

Автоматический гео-контекстный анализ на мобильных платформах

Юсуф-заде Гулам, Намиот Д. Е.

Аннотация—В связи с бурным ростом производительности мобильных устройств значительно расширился и спектр задач, которые могут выполняться на мобильных системах. Одной из основных тенденций стало развитие так называемых контекстно-зависимых вычислений. Концепция контекстно-зависимых вычислений, применительно к мобильным устройствам, исходит из того, что результат работы многих приложений может быть значительно улучшен за счет использования контекста. Под контекстом, в данном случае, могут подразумеваться такие параметры, как время суток, текущее географическое положение, сетевая активность пользователя и т.п. Все перечисленные составляющие могут отслеживаться в режиме реального времени благодаря многочисленным датчикам и сенсорам, встроенным в подавляющее большинство существующих портативных устройств.

Ключевые слова— контекстные вычисления, гео-контекстный анализ, Google Android, Symbian OS, Windows Phone.

I. ВВЕДЕНИЕ

Одними из первых определение контекста предложили Шилит и Теймер в 1994 [1]. Контекст — это информация о местоположении, людях, находящихся рядом, и их изменениях. Ими же было дано определение контекстно-зависимых вычислений как программного обеспечения, использующего контекст. Позже Деем было введено более общее определение контекста [2], как любой существенной информации, характеризующей ситуацию, в которой оказалась сущность. Под сущностью подразумевается человек, место или объект, считающийся важным для взаимодействия с пользователем приложения (конечным пользователем или же другими приложениями).

Контекстно-зависимые приложения, как правило, обладают следующими свойствами:

- Автоматизация. Например, автоматический вход в учетную запись пользователя, при его приближении или автоматическая переадресация звонков с мобильного телефона на городской.

Юсуф-заде Гулам Эльчин оглы – студент 2-го курса магистратуры факультета вычислительной математики и кибернетики Московского Государственного Университета имени М.В.Ломоносова.

Намиот Дмитрий Евгеньевич – кандидат физико-математических наук, старший научный сотрудник лаборатории открытых информационных технологий факультета вычислительной математики и кибернетики Московского Государственного Университета имени М.В.Ломоносова.

- Адаптация. Приложение может менять свое поведение в зависимости от доступного оборудования и особенностей среды, окружающей пользователя.
- Персонализация. Пользователь может изменить, специализировать поведение программы в определенном контексте.

Контекстно-зависимые вычисления ориентированы в первую очередь на конечного пользователя. Подобного рода системы динамически адаптируются к изменяющимся окружающим обстоятельствам и позволяют предоставить информацию, важную именно в конкретный момент [3].

Целью данной статьи является анализ существующих на данный момент работ, посвященных гео-контекстному анализу, а также выделение общих черт и принципов работы.

Помимо основной платформы Windows Phone были выбраны проекты на платформах Symbian OS и Google Android. Также был рассмотрен сервис Foursquare, являющийся ярким примером проекта, ориентированного на работу с гео-контекстными данными.

II. АНАЛИЗ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И СЕРВИСОВ

A. *gMaps*

Одним из наиболее популярных приложений на платформе Windows Phone, посвященных контекстно-зависимому анализу подобного рода, является *gMaps*, разработанный Алексеем Страхом и Александром Сороколетовым.

Программа обладает следующим функциями:

- Гео-кодирование и реверсивное гео-кодирование
- Локальный поиск — возможность найти интересующие заведения, прочитать отзывы о найденном объекте.
- Прокладка маршрута
- Отображение пробок и остановок общественного транспорта.
- Возможность отметить собственное местоположение и сделать его доступным для других пользователей.

Особенностью приложения является то, что вместо стандартных для платформы карт Bing используются

карты Google, которые на данный момент имеют лучшее качество покрытия. Это решение связано с дополнительными трудностями, поскольку имеющиеся программные примитивы приспособлены только для базовых карт и имеют не очень высокую скорость работы со сторонними ресурсами. Однако только благодаря тесной интеграции с сервисом карт Google, стало возможным реализовать функционал недоступный стандартным приложениям.

Помимо этого был обнаружен ряд прочих технических трудностей. Невозможной оказалась реализация offline режима работы. Следует заметить, что в большинстве случаев работа ведется с картами, загружаемыми со стороннего ресурса, что увеличивает объем передаваемых и получаемых данных. Режим, при котором приложение могло бы опираться на карты, уже сохраненные в памяти, мог бы уменьшить подобные накладные расходы. Однако режим крайне непросто в реализации, более того, в большинстве случаев, противоречит пользовательскому соглашению, заключаемому с провайдером сервиса.

Следует заметить, что платформа Windows Phone в ранних версиях поддерживала многозадачность весьма ограниченно. И для работы приложения требовалось, чтобы оно было постоянно запущено и активно. В случае открытия другой программы, приложение прекращало свою активность (отслеживание местоположения).

Приложение работает по так называемой клиент-серверной модели, где в качестве клиента выступает приложение на мобильном устройстве, а в качестве сервера — картографический сервис Google Maps.

B. Nokia Friends View

Подавляющее большинство современных мобильных телефонов имеет возможность выхода в интернет. В приложении Friends View, разработанном в Nokia Beta labs в 2008 году, выход в интернет используется для создания социальной сети, в которой были бы представлены все интересующие пользователя люди.

Программа позволяет пользователю отмечать свое местоположение, публиковать статусы и сообщения. Данные, предоставленные пользователем, таким образом, привязываются к карте и становятся доступными как с мобильного устройства, так и с помощью web-сервиса. Это позволяет оставаться на связи с интересующими людьми, а также знать, где они находятся.

Основа, на которой работает программа, — обновление сообщений и данных о местоположении, иными словами, комбинация микро-блоггинга и гео-сервиса. Пользователи могут загружать обновления относительно своего текущего местопребывания, а также отмечать места, куда собираются пойти в ближайшее время и оповестить об этом круг заинтересованных лиц (друзей, семью). Как и в любом сервисе микро-блоггинга, доступна возможность комментирования публикаций.

Подобные социальные сети, создающиеся посредством мобильных терминалов (телефонов) через общий промежуточный уровень (обычно, веб), называются мобильной социальной сетью. Такие социальные сети имеют большой интерес, как для бизнеса, так и для обычных пользователей, поскольку предоставляют возможность взаимодействия в режиме реального времени. Помимо этого, сервис может быть использован для предоставления рекомендаций, поиска и персонализации на основе данных генерируемых пользователем [5].

Интересной особенностью данного программного обеспечения является именно функция рекомендаций друзей на основе контекстных данных. Похожий сервис уже реализован в социальных сетях, таких как Facebook, Google+ , LinkedIn и обычно базируется на основе анализа профиля пользователя на предмет похожего или совпадающего содержания (совместные интересы, работа в одной и той же организации) и отношений friend-of-a-friend. Эти алгоритмы имеют ряд недостатков и часто страдают от эффекта friend spam и comment spam. Например, TwitRank - алгоритм ранжирования, используемый в Twitter. Он основан на количестве подписчиков и количестве комментариев, что приводит к риску рекомендовать в друзья человека, занимающегося спамом и не представляющего никакого интереса [6].

C. Context Watcher

Группой авторов во главе Йоханосом Кулваем было предложено приложение для мобильных телефонов под названием Context Watcher [7]. Программа позволяет пользователю делиться собственной контекстной информацией: географическим местоположением, частотой сердечных сокращений, скоростью передвижения и т.д. Подобные данные могут быть не только опубликованы, но и использованы в качестве входных данных для информационных сервисов, для адаптации приложений к окружающим условиям или облегчения выполнения каждодневных сценариев.

Приложение было разработано на языке Python для платформы Nokia Series 60. Основной задачей программы является:

- Дать возможность конечным пользователям автоматически записывать, сохранять, распространять и использовать контекстную информацию для облегчения жизнедеятельности. Например, сервисы погоды могут работать самостоятельно, зная текущее местоположение, и предоставлять данные только по нужному региону.
- Сбор контекстных данных с помощью большого количества добровольцев. Полученная таким образом информация может быть использована для создания и тестирования аналитических алгоритмов и обучения аналитических движков, которые из необработанных контекстных данных получают необходимую информацию.

Полученные алгоритмы вновь могут быть протестированы с помощью программы Context Watcher.

- Демонстрация возможностей контекстно-зависимых вычислений в повседневной жизни. Например, автоматическое ведение дневника, загрузка картинок с использованием данных, полученных из контекста.

Доступная информация зависит от поставщиков контекста, к которым программа имеет доступ. Список доступных поставщиков зависит от имеющихся сенсоров, ситуации и пользовательских настроек. Context Watcher имеет модульную архитектуру, что позволяет изменять конфигурацию используемых компонентов. Приложение позволяет записывать следующую информацию:

- географическое местоположение (на основе данных о базовых станциях GSM или GPS координат).
- Настроение и другие субъективные параметры, основанные на данных, вводимых пользователем.
- Встречи, каждодневные сценарии (на базе собранных данных).
- Погоду (по средствам сторонних сервисов).
- Данные о здоровье (пульс, давление и прочие медицинские показатели).
- Графическую информацию (изображения с привязанными контекстными параметрами).

Когда пользователь запускает приложение, оно, в зависимости от конфигурации, автоматически подключается к доступным сенсорам посредством технологии Bluetooth и удаленным контекст-провайдерам через сеть 3G. Все соединения могут быть активированы вручную. Например, если включен модуль GPS Context Watcher, то помимо информации о положении, полученной с базовых станций, он будет отправлять удаленному провайдеру еще и точные координаты. Обработанная информация возвращается пользователю, и он получает уведомления, что находится в известном кластере (дом, офис, отель) или, что какой-либо знакомый из списка друзей находится где-то неподалеку.

Приложение построено на основе MobiLife context management framework – среды, предоставляющей наиболее общий подход к исследованию контекста. В данной среде различные сущности и контекст-провайдеры взаимодействуют, используя сеть интернет.

Основными задачами среды MobiLife являются:

обнаружение контекст-провайдеров (поставщиков);

- стандартизация обмена контекстом между пользователями и провайдерами;
- поддержка различного рода анализа контекста с помощью специальных аналитических компонентов, работающих в режиме plug and play;
- создание различных групп контекстных

провайдеров для предоставления более высокоуровневой информации.

Архитектура среды, как и архитектура всего приложения, — модульная, каждый компонент может быть изменен или заменен на другой.

D. Social recommendation system

Группой исследователей из Roma Tre University был предложена система социальных рекомендаций [8] – программное обеспечение, предназначенное для платформы Google Android, основной задачей которого является рекомендация пользователю так называемых точек интереса (Points Of Interest). Этими точками интереса могут являться различного рода заведения, встречи, увеселительные мероприятия и т. д. Подобный функционал уже реализован в некоторых приложениях. Как правило, в них используются данные из различных социальных сетей (Facebook, Foursquare). Однако в большинстве случаев единственным критерием рекомендации того или иного мероприятия является географическая близость пользователя к месту проведения, а такие параметры, как день недели, время, погода, транспортная загруженность, не учитываются. В данной работе программное обеспечение собирает данные о пользовательских предпочтениях, планах и, исходя из собранной информации, дает рекомендации.

Задачами этого проекта ставились:

- Создание методологии определения пользовательских предпочтений, отличной от существующих систем, основанных только на географическом местоположении.
- Использование постоянно растущего количества информации в социальных сетях, отзывов пользователей и локальных web-ресурсов.
- Разработать процедуры определения контекста для использования при выдаче рекомендаций.

Данное приложение работает с картографическим сервисом Google Maps, хотя в качестве источников данных могут быть использованы и другие ресурсы (Bing Maps, Yahoo! Maps и пр.). В качестве информационных ресурсов предполагалось использовать не только глобальные источники, но и локальные web-ресурсы, поскольку нередко именно они обладают более полной информацией об интересующих объектах.

В качестве тестов были реализован ряд сценариев, учитывающих сколько времени пользователь готов затратить на посещение места, погоды, дорожной ситуации и количества средств, которые можно потратить. При этом помимо основного предположения программа предоставляла также ряда альтернатив.

Для решения поставленной задачи был разработан алгоритм Polar, который:

- моделирует пользовательские предпочтения с целью адаптировать рекомендации к специфическим нуждам;
- использует более детальные контекстные данные для уточнения результатов поиска;

- расширяет потенциала гео-контекстного поиска с помощью использования информации из социальных сетей и прочих интернет - ресурсов.

Алгоритм работы приложения следующий: прежде всего, требуется аутентификация по паре ключ-пароль, после чего приложение получает доступ к профилю пользователя на web-ресурсе. Затем определяется текущее местоположение и демонстрируется карта, на которой присутствует отметка, обозначающая место, где сейчас находится устройство. После чего пользователю предоставляется возможность сделать интересующий запрос (строка текста). Обычно запросы характеризуют категорию точек интереса. Например, “пищерия” или “fast food” совпадает с название категории, относящейся к предприятиям общественного питания. В случае несоответствия, например, при поиске по ключевым словам “спагетти” или “французская кухня” будут выбраны те точки интереса, чьи метаданные сходны с текстом запроса. Ранжирование найденных значений производится на основе пользовательских предпочтений и текущего контекста, после чего наиболее релевантные значения отмечаются на карте и сопровождаются подробным описанием. Приложение также дает возможность оценить точки интереса по шкале от 1 до 5 и поставить собственную пометку (может быть выбрана из отметок, предложенных системой, или дана самим пользователем). Впоследствии полученная информация фиксируется в профиле и может быть использована для уточнения результатов поиска

Приложение собирает данные об окружающем контексте и отправляет их вместе с запросом пользователя. Далее на web-сервисе происходит обработка. Следует заметить, что данные крайне персонифицированы и их утечка третьим сторонам недопустима. Поиск точек интереса по данной макро-категории (класс точек интереса со сходными характеристиками: рестораны, бары, культурные события) тесно связан с текущими потребностями и предпочтениями пользователя.

E. Foursquare

Foursquare - это проект, созданный Деннисом Кроули и Навином Селвадурай в 2009 году, представляющий собой социальную сеть с развитыми функциями гео-позиционирования. В первую очередь проект ориентирован на мобильные устройства и позволяет отмечаться в посещенных местах (check-in), давать отзывы, искать новых друзей и узнавать местоположение уже найденных.

На текущий момент клиент Foursquare существует для всех основных мобильных платформ: Google Android, iOS, Windows Phone 7, Symbian OS. Также благодаря открытому API, существует возможность использования сторонних клиентов или создания собственного. Следует заметить, что функционал социальной сети доступен и через online web-сервис.

Интересной особенностью Foursquare является присвоение пользователям различных значков или званий. Каждый раз, когда пользователь отмечается в каком-либо месте, он зарабатывает виртуальные очки, называемые значками. Существуют специальные значки, которые можно получить, отмечаясь только в определенных городах, или же отмечаясь только в определенных заведениях-партнёрах. Foursquare периодически меняет способы получения значков. Однако полный список значков не публикуется, а механизмы их получения не разглашаются.

Помимо значков, существует еще одно поощрение активных пользователей в виде назначения мэром какого-либо места. Для этого надо в течение последних 60 суток отмечаться в определенном объекте больше дней, чем кто-либо из прочих посетителей (имеется в виду следующий принцип подсчёта отметок: 1 день — 1 отметка). Если в профиль пользователя была загружена фотография, то он объявляется «Мэром» этого объекта. Мэр сохраняет это звание до того момента, пока количество отметок другого участника не превзойдёт количество отметок мэра. При этом существует возможность быть мэром нескольких мест сразу [9].

Подобные приемы позволяет поддерживать интерес пользователей и подталкивать их к более частому использованию приложения и сервиса.

Большая часть: код сайта, API и пакетные задания - написаны на языке Scala. Web-часть и API созданы в среде веб-приложений Lift. В проекте также используются скриптовые языки Python и Bash для автоматизации операционных задач, процессов разработки и внедрения. Динамическое содержимое сайта написано на языке javascript с применением библиотеки jQuery и Backbone.js для моделей объектов.

В сервисе используются карты от провайдера MapBox, созданные на основе данных открытого картографического проекта OpenStreetMap. Интерактивные карты Foursquare созданы при помощи библиотеки с открытым кодом Leaflet [10].

Сервис Foursquare поддерживает несколько вариантов своего API для различных нужд.

Для получения полного доступа ко всем функциям сервиса можно воспользоваться Core API. Инструментарий позволяет пользователям отмечаться, просматривать историю посещений, видеть друзей, создавать заметки и списки, искать информацию о местах и заведениях и получать доступ к рекомендациям. Взаимодействие происходит по защищенному протоколу HTTPS.

Real-Time API позволяет приложению получать доступ к определенной подгруппе отметок (check-in) в режиме реального времени. Данное API позволяет серверу разработчика получать стандартные HTTP запросы от сервиса, когда происходит намеченное событие.

Торговая платформа (Merchant Platform) foursquare позволяет разработчикам работать с менеджерами заведений для управления информацией и предоставления особых привилегий для пользователей, которые отмечаются в этом месте. Также доступна

аналитика и статистика посещения каких-либо мест или заведений.

Venues Platform позволяет быстро создать информацию о каком-либо месте без глубокой интеграции и аутентификации. Помимо этого, платформа позволяет клиентам, которые уже имеют свои данные о местах, соотнести их базой данных Foursquare [11].

Отметим также, что модель check-ins широко используется в других системах. Например, Facebook или Twitter. При этом, check-ins, как отметки о присутствии, могут быть подвержены кастомизации и их бизнес-применение может даже полностью отделено от гео-позиционирования [12].

Другим примером контекстно-зависимых отметок о присутствии служит система SpotEx [13]. В этой системе вместо географических координат используется информация о доступных точках доступа Wi-Fi. Иными словами, места в такой системе представляются так называемыми Wi-Fi fingerprints – цифровыми снимками сети. И, соответственно, вычисления, основанные на географических координатах, заменяются вычислениями, основанными на сетевой близости (network proximity)

F. Google Now

Одной из наиболее крупных разработок в области контекстного анализа на мобильных платформах является программный продукт компании Google названием Google Now. Данное приложение представляет собой персональный ассистент пользователя, доступный на платформе Google Android.

Google Now является частью другого приложения Google Search, осуществляющего поиск в сети интернет. Google Now, анализирует ежедневную повторяющуюся деятельность пользователя (нахождение в определенной географической точке или области, календарные события, поисковые запросы и т.п.) и на основе собранной статистики находит наиболее подходящую информацию. Данные потом формируются в специальный формат карточек

Программа активно использует систему Google Knowledge Graph – систему позволяющую уточнить поисковые запросы исходя из анализа значений данных и их связей.

В целом, проект Google Now анализирует весьма большой спектр контекстных данных. На текущий момент приложение находит информацию о следующем:

- Дни рождения
- События
- Данные о полетах
- Кинофильмы
- Новости
- Следующие встречи
- Спортивные результаты
- Трафик
- Точки интереса поблизости

- Погода
- Интересные точки для съемки рядом
- Биржевые сводки
- Дорога домой

Замечу, что это неполный список возможностей.

Google Now использует также систему распознавания голоса Google Voice, которая позволяет осуществлять запросы на естественном языке. На данный момент система поддерживает все основные европейские языки, включая русский, а также значительное количество азиатских языков. Более того, количество языков постоянно увеличивается.

Проект построен по клиент-серверной архитектуре, где в роли клиента выступает непосредственно само приложение, а в роли информационных серверов – сервисы Google.

III. ЗАКЛЮЧЕНИЕ

В данной работе были рассмотрены ряд приложений и сервисов, ориентированных на контекстный анализ. Было определено, что большинство приложений работает по клиент-серверному принципу, перекладывая хранение пользовательских и прочих данных и такие вычисления, как поиск в базах данных, обработка запроса, на удаленные мощности. Такое решение позволяет сэкономить место во встроенном хранилище телефона и уменьшить расход заряда батареи и получить доступ к информации, хранящейся на сервере, однако увеличивает объем получаемых и передаваемых данных. Рассмотренные приложения собирают данные о контексте, пользуясь сенсорами, которые могут быть встроены или подключены дополнительно и передают данные на обработку третьим сторонам, после чего лишь демонстрируют полученные данные.

В будущем полученную информацию планируется использовать для написания собственного гео-контекстного приложения на платформе Windows Phone. В работе будет реализован сервис, позволяющий собирать данные о местоположении друзей и демонстрировать их на экране телефона. Причем список друзей планируется брать из популярных социальных сетей, таких как Facebook, Twitter, Foursquare.

БИБЛИОГРАФИЯ

- [1] B. Schilit and M. Theimer, “Disseminating active map information to mobile hosts,” Network, IEEE, vol. 8, no. 5, 1994.
- [2] K. Dey, “Understanding and using context,” Personal Ubiquitous Computing, vol. 5, January 2001.
- [3] J. Giles, “Context-Aware and Location Systems”, University of Cambridge, January 1998
- [4] C. Schmidt, “Context-Aware Computing”, Berlin Institute of Technology, 2011.
- [5] “Nokia Technology Insights series”, Nokia Research Center, January 2009.
- [6] Chin, “Finding Cohesive Subgroups and Relevant

Membership the Nokia Friend View Mobile Social Network”, Nokia Research Center, 2009.

- [7] J. Koolwaaij, A. Tarlano, M. Luther, P. Nurmi, B. Mrohs, A. Battestini, R. Vaidya, “Context Watcher — Sharing context information in everyday life”, Telematica Instituut, 2006.
- [8] C. Biancalana, F. Gasparetti, A. Micarelli, G. Sansonetti, “An Approach to Social Recommendation for Context-Aware Mobile Services”, Roma Tre University, 2011.
- [9] <http://ru.wikipedia.org/wiki/Foursquare>
- [10] <https://foursquare.com/about/>
- [11] <https://developer.foursquare.com/overview/>
- [12] Namiot, D., Sneps-Sneppe, M. “A new approach to advertising in social networks-business-centric check-ins.” In Intelligence in Next Generation Networks (ICIN), 2011 15th International Conference on (pp. 92-96). IEEE. DOI: 10.1109/ICIN.2011.6081110
- [13] Namiot, D., Sneps-Sneppe, M. “Using Network Proximity for Context-aware Browsing”. International Journal On Advances in Telecommunications, 2012, vol. 5 (3 and 4), pp. 163-172.