

Д. В. Назаров

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **Д. Л. Коваленко**, канд. физ.-мат. наук, доцент

РАЗРАБОТКА СЕРВИСА ДЛЯ ПОВЫШЕНИЯ БЕЗОПАСНОСТИ ПЕШЕХОДОВ С ИСПОЛЬЗОВАНИЕМ АНАЛИЗА И ОБОГАЩЕНИЯ ДАННЫХ OPENSTREETMAP

Разработка данного приложения была основана на идее, которая звучала так: «Большинство современных детей проводят относительно много времени в процессе пользования смартфонами, и на улице, порой, отвлекаются на свои устройства, не обращая внимания на окружающую обстановку, которая часто предполагает от пешеходов внимательность. Необходимо программными средствами отвлекать детей от смартфонов в опасных местах, что снизило бы количество аварий с участием детей».

В процессе развития приведенной выше идеи, появилась основная концепция работы разрабатываемого приложения – отвлечь детей от телефонов в опасных зонах местности. Так как у университета налажено взаимодействие с органами автоинспекции, было решено работать с объектами, связанными с дорогами, такими как перекрестки и опасные участки дорог, пешеходные переходы.

Так как нам необходимо отвлечь детей силами только средствами смартфонов, необходимы средства, которые есть во всех устройствах, в любом ценовом сегменте и с любыми аксессуарами, связанными устройствами. Из-за данного фактора, оптимальными средствами будут вибрация и звук.

Наиболее распространенной платформой по количеству устройств в Беларуси, является платформа android, и данная платформа отлично подходит под наши цели.

Чтобы уведомлять детей при нахождении устройства рядом с перекрестком, необходимо иметь координаты устройства и координаты перекрестков в конкретном районе.

Если пользователь попадает в местность, где перекрестков достаточно много, а перемещение его осуществляется достаточно быстро, то в случае недостаточного количества координат на устройстве, он может легко выйти за пределы зоны, координаты перекрестков которой используются в данный момент.

Другой стороной данной проблемы является то, что координаты на устройстве могут занимать большое количество памяти, а также

заставлять устройство тратить больше времени и вычислительной мощности на обсчет расстояния до каждой из координат, что может спровоцировать быструю разрядку батареи, и нехватку вычислительного времени.

Исходя из вышеописанных фактов, было принято решение использовать для приложения клиент-серверную архитектуру, заставляя устройство получать список только близлежащих координат с сервера, и сохранять на устройстве пользователя некоторое количество координат для ближайших перекрестков, а также использовать службу, которая будет работать без открытого приложения, считать расстояние до перекрестков, и уведомлять пользователя.

Для того чтобы разработать серверную часть приложения, необходимо решить две задачи – разработать алгоритм, который поможет получить координаты перекрестков для заданной местности, и непосредственно веб-сервер, который будет отвечать на запросы клиента – мобильного приложения. Вариант с ручной разметкой перекрестков не рассматривался, так как это заняло бы очень много времени и ресурсов, из-за огромного числа перекрестков.

Было испытано 2 метода автоматической разметки.

Первый метод – создать датасет из скриншотов карт, на которых необходимо будет разметить перекрестки, после чего обучить сверточную нейросеть для распознавания объектов, которая будет распознавать перекрестки, после чего необходимо будет создать набор скриншотов карт, которые и будут использоваться для разметки с помощью нейросети.

Однако, нам потребуются большие вычислительные ресурсы, а также время для разметки тренировочного датасета.

Наконец, чтобы распознавать перекресток, требуется сделать скриншот карты, и тут мы опять сталкиваемся с проблемой из предыдущего абзаца, и даже в случае получения скриншота, появляется ещё одна проблема – требуется трансформировать координаты с каждого скриншота таким образом, чтобы координаты отсчета у найденного перекрестка соответствовали ширине и долготе на карте, потому как в ином случае мы не сможем искать эти перекрестки на карте, а так же каким-то образом их использовать.

Исходя из текущих фактов, можно признать, что этот метод может иметь успех, однако имеет достаточно «подводных камней» и сложностей в работе, которые могут помешать его реализации, и что заставляет нас попробовать другой метод.

Второй метод заключается в том, чтобы получить карту в том формате, который можно обработать некоторым способом, и получить элементы карты, путем обработки и комбинации которых можно получить необходимые данные.

Для данного метода отлично подходит система от OpenStreetMap. Данный сервис даёт нам возможность экспортировать карты в формате GPX, который по сути является измененным xml.

Вся карта представлена в виде набора тэгов, которые обозначают определенные объекты, и четко закреплены. Есть очень разные типы объектов, однако наш интерес вызывает тэг `<way>`, которым помечаются дороги. Каждый объект на карте является последовательностью точек, которые помечаются тэгом `<ref>`. Отдельно от дорог точки тоже представлены под тэгом `<node>`. У тэгов `<node>` нет координат, однако они включают в себя идентификаторы точек. Если дороги пересекаются, то существуют точки, общие для больше, чем двух дорог. Таким образом, первичный алгоритм работает так, что нам нужно найти общие точки дорог, сохранить их идентификаторы, а потом по идентификаторам найти точки с координатами, которые и будут координатам пересечения дорог, то есть перекрестков. Данный алгоритм не дал нужной точности, однако позволил сказать, что его можно использовать для данной задачи.

После изучения распределения элементов на карте, было выяснено, что если дорога не обозначена тэгом `<highway>`, то, с большой вероятностью, она для решения данной задачи незначительна. Также было решено добавить в алгоритм такие понятия как автомобильные развязки, которые тоже можно посчитать перекрестками и тем самым повысить общую точность распознавания.

Чтобы улучшить точность, можно предположить, что в перекрестках улиц, они каким-то образом объединяются, возможно исходят друг из друга, или наоборот сливаются – такой тип взаимодействия в OpenStreetMap называется relations, и также обозначается на карте, а значит может быть использован нами.

Также на многих перекрестках имеются светофоры, что также поможет улучшить точность, хоть и даст некоторый шум для светофоров вне перекрестков, что, однако, не будет лишним в задаче поиска опасных мест.

Эти 2 списка тэгов также имеют точки, по которым можно выяснить их местоположение, добавление которых также улучшило качество.

Из-за большого количества координат, было решено для начала запустить сервис в областных городах, с перспективой охватить всю Беларусь.

Для получения координат из определенной области карты был написан скрипт на python, который и предоставил нам координаты в json-формате, которые отправляются на устройство веб-сервером, использующим Flask.

В качестве клиента было реализовано приложение для android, которое в foreground service, независимо от того работает ли приложение на экране, или скрыто, как минимум каждые 2 секунды проверяет местоположение пользователя с помощью подписки на locationListener, а после проверяет насколько пользователь близко к перекрестку, и, в случае опасности, воспроизводит звук уведомления, выводит push-уведомление и заставляет устройство вибрировать 5 секунд.

На главном экране приложения пользователь может увидеть своё местоположение на карте, где опасные зоны помечены яркими маркерами.

Таким образом, был разработан программный комплекс, который будет запущен во всех областных городах, и, в перспективе, поможет снизить аварийность.

А. С. Прохоренко

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Леванцов**, ст. преподаватель

ПОДПИСОЧНАЯ МОДЕЛЬ МОНЕТИЗАЦИИ ТОВАРОВ И УСЛУГ

Развитие различных способов монетизаций приложений и сервисов привело к тому, что появился новый его тип – подписочная модель. Можно сказать подписочная модель больше всего была актуальна именно для оплаты агрегаторов контента: музыкальных стриминговых сервисов, площадок с фильмами, книгами, обучающими материалами. Но история подписок началась отнюдь не в цифровом мире: первоначально в ее рамках распространялись периодические печатные издания и другие физические товары.

Данная модель является более привлекательной для любого производителя. Во-первых: со стороны потребителя выглядит так, что продукт можно приобрести, тратя не большую сумму каждый месяц, это