



где c_1 и c_2 – известные весовые коэффициенты, установленные экспертами и определяющие цену ошибок от неверного принятия решения;
 p_1 и p_2 – вероятность ошибок первого рода (ошибочное прогнозирование проблемной ситуации) и вероятность ошибок второго рода (ошибочное прогнозирование отсутствия проблемной ситуации) соответственно:

$$p_1 = \sum_{X_1} \frac{P(A^*) \times P(x | A^*)}{P(A) \times P(x | A) + P(A^*) \times P(x | A^*)},$$

где $X_1 = \{x / P(A/x) \geq P_{кр}\}$ - область принятия решения о возникновении проблемной ситуации;

$$p_2 = \sum_{X_2} \frac{P(A) \times P(x | A)}{P(A) \times P(x | A) + P(A^*) \times P(x | A^*)},$$

где $X_2 = \{x / P(A/x) \leq P_{кр}\}$ - область не принятия решения о возникновении проблемной ситуации.

Процесс создания правил прогноза возникновения проблемных ситуаций состоит из следующих этапов:

1. Кодирование наблюдаемых значений признаков.
2. Автошкалирование кодированных численных значений признаков.
3. Нахождение коэффициентов корреляции между признаками и целевым признаком.
4. Анализ коэффициентов корреляции признаков с целевым признаком и выявление наиболее информативных из них.
5. Разбиение исходной выборки на контролирующую и обучающую выборки [2].
6. Автошкалирование кодированных численных значений признаков обучающей выборки.
7. Вычисление коэффициентов ковариации между признаками.
8. Нахождение собственных чисел и собственных векторов ковариационной матрицы.
9. Определение количества главных компонент в соответствии со значениями собственных чисел и заданным уровнем информативности.
10. Определение выражений для нахождения значений главных компонент.
11. Разбиение вариации значений главных компонент на интервалы и оценка условных вероятности попадания значений главных компонент при наличии и отсутствии проблемной ситуации.
12. Нахождение оценки апостериорных вероятностей проблемной ситуации.
13. Нахождение порога оценки апостериорной вероятностей проблемной ситуации, задающего решающее правило.



Данный метод может быть использован для формирования правил прогноза при создании баз знаний интеллектуальных информационных систем прогнозной диагностики в различных предметных областях.

Для экспериментального подтверждения метода использовалась выборка результатов генетических исследований 49 пациентов, среди которых у 30 пациентов не наблюдалась заболелание щитовидной железы, а у 19 пациентов оно было установлено. Проводились генетические исследования, в результате которых для пациентов были определены аллели девяти локусов генов [3]. В результате применения МГК было оставлено четыре главные компоненты. Априорные оценки вероятности наличия и отсутствия заболелания были взяты соответственно 0,21 и 0,79.

Оценка эффективности метода определялась на основе подсчета количества ошибок первого и второго рода. Количество ошибок в первого и второго рода в эксперименте для контролирующей выборки составили около 10% от всего объема выборки, что показывает об эффективности примененного метода.

Литература

1. Горбунов, В.М. Теория принятия решений: учеб. пособие / В.М. Горбунов. – Томск.: Изд-во Национального Исследовательского Томского политехнического университета, 2010. -67 с.
2. Аляутдинова, Ф.Р. Формирование обучающих выборок для машинного обучения в задачах распознавания образов / Ф.Р. Аляутдинова, В.А. Суздальцев // Юность и знание гарантия успеха. – Курск: Изд-во Юго-Западного гос. ун-та, 2015. – С. 246-249.
3. Аляутдинова, Ф.Р. Применение Байесовской вероятности в прогнозной диагностике / Ф.Р. Аляутдинова, В.А. Суздальцев, О.А. Кравцова // Инновации, качество и сервис в технике и технологиях. – Курск: Изд-во Юго-Западного гос. ун-та, 2015. – С. 30-34.

А.А. Бебнев, В.А. Моисеев, С.В. Малахов

СИСТЕМА синхронизации сервисов планирования событий

(Поволжский государственный университет телекоммуникаций
и информатики)

Современное общество невозможно представить без информационных технологий. Большинство людей пользуется мобильными устройствами для повседневных задач, планирования событий и других назначений.

Устройство может быть не одно и на разных платформах, в связи с этим появляется нужда обновлять информацию на каждом гаджете, тем самым тратится много времени. Большинство облачных сервисов — веб-приложения, требующие от пользователя только наличия браузера, в котором они работают, и интернет-подключения. Это позволяет использовать данные в любой точке



планеты и не быть привязанным к одному устройству. Однако некоторые из сервисов нуждаются в установке дополнительных программ (проигрывателя Flash-графики или клиента для мгновенных сообщений), кроме того, для комфортной работы необходимо высокоскоростное подключение (от 512 КБ/с для Google Video, от 256 КБ/с для Google Earth).[1] К тому же, услуги облачных сервисов могут предоставлять разные компании.

В данный момент существует несколько облачных сервисов, предоставляющие примерно одинаковые услуги. Эти услуги могут использовать в своих проектах разработчики приложений. У каждого облачного сервиса есть собственный API (Application Programming Interface). Разработчику, для реализации одной и той же функциональности в приложении, придется использовать API каждого из сервисов. Это увеличивает объем написанного кода, усложняет его поддержку.

Разработка унифицированного доступа поможет разработчикам основной системы ускорить написание проекта, уменьшить объем кода, упростить поддержку. Это позволит сконцентрироваться на более важных аспектах проекта, таких как удобный графический интерфейс пользователя.

Унифицированный доступ к API облачных сервисов будет реализован в виде библиотеки. Все доступные возможности API облачных сервисов будут доступны по единым интерфейсам. Все одинаковые возможности будут объединены одним интерфейсом, с возможностью выбора используемого облачного сервиса. Такие функции как синхронизация заметок, событий в календаре и т.п. можно будет использовать, вызвав лишь один метод для каждой функции облачного сервиса соответственно. Разработчику не нужно будет писать функционал для каждого отдельного сервиса, который он планирует использовать. Все будет реализовано в библиотеке.

Основная цель разработки унифицированного доступа к API облачных сервисов – это упростить жизнь разработчикам. Такой функционал поможет более тщательно подойти к разработке более важных аспектов проекта.

Система будет представлять собой клиент-серверное программное обеспечение, которое при изменении информации на одном устройстве, автоматически синхронизировалось с другими устройствами, даже независимо от сервиса и платформы. Программное обеспечение предоставит удобный сетевой доступ в режиме «по требованию» к коллективно используемого набора настраиваемых вычислительных ресурсов, такого как планировщика событий (в дальнейшем сетей, серверов, хранилищ данных, приложений и/или сервисов), которые пользователь может оперативно задействовать под свои задачи и высвободить при сведении к минимуму числа взаимодействий с поставщиком услуги или собственных управленческих усилий.[2]

Система будет применима в координации рабочего процесса таких служб как техническая поддержка, проектирование и разработка информационных систем, управление персоналом. Управление внутренним информационным ресурсом компании для коллективной работы над задачами, проектами и документами, для эффективных внутренних коммуникаций. Синхронизация



сервиса планирования событий между различными облачными сервисами и корпоративной системой (рис. 1).

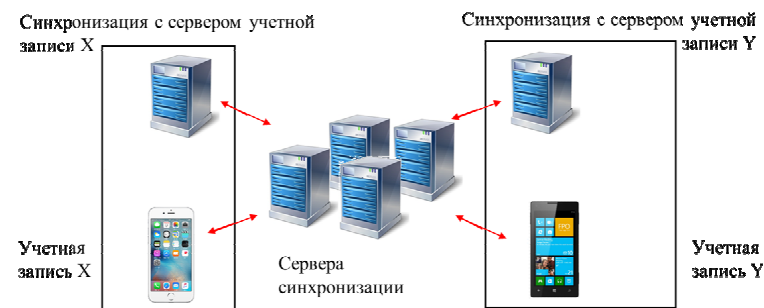


Рисунок 1 – Клиент серверная архитектура сервиса планирования событий
В повседневной жизни система позволит синхронизировать сервисы Google, Microsoft и Apple на различных платформах.

Существуют подобные системы синхронизации, которые позволяют синхронизировать данные между одним поставщиком услуг (синхронизация одного сервиса Google или других). Разрабатываемая система позволит достичь множественной синхронизации данных между различными платформами и сервисов Google, Microsoft, Apple и корпоративными системами (планирования событий или других). Серверную часть системы клиент может устанавливать самостоятельно на свое оборудование либо использовать сервер, разрабатываемый в рамках проекта.

Предложенные методы и средства защиты передачи и хранения персональной информации будут отличаться от традиционных. Эти методы и средства защиты информации можно будет использовать, не только в разрабатываемой системе, но и в корпоративных системах хранения, данных. Предложенный подход позволит увеличить безопасность хранения и обработки персональных данных.

Данное решение проблемы, позволит автоматизировать процесс синхронизации, тем самым значительно экономить время и средства, повышение доступности вычислительных ресурсов.

Литература

1. Список служб и инструментов Google [Электронный ресурс] // Википедия Свободная энциклопедия. -2015. –Режим доступа: https://ru.wikipedia.org/wiki/Список_служб_и_инструментов_Google. свободный. –Загл. с экрана.
2. Что такое облачные сервисы для небольших компаний? [Электронный ресурс] // Торговля, склад и CRM онлайн. -2015. –Режим доступа: http://www.moysklad.ru/chto_takoe_oblachnye_servisy. свободный. –Загл. с экрана.