Санкт-Петербургский государственный университет

Кафедра системного программирования

Минаев Александр Сергеевич

Разработка системы автоматизированного анализа данных мониторинга корпоративных информационных систем

Бакалаврская работа

Научный руководитель:
ст. преп. Я. А. Кириленко

Рецензент:
инженер-консультант ООО "САП Лабс" Н. А. Ражев

Санкт-Петербург
2019

SAINT PETERSBURG STATE UNIVERSITY

Software engineering

Alexander Minaev

Automated data analysis system development for corporate information systems monitoring

Graduation Thesis

Scientific supervisor:
senior lecturer I. A. Kirilenko

Reviewer:
engineer at SAP Labs N. A. Razhev

Saint Petersburg
2019

Оглавление

Введение

В наши дни онлайн-мониторинг различных систем используется повсеместно. Его целесообразность выражается в необходимости своевременно определять, находится ли система в аномальном состоянии, а также предотвращать возможные проблемы и быстро реагировать на них.

При внедрении системы онлайн-мониторинга одним из важнейших этапов является правильная настройка пороговых значений для метрик. В большинстве случаев до сих пор эта задача выполняется вручную. Инженеры при этом пытаются соблюсти баланс между частотой пропуска аномальных состояний и частотой ложноположительных срабатываний. Однако данные, получаемые системой мониторинга, могут зависеть от различных параметров, таких как время суток, день недели, выходной ли день. Также они могут быть цикличны и иметь тренды. Все это приводит к тому, что выставленные вручную пороговые значения показывают плохие результаты ввиду игнорирования вышеперечисленных сложностей. К сожалению, также встречаются аномалии, когда рассматриваемые метрики в каждый отдельный момент времени находятся в допустимых рамках, но в совокупности такое состояние является аномальным[12]. В этом случае пороговые значения оказываются вовсе бесполезными.

Одним из решений вышеперечисленных проблем является использование алгоритмов машинного обучения для автоматического анализа данных, которые могут на основе предыдущих значений метрик своевременно обновлять пороговые значения и классифицировать состояние системы в целом. Такие алгоритмы способны учитывать большинство особенностей изучаемых метрик, такие как цикличность и наличие тренда.[1, 12] Однако на них накладываются существенные ограничения. Например, алгоритм должен уметь работать с минимальным окном исторических данных, поскольку хранение длительных промежутков попросту невозможно. Другим важным фактором является затрачиваемое на переобучение время. Чем оно меньше, тем менее нагру-

жена будет система, и тем чаще мы сможем переобучать модель, что несомненно улучшит ее качество.

Одна из компаний-клиентов SAP указала на проблему наличия слишком частых ложноположительных срабатываний в существующем сервисе онлайн-мониторинга. Вследствие этого возникла идея по созданию расширения для онлайн-мониторинга, в котором будут использоваться различные алгоритмы для улучшения качества поиска аномалий.

1. Постановка задачи

Целью данной работы является создание расширения для онлайн-мониторинга для корпоративных информационных систем SAP, в котором будут использованы различные алгоритмы машинного обучения в целях уменьшения частоты ложноположительных срабатываний и пропуска аномалий. Для ее выполнения были поставлены следующие задачи.

- Провести анализ требований к системе.
- Сделать обзор существующих решений и алгоритмов.
- Реализовать алгоритмы, улучшающие качество поиска аномалий.
- Разработать общую архитектуру решения.
- Создать прототип.
- Провести апробацию.

2. Анализ требований

В данной главе будут изложены требования к разрабатываемой системе, улучшающей качество поиска аномалий в существующем сервисе онлайн-мониторинга. Требования были разделены на функциональные и нефункциональные.

2.1. Функциональные требования

2.1.1. Требования к системе оповещения

Разрабатываемое решение должно реализовывать функциональность системы оповещения. При этом необходимо снизить частоту ложноположительных срабатываний, даже если это будет сделано в ущерб времени реагирования.

2.1.2. Требования к визуализации

Требуется возможность отображения результатов работы алгоритма в существующей системе онлайн-мониторинга в SAP Solution Manager. В данной системе исторические данные отображаются с помощью графиков. Есть возможность добавлять новые метрики и определять для них пороговые значения. Также допустимо изменение пороговых значений для ранее определенных метрик.

2.1.3. Требования к хранению данных

Для последующего анализа причин возникновения аномальных ситуаций аналитиками разрабатываемое решение должно сохранять информацию о найденных аномалиях. Необходима возможность предоставлять данные с помощью выгрузки информации о найденных аномалиях за последний месяц в отдельный csv файл.

2.1.4. Требования к входным данным

Разрабатываемое решение должно использовать в качестве входных данных исключительно исторические значения метрик, собранные системой мониторинга в SAP Solution Manager [6]. Не допускается возможность использования алгоритмов машинного обучения с учителем, поскольку разметка данных является слишком трудозатратным процессом.

2.2. Нефункциональные требования

2.2.1. Требования к используемым продуктам

Решение должно быть построено на базе SAP Cloud Platform [9] или SAP XSA [13].

2.2.2. Требования к производительности

Разрабатываемое решение должно работать в онлайн режиме, значения метрик собираются с 5-ти минутной гранулярностью и подаются на вход с соответствующей частотой алгоритму. Таким образом, разрабатываемое решение должно успевать обрабатывать данные со всех систем и инстанций клиента в этот промежуток времени.

3. Обзор предметной области

В этой главе будут рассмотрены решения в области онлайн-мониторинга, в которых используются различные алгоритмы для улучшения качества поиска аномалий. Кроме того, будут рассмотрены различные алгоритмы для поиска аномалий во временных рядах.

3.1. Существующие решения

3.1.1. Zabbix

Zabbix [14] - это открытое программное обеспечение, а именно, свободная система мониторинга, которая использует различные алгоритмы для контроля за частотой ложноположительных срабатываний и поиска аномалий в данных. Несмотря на все преимущества данного продукта, было принято решение отказаться от его использования в рамках данной работы, ввиду сложности интеграции с SAP Solution Manager, а также развертывания на базе SAP Cloud Platform и SAP XSA. Кроме того, эта система мониторинга крайне сложна. Компанией Zabbix даже проводится сертификация специалистов по своему решению.

3.1.2. Microsoft Azure Monitor

В сервисе Azure Monitor [7] существует возможность настроить оповещения с использованием динамических пороговых значений. Используются алгоритмы машинного обучения для анализа метрик на основе их значений за последние 7 дней. Поддерживается только часовая, дневная и недельная сезонность. Это решение не подходит для использования в рамках данной работы, поскольку отсутствует не только возможность интеграции с системой SAP Solution Manager, но и развертывания на базе SAP Cloud Platform.

3.1.3. Oracle Enterprise Manager Cloud Control

Решение от компании Oracle [2] позволяет использовать динамические пороговые значения для части метрик. Однако существует несколько проблем. Во-первых, используются исключительно статистические вычисления, а именно, есть возможность оповещать при попадании значения метрики в определенный перцентиль или превышении максимального значения на определенный процент в рассматриваемом окне. Во-вторых, администратор сам должен выбирать пороговые значения, например, 0.95 перцентиль. Это решение также нельзя использовать в рамках данной работы по причине невозможности интеграции с SAP Solution Manager.

3.1.4. SAP Focused Run

Решение от компании SAP [11], использующее машинное обучение в системе мониторинга. Это отдельная система, которая реализует собственный мониторинг SAP систем. Также в нем есть существенный недостаток, а именно, используемые алгоритмы машинного обучения для поиска аномалий основаны на обучении с учителем. Это существенно усложняет внедрение такой системы, поскольку необходима ручная разметка данных клиента. Данное решение конкурирует с классической системой мониторинга в SAP Solution Manager.

Таким образом, рассмотренные решения не подходят для использования в рамках данной работы. Собранные требования к разрабатываемой системе предполагают расширение существующей системы мониторинга в SAP Solution Manager, а также построение решения на базе SAP Cloud Platform или SAP XSA. По этим причинам становится нецелесообразно использовать готовые полноценные продукты в сфере системного мониторинга реального времени.

3.2. Обзор литературы

Для увеличения качества поиска аномалий во временных рядах используются различные подходы. Для этого можно использовать как классические алгоритмы поиска аномалий, так и алгоритмы по оптимизации пороговых значений.

3.2.1. Оптимизация пороговых значений

Для уменьшения количества ложноположительных срабатываний можно попытаться оптимизировать пороговые значения метрик. Этот подход зачастую предполагает расчет динамических пороговых значений, которые своевременно обновляются в соответствии с текущими абсолютными значениями метрик[1]. У этого подхода есть несколько преимуществ. Во-первых, можно полностью отказаться от ручной настройки пороговых значений, предоставив всю работу алгоритму. Во-вторых, он позволяет настраивать необходимую частоту срабатываний, исходя из предполагаемой частоты аномальных ситуаций и важности конкретных метрик. Однако у такого подхода есть недостатки. Он применим лишь для метрик, в которых не встречаются всплески их абсолютных значений, поскольку для таких метрик любое пороговое значение будет не информативно: либо порог низкий, в этом случае частота ложноположительных срабатываний будет высока, либо порог будет слишком высоким для выявления аномалий. Также такой подход не способен находить коллективные аномалии.

3.2.2. Метрические алгоритмы

Типичными представителями метрических алгоритмов являются k-NN и k-means. Этот подход основан на предположении, что аномальные значения находятся дальше от обычных, а сами обычные расположены близко друг к другу [15]. Преимуществом данного подхода является наглядная визуализация и интуитивное понимание построенной модели. Однако такие алгоритмы обладают рядом недостатков. Без предобработки они способны выделять только точечные аномалии. Также на

качество метрических алгоритмов сильное влияет выбранная функция расстояния, которую бывает нелегко подобрать [4]. К тому же, если выбрана нетривиальная функция расстояния, сильно возрастает вычислительная сложность данных алгоритмов.

3.2.3. Статистическое моделирование

Данный подход основан на построении распределения исследуемой метрики и последующем предположении, что нормальные значения расположены в регионах с высокой вероятностью, в то время как аномальные значения расположены в регионах с низкой вероятностью. Данный подход обладает следующими преимуществами. Во-первых, если предположения верны, то алгоритм предоставляет статистически обоснованное решение. Во-вторых, значения системных метрик часто распределены согласно нормальному закону. Это сильно упрощает построение математической модели и последующих вычислений. Однако у него также есть недостатки. Если распределение, построенное по какой-то выборке, плохо соответствует общей картине, то все полученные результаты нельзя считать достоверными [4].

3.2.4. Скользящие окна

Алгоритмы на основе скользящих окон работают в комбинации с другими методами поиска аномалий. Основная идея данного подхода заключается в том, что аномальность определяется не для конкретного абсолютного значения, а для целого окна [15]. Это позволяет игнорировать единичные выбросы в данных, которые и порождают частые ложноположительные срабатывания. В то же время, этот подход позволяет находить коллективные аномалии, например, если отдельные значения исследуемой метрики находятся в рамках допустимых границ, однако общая картина предполагает наличие аномалии.

4. Реализация алгоритма

4.1. Выбор общего подхода

Для уменьшения частоты ложноположительных срабатываний необходимо искать не точечные, а коллективные аномалии. Системы SAP, метрики которых собираются, являются серверами приложений, на которых выполняется множество различных задач. Ввиду нагруженности, в значениях метрик постоянно наблюдаются всплески, однако они не являются аномальными ситуациями. Таким образом обычные пороговые значения, которые используются в существующем сервисе мониторинга, не являются информативными и возникает проблема частых ложноположительных срабатываний. Чтобы бороться с этим явлением, необходимо отказаться от поиска точечных аномалий. Разрабатываемое решение должно осуществлять поиск коллективных аномалий. Было принято решение реализовывать алгоритм на основе статистического моделирования и скользящих окон. За основу был взят алгоритм из статьи [12]. В этой работе строится распределение ошибок прогнозирования метрик, используются скользящие окна. На их основе вводится новая метрика, называющаяся вероятностью аномалии и принимающая значения от 0 до 1. Затем уже для этой метрики выбирается статический порог, исходя из требуемого баланса между частотой ложноположительных срабатываний и частотой пропуска аномалий. В этой работе также предлагается использовать пороговое значение для аномальности равное $1 - 10^{-4}$. Преимуществами данной работы являются интуитивно понятный способ поиска аномалий и использование прогнозирования значений метрик. К тому же введение новых метрик и установка их пороговых значений легко интегрируется в существующую систему мониторинга в SAP Solution Manager

4.2. Выбор алгоритма прогнозирования

SAP системы разделены на инстанции, для каждой из которых определено большое число метрик. На разных системах или даже инстан-

циях тренды и цикличность могут не совпадать. Поэтому становится нецелесообразно использовать классические подходы с временными рядами, например, ARIMA и его вариации. Для качественной работы таких алгоритмов требуется тонкая настройка параметров модели и предобработка данных. Поэтому необходимо использовать решения, которые могут работать при минимальной настройке. Было опробовано несколько библиотек.

В работе [12] в качестве предсказателя использовалась авторская библиотека машинного обучения NuPIC [8]. При попытке использовать ее на данных заказчика возникло множество проблем. Библиотека оказалась неспособной качественно прогнозировать значения метрик.

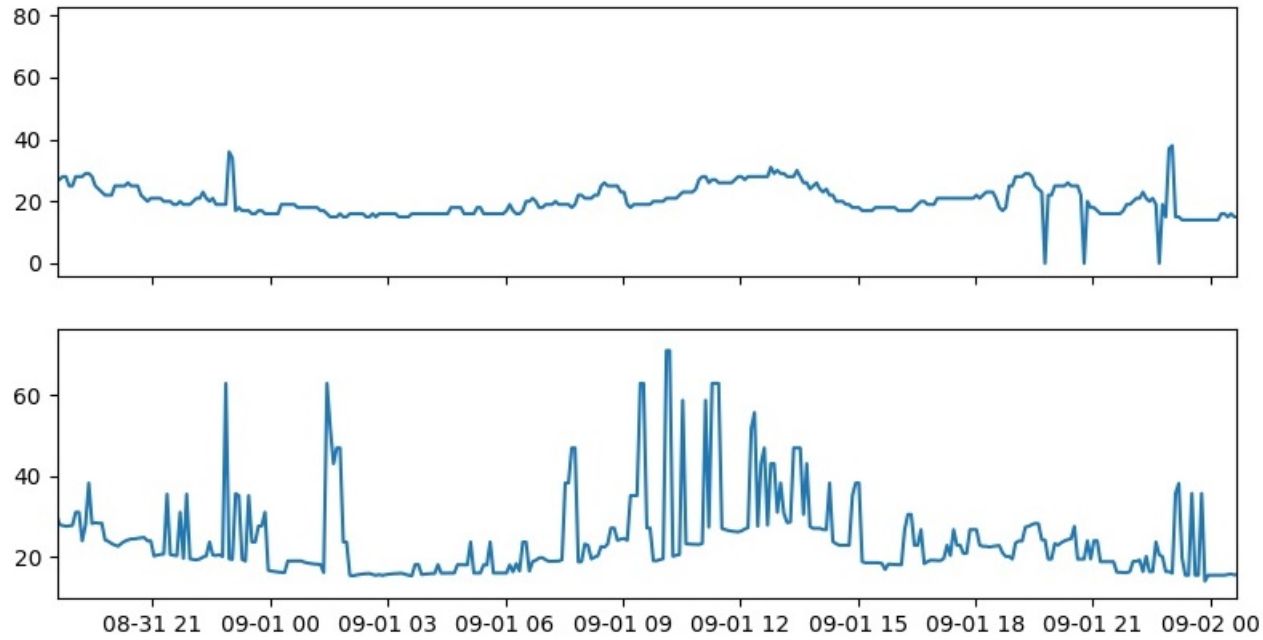

Рис. 1: MEMORY_EM_USED_nupic

В верхней части рис. 1 можно наблюдать значения метрики ABAP_INST_MEMORY_EM_USED, которая отражает процент потребляемой дополнительной памяти в инстанции. В нижней части - прогнозируемые значения с помощью библиотеки NuPIC.

Использование интерполирования для получения предполагаемого

значения метрики [5] также оказалось нецелесообразно из-за низкого качества предсказательной способности. В этой работе предлагалось использовать квадратичное интерполирование для получения значения метрик в определенное время суток с помощью сетки исторических значений метрик по часам. Эта сетка обновляется с помощью скользящего взвешенного среднего. При попытке использовать этот подход на данных клиента время от времени возникали случаи, когда квадратичная интерполяция выдавала отрицательные значения метрик.

Было найдено подходящее решение. Библиотека для работы с временными рядами Prophet [10] от компании Facebook обладает рядом преимуществ. Для прогнозирования не требуется предобработка данных, она обладает достаточной предсказательной способностью. Также для правильной работы не критичны незначительные пропуски в данных, поскольку эта библиотека самостоятельно обрабатывает такие случаи.

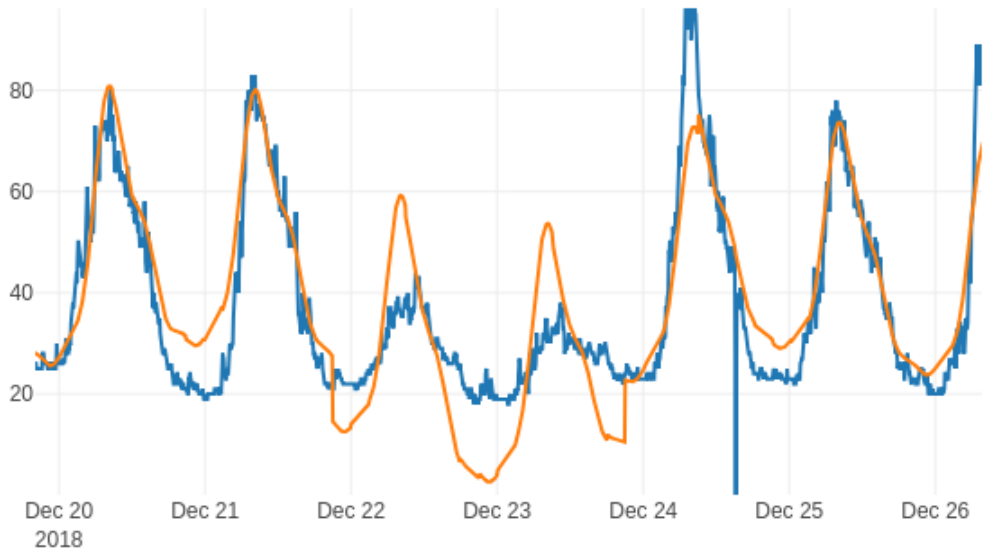

Рис. 2: MEMORY_EM_USED_prophet_1

Однако, несмотря на все преимущества Prophet, всё же возникли сложности. Данные с системы мониторинга SAP имеют сильно выраженную периодичность длиною в 7 дней. При этом в выходные дни значения большинства метрик сильно ниже, чем в рабочие. Это отражено на рис. 2. Синяя кривая показывает изменение реального значения метрики во времени, оранжевая - прогнозируемое значение.

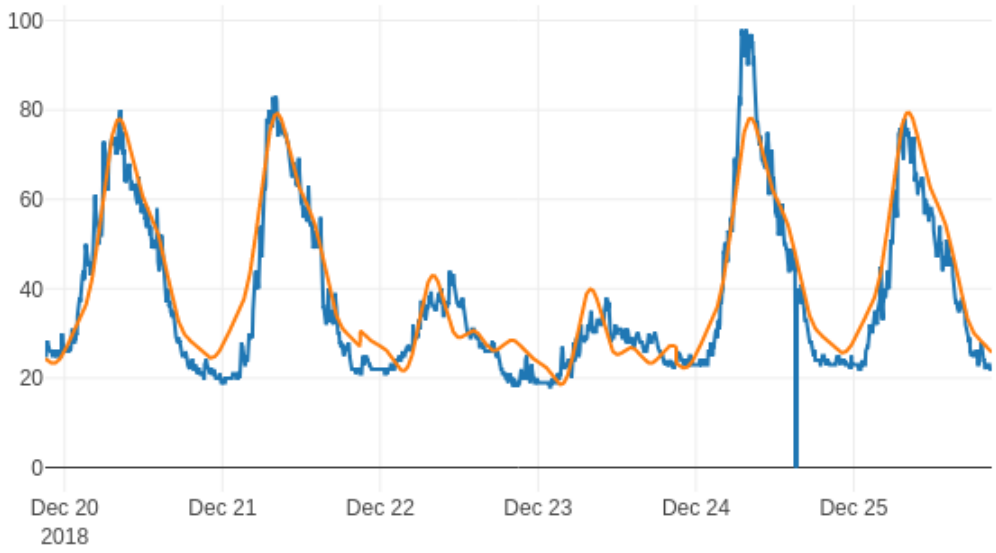

Рис. 3: MEMORY_EM_USED_prophet_2

Эту проблему удалось решить с помощью введения множества дополнительных регрессоров, что позволило улучшить дневную сезонность модели в выходные дни. Это отражено на рис. 3

5. Архитектура системы и особенности реализации

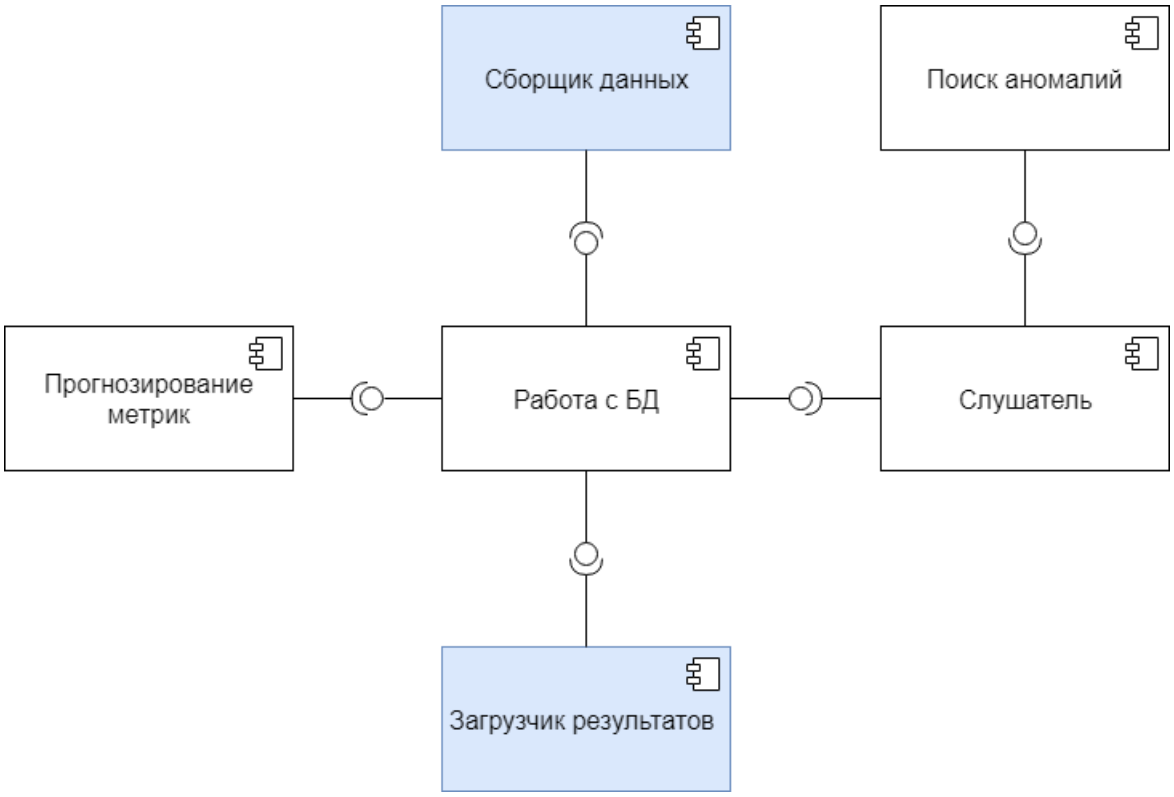

Рис. 4: Общая архитектура системы

На рисунке 4 представлена диаграмма компонент общей архитектуры решения, построенной на основе собранных требований к системе. Синим цветом выделены компоненты, реализованные одним из разработчиков SAP Labs, который работает с клиентом-заказчиком, поскольку для их реализации необходим доступ к системам клиента. Эти компоненты отвечают за сбор данных с системы мониторинга и их загрузку в базу данных, а также последующую загрузку результатов работы алгоритмов в систему SAP Solution Manager. Значения метрик в системе мониторинга собираются с периодичностью в 5 минут.

В качестве основного языка программирования был выбран Python, так как он предполагает легкое прототипирование, удобен для использования в задачах машинного обучения и научных вычислений. Используемая библиотека для прогнозирования временных рядов Facebook

Prophet также реализована на языке Python. В качестве фреймворка для создания веб-приложений в SAP Cloud Platform был использован Flask [3]. Он удобен ввиду своей простоты, модульности, а также идеально подходит для создания небольших приложений-оберток над научными вычислениями. В решении используется база данных PostgreSQL, так как она предоставляется в качестве сервиса в облачном окружении SAP Cloud Platform.

5.1. Компонент прогнозирования метрик

С периодичностью в 23 часа данные мониторинга всех систем и инстанций клиента выгружаются из базы данных PostgreSQL компонентом прогнозирования аномалий. Для формирования прогноза требуются данные за последние три недели. Далее формируется прогноз на следующие 24 часа. Промежуток в час выбран из необходимости, так как для обучения на данных со всех систем и инстанций клиента и последующего прогнозирования требуется продолжительное время.

5.2. Компонент поиска аномалий

Данный компонент отвечает за поиск аномалий в данных. Он предоставляет интерфейс для получения значений метрик, а также значений прогнозов. Алогритм, реализованный в данном компоненте, в отличие от прогнозирования метрик не потребляет значительных ресурсов и его время работы пренебрежительно мало. Однако он также выделен в отдельный компонент для упрощения возможных улучшений или замены данного компонента в будущем. Кроме того для его работы требуется наличие исторических значений, а именно скользящих окон, описанных в работе [12]. Данный компонент сохраняет эти окна, своевременно обновляя их по мере получения данных.

5.3. Компонент слушатель

Данный компонент необходим из-за нечетких интервалов поступления данных из системы мониторинга в SAP Solution Manager. С периодичностью в 10 секунд он опрашивает базу данных PostgreSQL на предмет наличия новых данных, ранее не обработанных. В случае успеха, он с помощью предоставляемого интерфейса передает их в компонент поиска аномалий. В случае, если были найдены аномалии, он сохраняет информацию о них в отдельную таблицу в базе данных.

5.4. Компонент работы с БД

Данный компонент предоставляет интерфейсы для работы с базой данных PostgreSQL для остальных компонентов. В качестве драйвера базы данных используется библиотека py-postgresql. Кроме того, этот компонент реализует интерфейс для выгрузки информации об аномалиях в формате csv файла для возможности последующего анализа аналитиками клиента.

6. Апробация

В данной работе для тестирования использовались данные с клиентской системы мониторинга в SAP Solution Manager, которые собирались с начала августа 2018 г. до конца апреля 2019 г. Вместе со значениями метрик также собирались данные о соответствующих текущих пороговых значениях. Кроме того, данные, собранные по метрике ABAP_INST_MEMORY_EM_USED, были размечены вручную ведущим инженером поддержки команды SAP Solution Manager, который работал с данным клиентом. Было найдено всего три существенных аномалии за текущий промежуток времени. Это связано с общей высокой стабильностью решений компании SAP.

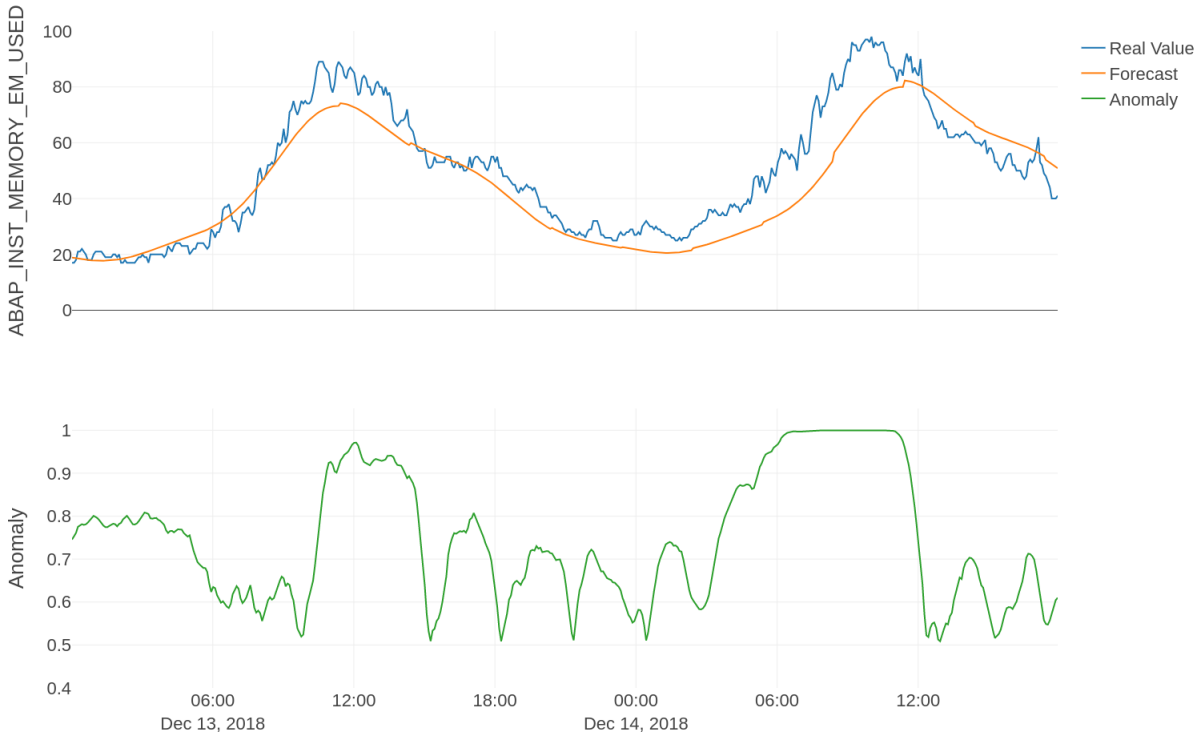

Рис. 5: Аномалия в декабре

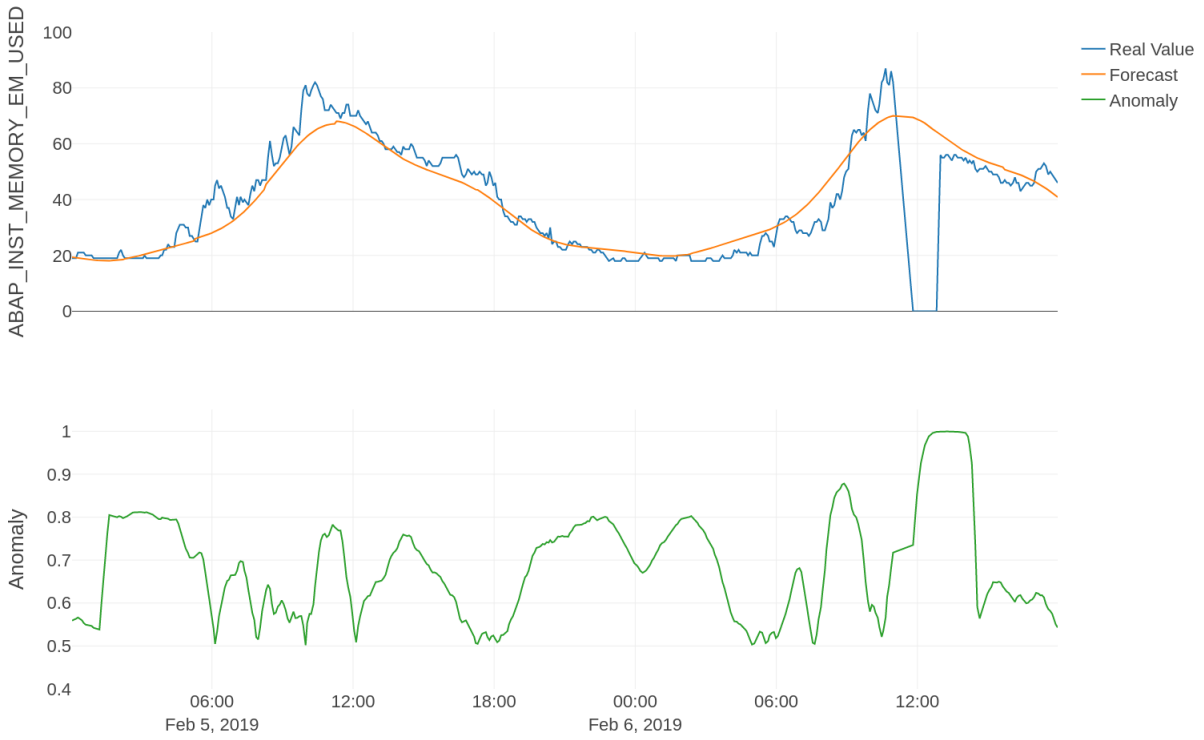

Рис. 6: Аномалия в феврале

На рис. 5 и рис. 6 можно наблюдать результаты работы реализованного алгоритма при настоящих аномальных ситуациях при выборе порогового значения аномальности равного $1 - 10^{-4}$. В первом случае загрузка памяти началась намного раньше прогнозируемого времени, кроме того абсолютные значения метрики практически достигли своего максимума и превысили критический порог, установленный в системе мониторинга. На втором примере можно наблюдать отсутствие собранных данных в системе мониторинга SAP Solution Manager. В обоих случаях алгоритм успешно классифицировал рассмотренные промежутки как аномальные. В первом примере это удалось сделать за несколько часов до соответствующего срабатывания в существующей системе мониторинга при превышении порогового значения.

Всего на промежутке с сентября по апрель рассматриваемая метрика превысило первое пороговое значение, которое равно 90, 38 раз. В то время как решение реализованное в рамках данной работы превысило свое пороговое значение только 9 раз.

Прототип системы, реализованный в SAP Cloud Platform, был успешно протестирован с использованием тестовой SAP системы. На момент написания данной статьи планируется внедрение разработанного решения компании-клиенту, которая предоставляла данные.

Заключение

В ходе выполнения данной работы были выполнены следующие задачи.

- Проведен анализ требований к системе со стороны заказчика.
- Сделан обзор существующих решения на рынке от компаний Microsoft, Oracle и SAP. Выявлены их недостатки и преимущества. Рассмотрены различные методы поиска аномалий во временных рядах.
- Реализован алгоритм из работы [12] и опробован на данных заказчика.
- Разработана общая архитектура решения на основе SAP Cloud Platform и SAP XSA.
- Создан прототип с использованием Flask, Facebook Prophet и PostgreSQL.
- Проведена апробация системы.

Список литературы

[1] Breitgand David, Goldstein Maayan, Shehory E.H. Efficient Control of False Negative and False Positive Errors with Separate Adaptive Thresholds // IEEE Transactions on Network and Service Management. — 2011. — no. 8(2). — P. 128–140.

[2] Control Oracle Enterprise Manager Cloud. — URL: https://www.oracle.com/ru/enterprise-manager/.

[3] Flask. — URL: http://flask.pocoo.org/.

[4] Hayes Michael. Contextual Anomaly Detection Framework for Big Sensor Data // Electronic Thesis and Dissertation Repository. — 2014.

[5] Lambert Diane, Liu Chuanhai. Adaptive Thresholds: Monitoring Streams of Network Counts Online // Journal of the American Statistical Association. — 2006. — no. 101. — P. 78–88.

[6] Manager SAP Solution. — URL: https://www.sap.com/products/solution-manager.html.

[7] Monitor Microsoft Azure. — URL: https://azure.microsoft.com/ru-ru/services/monitor/.

[8] NuPIC. — URL: https://github.com/numenta/nupic.

[9] Platform SAP Cloud. — URL: https://cloudplatform.sap.com/index.html.

[10] Prophet Facebook. — URL: https://cloudplatform.sap.com/index.html.

[11] Run SAP Focused. Focused Run. — URL: https://support.sap.com/en/alm/focused-solutions/focused-run.html.

[12] Unsupervised real-time anomaly detection for streaming data / Subutai Ahmad, Alexander Lavin, Scott Purdy, Zuha Agha // Neurocomputing. — 2017.

[13] XSA SAP. — URL: https://developers.sap.com/cis/tutorials/xsa-explore-basics.html.

[14] Zabbix. — URL: https://www.zabbix.com/.

[15] Викторович Соболев Константин. Автоматический поиск аномалий во временных рядах // Московский физико-технический институт. — 2018.