Анализ существующих методов и алгоритмов анализа программных систем, использующих системы контроля версий

Перепелицына Екатерина ПИмд-11

1 Введение

Система контроля версий(СКВ)—это система, записывающая изменения в файл или набор файлов в течение времени и позволяющая вернуться позже к определённой версии. Анализ проектов, использующих СКВ, позволяет отслеживать изменения проекта в любой момент времени. Например, анализ подключаемых библиотек к проекту можно сделать по последнему коммиту, а можно отследить все подключаемые к проекту библиотеки, используя СКВ, переключаясь между коммитами и получить некоторую динамику. Исследование существующих методов и алгоритмов анализа любых изменений и действий через СКВ помогает лучше понять, как использовать СКВ для получения максимальной информации о проекте, на какие изменения стоит обращать внимание и какие данные было бы интересно отслеживать.

2 Публикации

В статье [1] приведен анализ характера изменений программ и поиск неисправленных фрагментов кода. Целью данной работы является разработка методов анализа характера изменений между версиями компонентов ПО, для которых отсутствует исходный код.

В статье [2] описан прототип автоматизированной системы, основной задачей которой является поиск и подбор команд специалистов на основе данных открытых репозиториев исходного кода и связанных артефактов. В статье подробно рассматриваются состав архитектуры системы, алгоритм выбора основной команды проекта, выявленные в ходе исследования метрики деятельности группы, формулы расчетов значений метрик, а также их применение при решении задачи анализа проектного репозитория.

В статье [3] рассмотрена архитектура системы обработки больших данных, основанной на инструментах Apache Hadoop, Apache Flume и Apache Spark. Продемонстрировано применение разработанной системы для хранения и анализа наборов данных, состоящих из генерируемых событий в репозитории GitHub – крупнейшего в мире веб-сервиса на базе системы контроля версий Git.

В статье [4] рассматриваются вопросы, связанные с использованием методов анализа данных применительно к программным репозиториям. В работе делается попытка представить обзор технологий, которые используются при анализе программ и базируются на статических данных, которые могут быть извлечены непосредственно из программного кода или репозиториев кода. В работе приводится обзор работ, использующих методы глубокого обучения (рекуррентные нейронные сети), методы классификации, основанные на других моделях машинного обучения, а также использование кластеризации в программной инженерии. Практические области применения рассматриваемых методов включают в

себя, например, классификацию и предсказание ошибок, определение характеристики изменения кода во времени, поиск дублирующих фрагментов, автоматическое обнаружение ошибок проектирования, выдачу рекомендаций по рефакторингу кода.

В ВКР [5] рассматривается разработка системы отслеживания ошибок в программных продуктах. Проведен анализ процесса тестирования программных продуктов на наличие ошибок, описывается процесс создания системы отслеживания ошибок в программных продуктах.

В статье [6] рассматривается возможность прогнозирования тенденций репозитория и языков программирования с использованием временных рядов и анализом событий репозиториев GitHub.

В статье [7] приводят результаты анализа тональности текста коммитов, так как эмоции имеют высокое влияние на продуктивность, качество выполнения задач и прочее. В основном сообщения к коммитам являются нейтральными, но бывают исключения. В исследовании были рассмотрены проекты на 14 разных языках программирования, самые негативные комментарии к коммитам наблюдаются в проектах, написанных на языке программирования Java, также самые негативные комментарии были оставлены в понедельники.

В статье [8] рассматривают анализ репозиториев и пользователей GitHub. Рассматриваются зависимости между поведением пользователя и успешности проектов, в которых он участвует. В статье пытаются выявить закономерности, понять, что имеет наибольшее влияние на успешность проекта для возможности дальнейшего прогноза.

В статье [9] рассматривается получение информации с сайта GitHub. Так как Github предоставляет API для доступа к большому количеству информации, многие пытаются получить какую-то полезную информацию с этого ресурса. В статье приводятся основные ошибки, допускаемые при анализе данных, получаемых при неправильных запросах и как лучше строить свои запросы к сервису.

В статье [10] предложен к рассмотрению инструмент, созданный для получений структурированных данных по ряду параметров, заданных в запросе к API Github. Инструмент предлагает интерфейс для создания запросов более удобный, чем прямое обращение к API.

Заключение

В результате анализа имеющихся алгоритмов и результатов анализа предыдущих исследователей были рассмотрены разработанные инструменты,алгоритмы и методы анализа и получения данных с систем контроля версий, какие данные можно получить и как ее можно анализировать.

Список литературы

[1] АРУТЮНЯН М.С., ИВАНОВ Г.С., ВАРДАНЯН В.Г., АСЛАНЯН А.К., АВЕТИСЯН А.И., КУРМАНГАЛЕЕВ Ш.Ф. АНАЛИЗ ХАРАКТЕРА ИЗМЕНЕНИЙ ПРОГРАММ И ПОИСК НЕИСПРАВЛЕННЫХ ФРАГМЕНТОВ КОДА. [Электронный ресурс] - URL:https://elibrary.ru/item.asp?id=37313183

[2] ЯРУШКИНА НАДЕЖДА ГЛЕБОВНА, ЖЕЛЕПОВ АЛЕКСЕЙ СЕРГЕЕВИЧ ПРОТОТИП СИСТЕМЫ ПОИСКА И ВЫБОРА "СФОРМИРОВАННЫХ"КОМАНД ИТ-СПЕЦИАЛИСТОВ НА ОСНОВЕ ДАННЫХ ПРОЕКТНЫХ РЕПОЗИТОРИЕВ. [Электронный ресурс] - URL:https://elibrary.ru/item.asp?id=42665427

[3] ВОИНОВ Н.В., К. РОДРИГЕС ГАРСОН, НИКИФОРОВ И.В., ДРОБИНЦЕВ П.Д. СИСТЕМА ОБРАБОТКИ БОЛЬШИХ ДАННЫХ ДЛЯ АНАЛИЗА СОБЫТИЙ РЕПОЗИТОРИЯ GITHUB. [Электронный ресурс] - URL:https://elibrary.ru/item.asp?id=38309155

[4] Д.Е. Намиот, В.Ю. Романов Анализ данных для программных репозиториев [Электронный ресурс] - URL:http://injoit.org/index.php/j1/article/view/560

[5] Яцутко, Сергей Анатольевич Разработка системы отслеживания ошибок в программных продуктах. [Электронный ресурс] - URL:http://elib.sfukras.ru/handle/2311/141513

[6] T V Varuna; Anuraj Mohan Trend Prediction of GitHub using Time Series Analysis [Электронный ресурс] - URL:https://ieeexplore.ieee.org/abstract/document/8944878

[7] Emitza Guzman, David Azócar, Yang Li Sentiment Analysis of Commit Comments in GitHub: AnEmpirical Study. [Электронный ресурс] - URL:https://www.researchgate.net/publication/266657943$_Sentiment_analysis_of_commit_comments_in_G$

[8] Fragkiskos Chatziasimidis; Ioannis Stamelos Data collection and analysis of GitHub repositories and users. [Электронный ресурс] - URL:https://ieeexplore.ieee.org/abstract/document/7388026

[9] Georgios Gousios; Diomidis Spinellis Mining Software Engineering Data from GitHub. [Электронный ресурс] - URL:https://ieeexplore.ieee.org/document/7965403

[10] Shreyansh Surana, Smit Detroja, Saurabh Tiwari A Tool to Extract Structured Data from GitHub. [Электронный ресурс] - URL:https://arxiv.org/abs/2012.03453