Набор данных для определения пользовательских предпочтений участников дорожного движения на личном транспорте

А.А. Бородинов[1], В.В. Мясников[1,2]

[1]Самарский национальный исследовательский университет имени академика С.П. Королева, Московское шоссе 34А, Самара, Россия, 443086
[2]Институт систем обработки изображений РАН – филиал ФНИЦ «Кристаллография и фотоника» РАН, Молодогвардейская 151, Самара, Россия, 443001

Аннотация. В работе рассматривается задача привязки GPS треков поездок к улично-дорожной сети. Представлен алгоритм привязки точек к конкретному пути на основе динамического программирования. Для проверки предложенного алгоритма были собраны треки поездок нескольких участников дорожного движения на личном транспорте с различными типами поездок. Собранные данные после привязки к улично-дорожной сети могут быть использованы для дальнейшего выявления пользовательских предпочтений и для построения транспортной рекомендательной системы.

1. Введение

Сфера применения рекомендательных систем значительно увеличилась за последние несколько лет. Реклама на интернет ресурсах предлагает пользователям различные товары [1,2], опираясь на историю покупок и просмотр товаров в интернет магазинах, а стриминговые сервисы подбирают фильмы и составляют плейлисты из музыкальных композиций для каждого отдельного пользователя [3]. Для сравнения различных методов машинного обучения, применяемых в рекомендательных системах, для каждой сферы применения были составлены наборы данных. Одним из новых направлений применения рекомендательных систем являются транспортные навигационные системы [4]. Для подобных систем еще не существует общепринятых методов машинного обучения и наборов данных. На данный момент исследователи пытаются применять общедоступные данные о маршрутах передвижения пользователей, такие как OpenStreetMap, Strava или данные о поездках водителей такси [5,6]. Главным недостатком таких данных является невозможность разделить имеющиеся треки по пользователям для выявления их предпочтений в выборе маршрута передвижения. Еще одним недостатком является отсутствие информации о типе поездки. В случае, когда поездка является рабочей или была использована навигационная система с определением кратчайшего пути, полученные пользовательские предпочтения будут недостоверными.

Во втором разделе работы представлены данные о собранных треках поездок на личном транспорте. Алгоритм привязки треков к улично-дорожной сети и результаты его работы описаны в третьем разделе. В завершении работы представлены заключение и возможные направления дальнейших работ и исследований.

2. Сбор данных

Сбор данных осуществлялся в городе Самара на протяжении 6 месяцев с июня по декабрь 2019 года. Девять человек разного пола, возраста, семейного положения и дохода, являющиеся сотрудниками университета, записывали треки своих поездок. Было разделение на рабочие поездки (поездка из дома на работу и с работы домой) в количестве не менее 25 треков и личные поездки (все остальные поездки) в количестве не менее 25 треков. За поездку считается маршрут из точки отправления в целевую точку. Всего было собрано 489 треков. В будние дни было собрано 338 треков и 151 трек записан в выходные. Обобщенные характеристики полученных данных для всех записанных треков представлены в таблице 1.

Таблица 1. Обобщенные характеристики полученных данных.

Характеристика данных	Расстояние поездки	Время поездки
Суммарное значение	4523 км	183 ч 54 мин 20 с
Среднее значение	9249 м	22 мин 33 с
Значение медианы	5783 м	16 мин 35 с
Максимальное значение	74405 м	2 ч 18 мин 50 с
Минимальное значение	1264 м	2 мин 13 с

Сбор данных производится при помощи личных смартфонов пользователей на операционной системе Android и iOS. Подобный метод записи является приближенным к реальному сценарию использования навигационных систем, при котором пользователь использует свой смартфон для получения маршрута движения или информации о загруженности транспортной сети. В таком случае навигационное приложение на смартфоне может записывать данные о передвижении пользователя во время взаимодействия с приложением.

Все записанные треки, наложенные на карту google maps, представлены на рисунках 1 и 2.

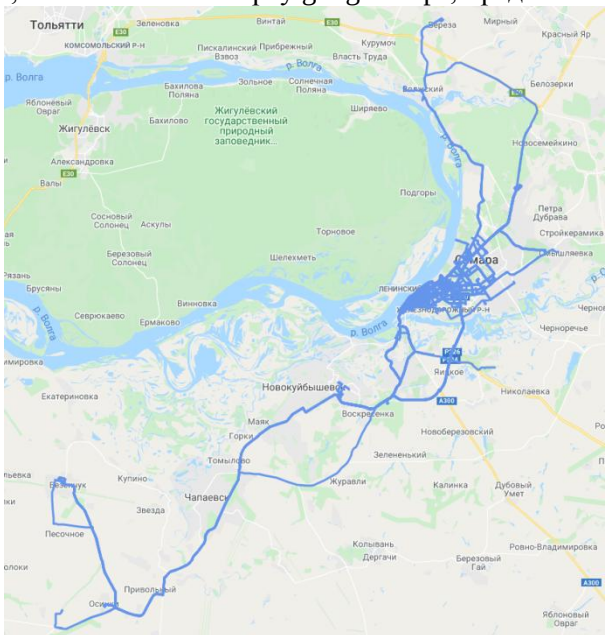

Рисунок 1. Собранные треки на карте большого масштаба.

Рисунки 1 и 2 демонстрируют покрытие значительной части дорожной сети города Самары. В наборе данных присутствуют пользователи, которые на протяжении шести месяцев используют один и тот же маршрут на работу и с работы, как представлено на рисунке 3 а, так и пользователи, которые меняют свои предпочтения и используют различные маршруты для

передвижения в зависимости от загруженности дорожной сети или погодных условий, как показано на рисунке 3 б.

Рисунок 2. Собранные треки на карте малого масштаба.

Рисунок 3. Предпочтения пользователей в выборе маршрута.

3. Алгоритм построения трека по GPS-точкам перемещений

3.1. Входные данные

1) Пусть $\{\overline{x_i}, t_i\}_{i=\overline{0,I-1}}$ - записанные во время поездки данные, где $\overline{x_i} = (x_i, y_i, z_i)$ являются GPS координатами поездки, t_i является временем записи *i*-й координаты маршрута. За время начала записи можно принять $t_0 = 0$.

2) Для работы алгоритма представим улично-дорожную сеть (УДС) в виде ориентированного графа $G = (V, W)$, где V - множество вершин графа, а W - множество ребер графа, соединяющих вершины из V. Вершины имеют координаты $\overline{x_v} = (x_v, y_v, z_v)$ и факт наличия светофора $S(v) = \begin{cases} 0, & нет, \\ 1, & да \end{cases}$. Ребро опишем как $w_{v1,v2} = \begin{cases} \varnothing, & если\ нет\ пути\ из\ v1\ в\ v2, \\ (l^w; \upsilon^w_{\max}; h^w; X^w; c^w), & иначе \end{cases}$, где l^w - длина ребра w, $\upsilon^w_{\max}$ - максимальная допустимая скорость на w, X^w - набор точек,

определяющих ребро w, код кольца УДС $c^w = \begin{cases} 0, & не\ в\ кольце, \\ номер\ кольца\ в\ базе\ УДС, & иначе \end{cases}$. Тип ребра h^w может принимать следующие значения:

$$h^w = \begin{cases} 0 - 1\ полоса\ движения \\ 1 - 2\ полосы\ движения \\ 2 - 3\ полосы\ движения \\ 3 - более\ 3\ полос\ движения\ без\ центральной\ разделительной\ полосы \\ 4 - более\ 2\ полос\ движения\ с\ центральной\ разделительной\ полосой \\ 5 - более\ 4\ полос\ движения\ с\ центральной\ разделительной\ полосой, \\ (скоростная\ дорога\ или\ автомагистраль) \end{cases}.$$

3) Максимально допустимая скорость на графе $\upsilon^0_{\max} = \max\limits_{w \in W} \upsilon^w_{\max}$ и текущие средние скорости υ^w_{avr} для каждого ребра.

3.2. Параметры алгоритма

$\rho_{\min}$ - минимальное расстояние привязки (10 м); γ - фактор увеличения времени; δ - доля увеличения области осмотра (0,2 м); K - число точек в группе (3-5). Параметр α: $\alpha = \left(2\sigma^2\right)^{-1}$, $\sigma = 20$.

3.3. Результат работы алгоритма

Результатом работы алгоритма является множество $\{x_i, t_i, w_i, r_i, \upsilon_i\}_{i=\overline{0,I-1}}$, состоящее из скорректированной последовательности точек $\{x_i, t_i\}_{i=\overline{0,I-1}}$, для любой из которых указано ребро w_i в УДС такое, что $x_i \in X^{w_i}$, и индикатор выбросов r_i такой, что $r_i = \begin{cases} 1, & если\ i\text{-ая точка не выброс}, \\ 0, & если\ i\text{-ая точка выброс} \end{cases}$, а также расчетная скорость в точке υ_i.

3.4. Алгоритм

Шаг 1. Для каждой точки $\bar{x}_i, t_i$ находим ближайшее ребро w и проецируем на него точку следующим образом (ρ евклидово):

$$w_i = \arg\min_{w \in W} \rho(\bar{x}_i, \bar{w}),$$

$$\bar{x}_i = \arg\min_{\bar{x} \in w_i} \rho(\bar{x}, \bar{x}_i).$$

Шаг 2. Последовательно рассматриваем все точки по K штук. Для $K = 3$: $\overline{x_{i-1}}, \overline{x_i}, \overline{x_{i+1}}$, $k = \overline{-1,\ldots,1}$. Если все $\overline{x_{i\pm k}} \in w_i$ & $\rho(\overline{x_{i\pm k}}, \overline{x_{i\pm k}}) < \rho_{\min}$, , то записываем точки сразу в результат $\overline{x_{i\pm k}} := \overline{x_{i\pm k}}$, $w_{i\pm k} := w_{i\pm k}$, $r_{i\pm k} := 1$.

Затем выделяем и рассматриваем все такие последовательности точек, находим минимум и максимум. В случае, когда последовательность нарушается по времени и положению, используем описанный в п.4.4 алгоритм привязки точек к конкретному пути.

После выполнения шага 2 мы получаем привязанные участник пути с разрывами, как показано на рисунке 4. Голубым цветом представлены привязанные точки к соответствующим ребрам графа УДС.

Далее рассматриваем некоторый произвольный фрагмент с i_0 по i_1, т.е. точки $\{\overline{x_{i_0}}, \overline{x_{i_0+1}}, \ldots, \overline{x_{i_1}}\}$

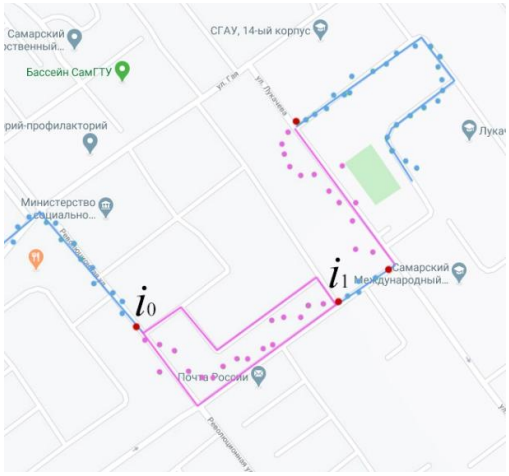

Рисунок 4. Привязанные участки пути.

Шаг 3. Определяем интервал времени для каждой точки из $i_0 \to i_1$ до крайней и определяем физическую возможность появления этой точки. Если $\frac{\rho(\overline{x_i}, \overline{x_{i_0}})}{t_i - t_{i_0}} > \upsilon_{\max}$ или $\frac{\rho(\overline{x_{i_1}}, \overline{x_i})}{t_{i_1} - t_i} > \upsilon_{\max}$, то точка не рассматривается и в дальнейшем считается выбросом $r_i := 0$.

Шаг 4. Определяем подграф от точки i_0 до i_1. Для этого определяем кратчайший путь $i_0 \to i_1$ Далее находим точку $\overline{x}$ в центре кратчайшего пути и строим круг с радиусом $R = (1+\delta) \cdot \max(\rho(\overline{x_i}, \overline{x_{i_0}}), \rho(\overline{x_{i_1}}, \overline{x_i}))$. В подграф относим все вершины, попавшие в этот круг, и соответствующие им ребра.

Шаг 5. Находим все пути без циклов в получившемся подграфе между i_0 и i_1. Обозначим это множество P_{i_0, i_1}, где $\forall\ p \in P_{i_0, i_1} : p = (w_{i_0, i*}; w_{i*,\dots}; \dots; w_{\dots, i_1})$.

Для каждого пути $p \in P_{i_0, i_1}$ используется разработанный *алгоритм привязки точек к конкретному пути* на основе динамического программирования, описанный и формально представленный ниже.

3.5. Алгоритм привязки точек к конкретному пути

Далее для упрощения изложения (но без потери общности) считаем $i_0 = 0$ и $i_1 = I - 1$. В качестве критерия качества привязки используем следующий:

$$J_p = \sum_{i=0}^{I-1} \exp(-\alpha \left\| \overline{x_i} - \overline{x_i^p} \right\|^2).$$

Пусть есть точки $\{\overline{x_i}\}_{i=0}^{I-1}$ $(i_1 - i_0 + 1 = I)$ и они должны быть уложены на путь $p = \{v_{n_0}, v_{n1}, \dots, v_{K-1}\}$, где любая v_n имеет координаты, а v_{n_j} и $v_{n_{j+1}}$ связаны ребром (прямой сегмент, набор отрезков). Дискретизируем возможные положения точек на $p(w_{v_{n_j} v_{n_{j+1}}}) = \{v_{n_0}, v_{n1}, \dots, v_{K-1}\}$ для автомобильного транспортного средства дискретизация $\Delta \approx 2$ м. Пусть итоговое число позиций N, причем $p(0) \sim v_{n_0}$, $p(N-1) \sim v_{n_{K-1}}$. Рассчитаем I массивов характеристик «близость» точки x_i к p как $\varphi_i(n) = \exp(-\alpha \left\| \overline{x_i} - p(n) \right\|^2)$.

Задача: найти последовательность $n(i)_{\overline{i=0, I-1}} : \sum_{i=0}^{I-1} \varphi_i(n(i)) \to \max$, где $n(i) \geq n(i-1)$.

Основное рекуррентное соотношение (для алгоритма динамического программирования):

$$\max_{n(i)} \sum_{i=0}^{I-1} \varphi_i(n(i)) = \max_{n(i_l)=n(i_{l-1}),N} \left\lceil \varphi_i(n(i_l)) + \max_{n(i)\le n(i_l)} \sum_{i=0}^{i_l-1} \varphi_i(n(i)) + \max_{n(i)\ge n(i_l)} \sum_{i=i_l-1}^{I-1} \varphi_i(n(i)) \right\rceil,$$

Обозначим далее $\varphi_j(n) = \max_{n(i):i\le j} \sum_{i=0}^{j} \varphi_i(n(i))$,

$\varphi_i(n)$ - похожесть, $\varphi_i(n)$ - max интегральная похожесть, $\pi_i(n)$ - список положений точек.

Результат содержится в $\pi_0(0)$ и $\varphi_0(0)$.

Алгоритм (начинаем с конца):

for $i = I-1,0$

for $n = \overline{N-1,0}$

if ($i == I-1$)

if $(n == N-1)$

$\varphi_i(N-1) = \varphi_i(N-1)$

$list = new\ List$

$list.add(N-1)$

$\pi_i(N-1) = list$

else

if $\varphi_i(n) > \varphi_i(n+1)$

$list = new\ List$

$list.add(n)$

$\pi_i(n) = list$

$\varphi_i(n) = \varphi_i(n)$

else

$\varphi_i(n) = \varphi_i(n-1)$

$\pi_i(n) = \pi_i(n-1)$

else // ($i == I-1$)

if $(n == N-1)$

$\varphi_i(N-1) = \varphi_i(N-1) + \varphi_{i+1}(N-1)$

$\pi_i(N-1) = \pi_{i-1}(N-1)$

$\pi_i(N-1).add(N-1)$

else

if $\varphi_i(n) + \varphi_{i+1}(n) > \varphi_i(n+1)$

$list = new\ List$

$\pi_i(n) = list$

$list = copy(\pi_{i+1}(n))$

$list.add(n)$

$\varphi_i(n) = \varphi_i(n) + \varphi_{i+1}(n)$

else

$\varphi_i(n) = \varphi_i(n-1)$

$\pi_i(n) = \pi_i(n-1)$

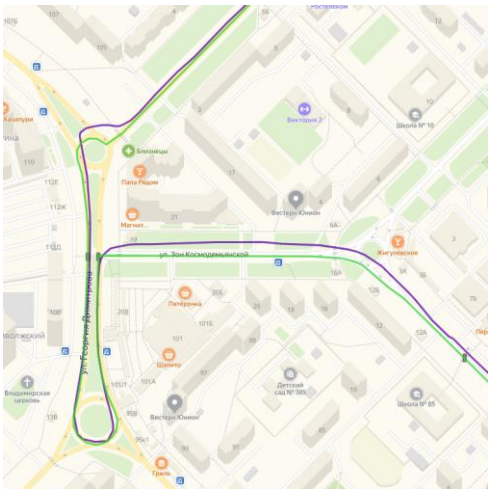

Рисунок 6. Привязанный к УДС трек.

Результат работы алгоритма привязки трека к УДС представлен на рисунке 6. Фиолетовая линия отображает GPS координаты трека, зеленая линия отображает привязку к УДС.

4. Выводы

В работе представлен набор данных, содержащий треки поездок участников дорожного движения на личном транспорте. Также в работе представлен алгоритм привязки GPS треков

поездок к улично-дорожной сети. Результаты работы алгоритма продемонстрированы на УДС города Самары. Дальнейшим направлением исследований является применение полученного набора привязанных к УДС треков в разработке транспортной рекомендательной системе для получения профиля индивидуальных предпочтений пользователей.

5. Благодарности

Работа выполнена при финансовой поддержке Министерства науки и высшего образования РФ (уникальный идентификатор проекта RFMEFI57518X0177).

Авторы выражают особую благодарность сотрудникам кафедры геоинформатики и информационной безопасности Самарского национального исследовательского университета за помощь в сборе треков поездок на личном транспорте.

6. Литература

[1] Joachims, T. Optimizing search engines using clickthrough data // Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2002. – P. 133-142.
[2] He, X. Practical lessons from predicting clicks on ads at Facebook // Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2014.
[3] Koren, Y. Matrix Factorization Techniques for Recommender Systems / Y. Koren, R. Bell, C. Volinsky // Computer. – 2009. – Vol. 42(8). – P. 30-37.
[4] Campigotto, P. Personalized and Situation-Aware Multimodal Route Recommendations: The FAVOUR Algorithm // IEEE Transactions on Intelligent Transportation Systems. – 2017. – Vol. 18(1). – P. 92-102.
[5] Huang, X. Grab-posisi: An extensive real-life GPS trajectory dataset in southeast Asia // Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Prediction of Human Mobility, PredictGIS, 2019. – P. 1-10.
[6] Lian, J. One-month Beijing taxi GPS trajectory dataset with taxi IDS and vehicle status / J. Lian, L. Zhang // DATA Proceedings of the 1st Workshop on Data Acquisition To Analysis, Part of SenSys, 2018. – P. 3-4.

Personal vehicle travel dataset to determine user preferences

A.A. Borodinov[1], V.V. Myasnikov[1,2]

[1]Samara National Research University, Moskovskoe Shosse 34A, Samara, Russia, 443086
[2]Image Processing Systems Institute of RAS - Branch of the FSRC "Crystallography and Photonics" RAS, Molodogvardejskaya street 151, Samara, Russia, 443001

Abstract. The paper considers the task of linking GPS tracks to a road network. The authors presented an algorithm for linking points to a specific path based on dynamic programming. Tracks of trips of several road users on personal vehicles were collected to verify the proposed algorithm. The data collected after binding to the road network can be used to further identify user preferences and to build a transport recommendation system.