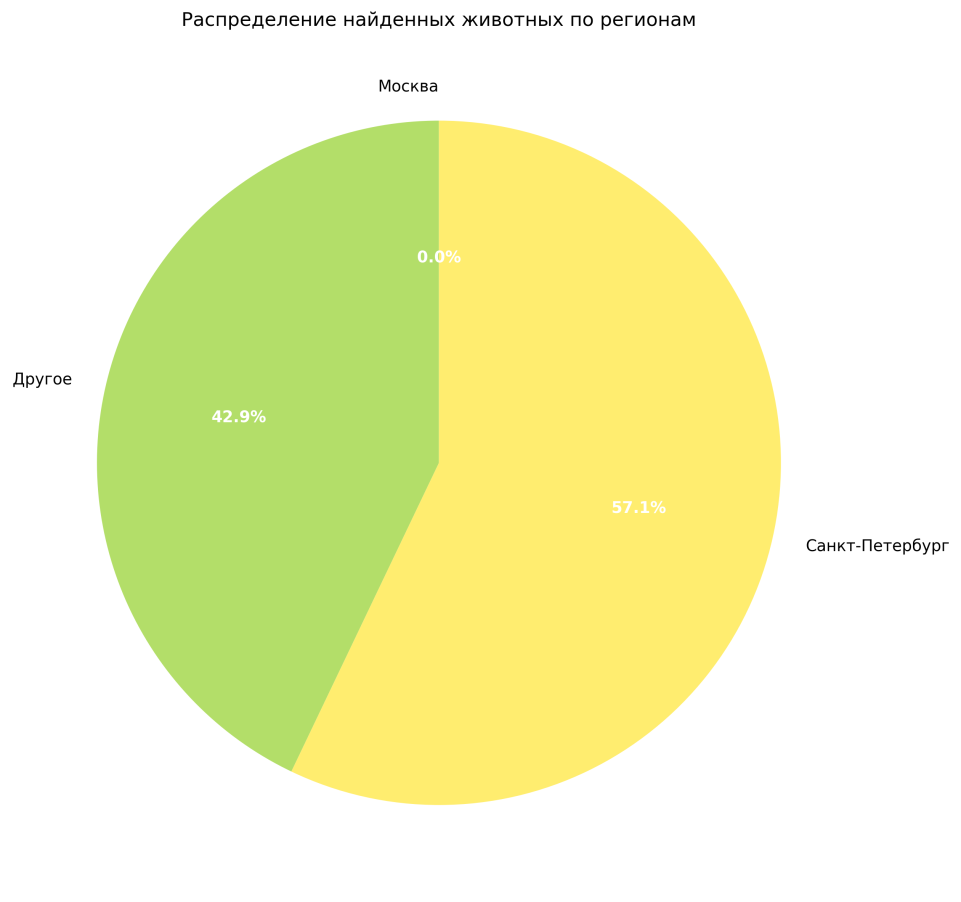
Для выполнения задачи 1.1. был написан скрипт "analysis\_for\_1.1.py" – а**нализ региональной статистики. Его основные функции:**

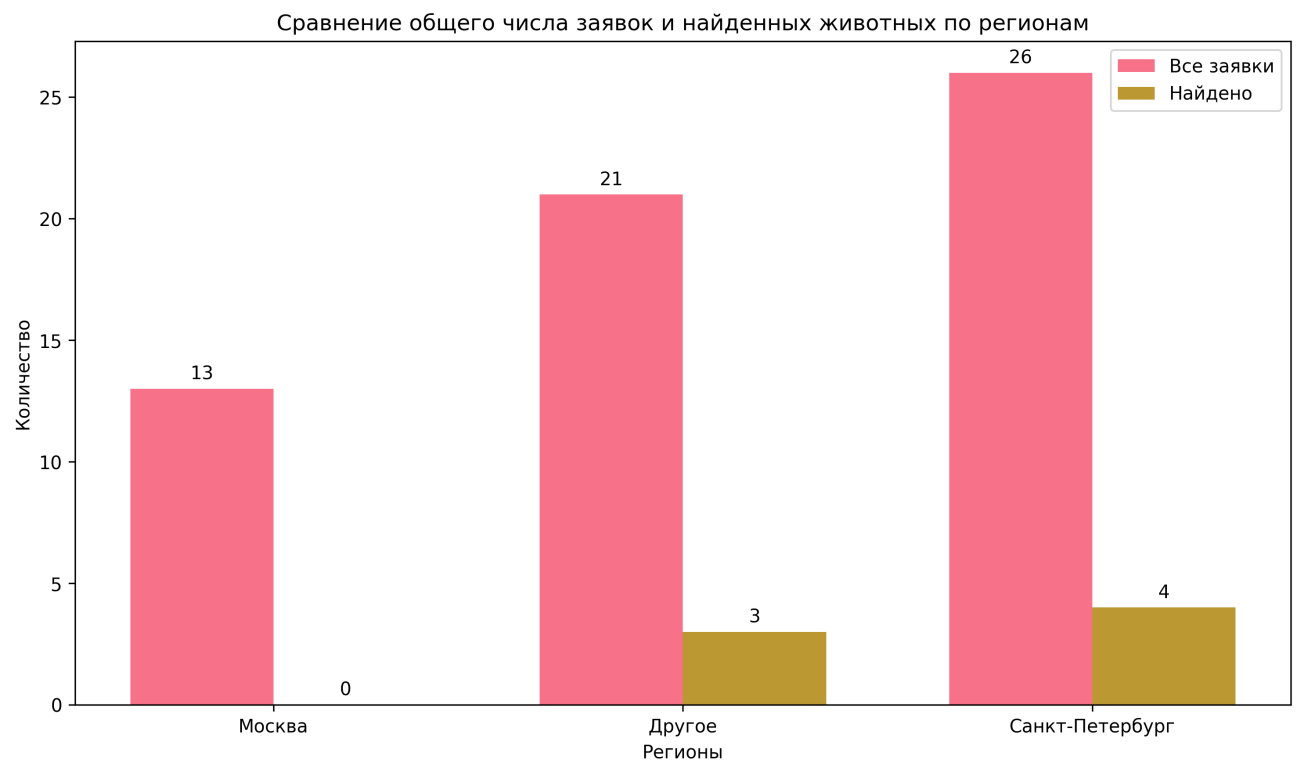
1. **Группировка регионов** - объединяет все регионы кроме Москвы и Санкт-Петербурга в категорию "Другое"
2. **Статистический анализ** - рассчитывает по регионам:
   * Общее количество заявок
   * Количество найденных животных
   * Процент успешных находок
3. **Визуализация данных** - создает 5 типов графиков:
   * Гистограмма распределения заявок по регионам
   * Круговая диаграмма найденных животных
   * График эффективности поиска по регионам
   * Сравнительная столбчатая диаграмма
   * Таблица топ-10 регионов по эффективности (показывает ситуацию подробнее, чем при использовании термина «Другое»)

Скрипт выводит следующие диаграммы и сохраняет их в формате png:







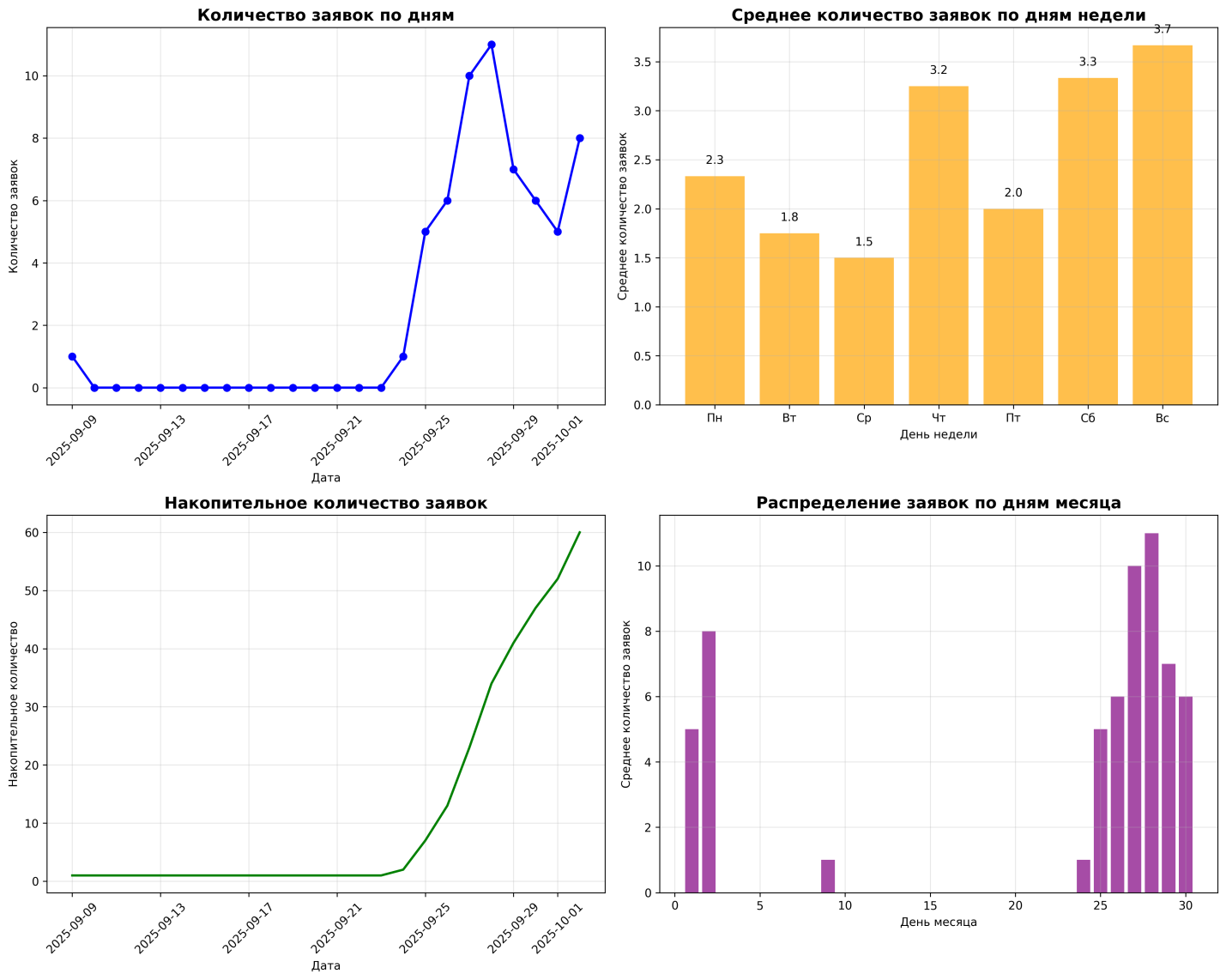




Для выполнения задачи 1.2. был написан скрипт "analysis\_for\_1.2.py". Не совсем понятно по заданию, какого рода прогнозная модель нужна. С помощью использования нейросети было сделано примерное видение реализации этой задачи – анализ временных рядов и прогнозирование нагрузки. Его основные функции:

1. **Анализ сезонности** - изучает распределение заявок по:
   * Дням недели
   * Дням месяца
   * Временным трендам
2. **Прогнозирование** - строит прогноз на 14 дней вперед на основе:
   * Исторических данных
   * Сезонных паттернов
   * Средних значений по дням недели
3. **Рекомендации** - генерирует практические советы для волонтеров:
   * Пиковые дни нагрузки
   * Необходимый запас ресурсов
   * Прогноз на следующую неделю

Скрипт выводит следующие диаграммы и сохраняет их в формате png:



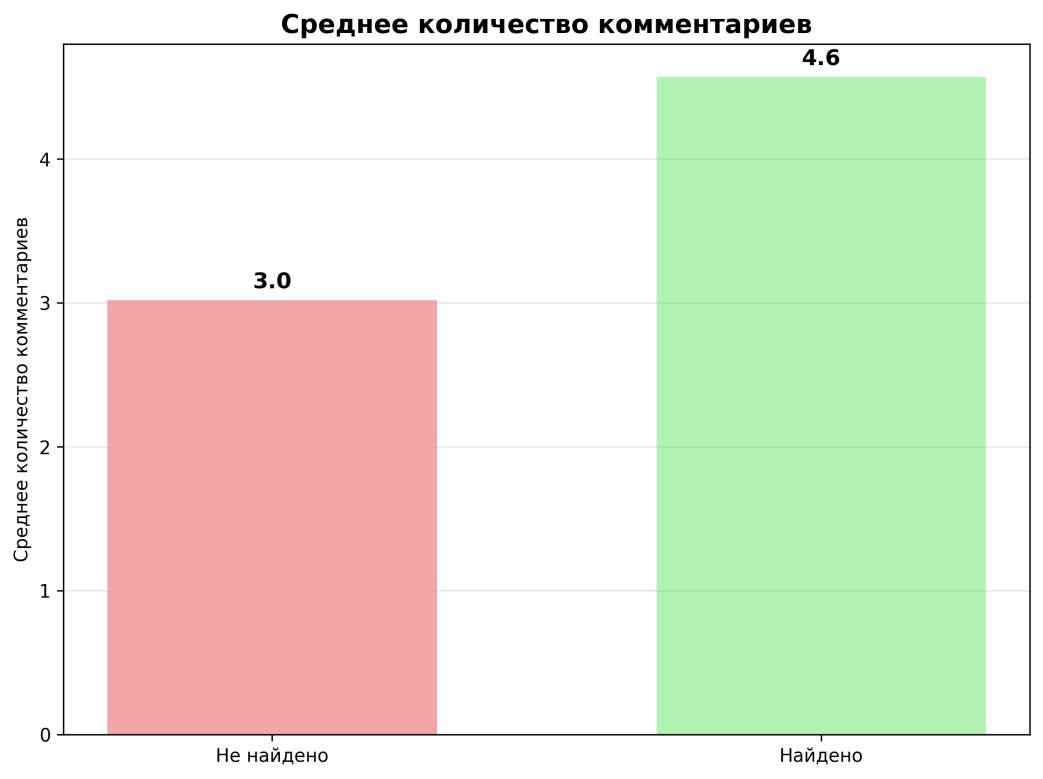


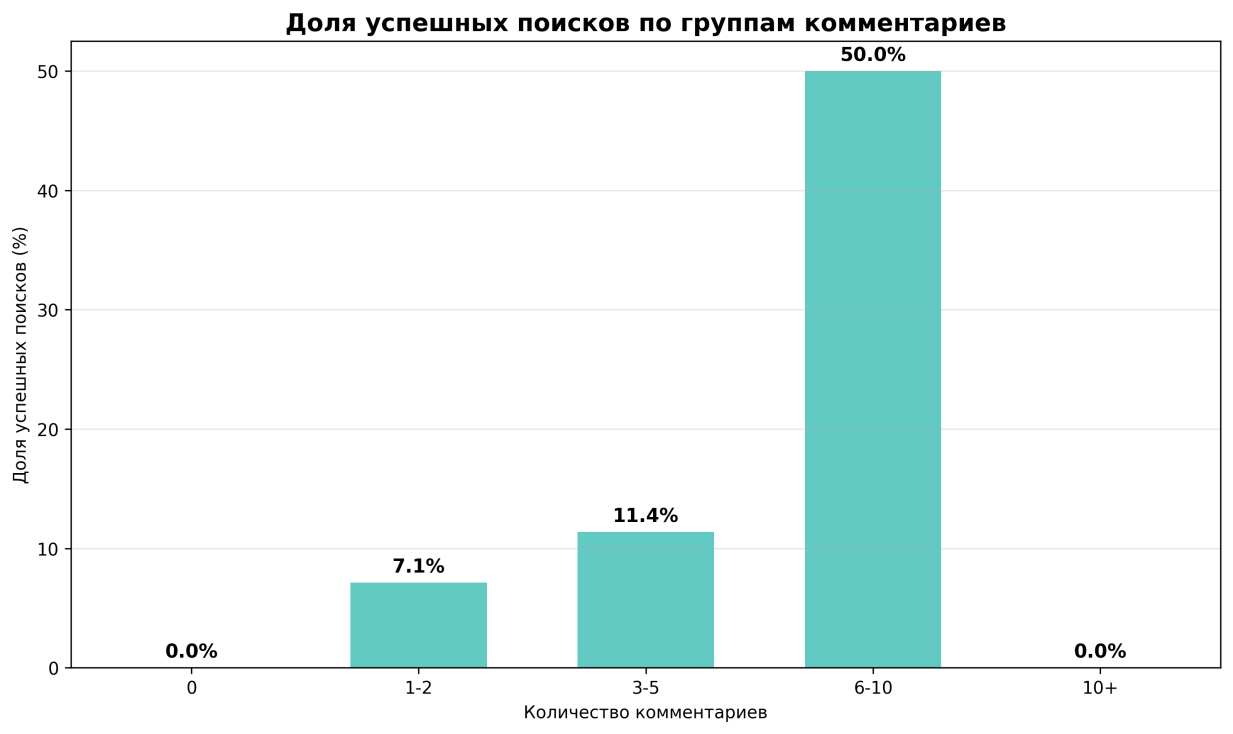


Для выполнения задачи 2.1. был написан скрипт "analysis\_for\_2.1.py" – анализ комментариев и успешности поиска. Его основные функции:

1. **Корреляционный анализ** - вычисляет корреляцию между количеством комментариев и успешностью поиска
2. **Статистические тесты** - проводит t-тест для проверки значимости различий между группами
3. **Анализ по группам** - разделяет объявления на группы по количеству комментариев (0, 1-2, 3-5, 6-10, 10+)
4. **Визуализация данных** - создает 2 типа графиков:
   * Столбчатая диаграмма среднего количества комментариев для найденных/ненайденных животных
   * Столбчатая диаграмма доли успешных поисков по группам комментариев (проценты показывают долю успешных объявлений внутри каждой группы)

Скрипт выводит следующие диаграммы и сохраняет их в формате png:





Для выполнения задачи 2.2. был написан скрипт "analysis\_for\_2.2.py" – анализ влияния характеристик публикации на успешность. Его основные функции:

1. **Анализ наличия фото** - сравнивает успешность поиска для объявлений с фото и без фото
2. **Анализ количества фото** - исследует влияние количества фотографий на успешность поиска
3. **Анализ длины описания** - изучает связь между длиной описания (в словах) и успешностью поиска
4. **Комбинированный анализ** - оценивает совместное влияние фото и описания на успешность
5. **Визуализация данных** - создает 4 типа графиков (проценты показывают долю успешных объявлений внутри каждой группы):
   * Успешность поиска по наличию фото
   * Успешность поиска по количеству фото
   * Успешность поиска по длине описания
   * Успешность поиска по комбинации факторов (фото + описание)

Скрипт выводит следующие диаграммы и сохраняет их в формате png:

