

УДК 004.413.5

БАЙЕСОВА ОЦЕНКА КАЧЕСТВА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**Смыков Максим Иванович,**

студент,

Северный (Арктический) федеральный университет имени М.В. Ломоносова

Кафедра информационных технологий

maks015@rambler.ru

Аннотация

Статья посвящена проблеме релевантного оценивания качества программного обеспечения. Задача актуальна в условиях эволюции методов проектирования и тестирования программ, их комплексов с учетом потребительских ожиданий и потребностей, улучшения качества и структурированности процесса их разработки. Методами системного анализа в работе были исследованы характеристики и атрибуты программных комплексов. В частности, функциональности, надежности (отказоустойчивости), дружелюбности (удобства применения), эффективности, сопровождения, переносимости и безопасности программных комплексов. Предложена математическая модель для оценки релевантности, а также качества комплексов при неопределенностях по Байесу.

Ключевые слова: Байес, программы, оценивание, комплекс, качество, надежность, модель

BAYES SOFTWARE QUALITY ASSESSMENT**Maksim I. Smikov**

Студент-магистрант,

Federal State Autonomous Educational Institution of Higher Education «Northern (Arctic)

Federal University named after M. V. Lomonosov»

Department of Information Technologies

maks015@rambler.ru**ABSTRACT**

The article deals with the problem of relevant assessment of software. The task is relevant in the context of the evolution of methods for designing and testing programs, their complexes, taking into account consumer expectations and needs, improving the quality and structuring of the process of their development. By the methods of system analysis, the characteristics and attributes of software complexes were investigated in the work. In particular, functionality, reliability (fault tolerance), friendliness (usability), efficiency, maintenance, portability and security of software systems. A mathematical model for assessing the quality of complexes in conditions of uncertainty and subjectivity of estimates based on Bayes formula is proposed.

Keywords: Bayes, software, quality, complex, reliability, evaluation, model

Введение

Качество программного обеспечения (КПО) и его верификации (тестирования) – гарантия устойчивости и эффективности на всех этапах жизненного цикла (ЖЦ) ПО, имиджевости разработчика и лояльности потребителя [1]. КПО позволяет также развивать бизнес-процессы проектирования и разработки, сертификации, унификации и стандартизации [2]. Можно также контролировать программный продукт согласно спецификациям заказчика и потребительским (эксплуатационным) ожиданиям пользователя, т.е. в соответствии с моделью CapabilityMaturityModelIntegration (СММІ) комплексного подхода [3].

СММІ – результат эволюции первоначальной модели, лучших практик по отдельным процессам, количественных техник контроля качества выполнения этих процессов, которые классифицируются на 22 процессные области. Внедрение СММІ – путь к улучшению структуры и качества процессов программных разработок, стабилизации их высокого качества.

Есть различные модели оценивания КПО (например, [4]). Важна измеримость, интегральная метрика, процедура шкалирования и оценки. Если не удастся ввести количественно оцениваемую формализованную метрику, модель (согласно ISO/IEC 9126-1:2001, р.1; ISO/IEC 9126), то используют экспертно-эвристические процедуры, тесты (аттестации), программные инструменты (утилиты) [5]. Но такой подход может усилить (внести шумы) неопределенности в данных, поэтому важна исходная гипотеза распределения данных, ее релевантность.

Характеристики оценки КПО и процессов оценки

Согласно требованиям ISO/IEC (документам 9126-1:2001, часть 1. Модель качества; TR 9126-2:2003, часть 2. Внутреннее качество и часть 3. Внешнее качество; TR 9126-4:2004, часть 4. Метрики качества использования) уровень и направленность (внутренняя/внешняя) оценивания КПО может различаться, имеет различные наборы характеристик КПО. Кроме общесистемного характеристического свойства – эмерджентности, могут использоваться следующие характеристики и атрибуты КПО, близкие к указанным в [1], но расширенные и дополненные:

1. функциональность – способность решать задачи, согласно спецификациям (заказчика, ТЗ) и потребительским ожиданиям, реализуется пригодностью, точностью, полнотой, взаимодействием, стандартизацией и защищенностью;
2. надежность – способность поддерживать требуемый уровень и необходимую степень устойчивости и работоспособности на весь период использования ПО, реализуется отказоустойчивостью, восстанавливаемостью после сбоя;
3. дружелюбность – способность быть комфортным, удобным («юзабилити» – для веб-ресурсов) как при освоении, так и при использовании (администрировании – для системного ПО), реализуется понятностью, простотой активации, практичностью, доступностью, привлекательностью (дизайном экрана и интерфейса);
4. эффективность работы – способность оптимизировать меру типа «уровень предоставляемого качества функционирования – объем активизируемых при

этом ресурсов», реализуется временными, вычислительными, сложностными параметрами производительности;

5. сопровождаемость, поддерживаемость – способность развивать(ся) ПО и КПО, реализуется параметризуемостью, модульностью, идентифицируемостью, тестируемостью (верифицируемостью);
6. переносимость – способность устойчиво работать и при переносе с одной среды в другую, при изменении организационных, аппаратно-программных, архитектурных аспектов окружения и платформы, реализуется, адаптивностью, мультиплатформенностью, взаимозаменяемостью, стандартизованностью, совместимостью;
7. безопасность – защищённость данных, ПО, конфиденциальность операций, авторизуемость и аутентифицируемость, реализуется мерами и средствами защиты.

В работе [1] предложено ранжирование метрик качества: низкое, среднее, хорошее, отличное. Низкое – худший (возможно, гипотетический) случай. Высокий уровень – наилучший (чаще также гипотетический) случай на горизонте наблюдений.

При неопределенности ситуаций, неточности данных и экспертных оценок, такое шкалирование может оказаться излишним, но всегда нужна адаптивная шкала, которая может более тщательно ранжироваться между двумя и более вышеприведенными уровнями, не столь тщательно отслеживая метрики на других уровнях, т.е. сочетая «жесткую» и «мягкую» форму ранжирования (шкалирования).

Байесовский подход и критерий оценивания КПО

Неопределенность и субъективность оценок могут быть сглажены использованием формулы Байеса [6].

Пусть даны k уровней ранжирования – L_1, L_2, \dots, L_k , n атрибутов A_1, A_2, \dots, A_n и матрица метрик M , которые применимы в задаче оценки КПО. Гипотеза H_i : КПО соответствует рассматриваемому текущему уровню L_i ($i = 1, 2, \dots, k$).

Согласно гипотезам строится распределение $P(H_i), i = 1, 2, \dots, k$. Априори, до результатов выбранных метрик. Например, можно взять средневзвешенную инструментальную $P(H_i) = \omega_i$ или даже равновероятную: $P(H_i) = 1/k$.

Чаще принято брать априорными вероятностями на шаге l условные (апостериорные) вероятности по Байесу

$$P(H_i) \equiv P(H_i | A_1, \dots, A_{i-1}, A_{i+1}, \dots, A_n) \equiv P(H_i | A_1, A_2, \dots, A_n),$$

полученные в ходе измерения КПО на шаге $l - 1$. Вероятность $P(A_j | H_j)$ может быть показателем меры соответствия результатов, измеряемых метриками для A_j , гипотезе H_j , т.е. отражением правдоподобности.

Значение

$$P(A_j | H_j) = \frac{\text{count}_i(M_{jl} \in L_i)}{rS_j},$$

где M_{jl} – значения метрик атрибута A_j , $l = 1, 2, \dots, S_j$, $r \geq 1$ – количество измеренных метрик (экспертов, привлеченных к экспертному оцениванию метрик), S_j – количество метрик для оценивания $A_j, j = 1, 2, \dots, n$.

Вероятность $P(A_j | H_j)$ – отношение количества принадлежностей метрических значений текущему ранжированию (рассматриваемому мнению эксперта) L_i и числа всех измерений.

Условная вероятность:

$$P(H_i|A_1, \dots, A_n) = \frac{P(H_i)P(A_1|H_i)P(A_2|H_i) \cdots P(A_n|H_i)}{\sum_{h=1}^k P(H_h)P(A_1|H_h)P(A_2|H_h) \cdots P(A_n|H_h)}$$

отражает гипотезы H_1, H_2, \dots, H_k и может служить интегральной оценкой КПО, правдоподобия гипотез, что интегральное КПО достигает каждый уровень.

Совпадение $P(H_i)$ и $P(H_i|A_1, \dots, A_n)$ распределений на $\{H_1, \dots, H_k\}$ свидетельствует о противоречивости измерений, не влияет на интегральное распределение. Если

$$P(H_i|A_1, \dots, A_n) = 1, P(H_h|A_1, \dots, A_n) = 0 \quad \forall h = 1, 2, \dots, k,$$

то распределение на $\{H_1, \dots, H_k\}$ вырождается: гипотеза H_i – достоверна, а остальные – невозможны.

При средневзвешенных оценках низкое КПО по атрибуту компенсируемо весами, высокими КПО (эффект компенсации [7]).

Общее, интегральное оценивание КПО может быть искаженным, не позволяющим обосновать уровень КПО. Нет согласия в экспертном сообществе по измерению КПО [8]. Оценщику КПО сложно отдавать предпочтение некоторой из гипотез, апостериорная вероятность позволяет по заранее определенным уровням, без такого предпочтения, оценить обобщенно КПО.

Системная инженерия, ISO позволит объединить технологически все управленческие средства для развития потребительских ожиданий и решений [9], используя тестирование, принятие решений, ситуационное моделирование и др.

В разработке ПО необходимо учитывать системные принципы Д.Хитчинса [10] – единства, динамизма, оптимизации, разумной достаточности, синергии, снижения энтропии и др. Инженерия в управлении организациями, проектами, решениями, технологиями, стандартами и другими всегда успешна.

Заключение

Системные программисты, аналитики формируют новый комплексный подход к сложноорганизованным системам, их системной, но клиент-ориентированной организации. Выделяют ключевые направления: ориентированное на системные решения (ИТ-экосистемы) и на управляющие решения (технологии, их иерархическое использование).

Системная инженерия направлена на создание документов, а не оборудования, поддержку процессов разработки (жизненного цикла) всего проекта. Она направлена на ключевые функции – уточнение потребностей и взаимодействий с клиентом, анализ решений, планирование релевантных процессов, управление их мониторингом и реализацией, оценку продукта.

Процедуры проектирования, принятия решений, оценивания обеспечивают коммуникативность, инновационность, креативность программистов.

Список литературы:

1. Бураков Д.П., Кожомбердиева Г.И. Использование формулы Байеса при оценивании качества программного обеспечения по стандарту ISO/IEC 9126 // Программные продукты, системы. 2019. №1(32). с.34-46.
2. ISO/IEC 25010:2011. Systems and Software Engineering Systems and Software Quality Requirements and Evolution (SQuaRE) – System and software quality models. URL: <https://www.iso.org/standard/35733.html> (дата обращения: 02.05.2022).
3. Ахен Д., Клауз А., Тернев Р. CMMI: Комплексный подход к совершенствованию процессов. Практическое введение. –М.: МФК. 2005. –330с.

4. Жарко Е.Ф. Сравнение моделей качества программного обеспечения: аналитический подход // XII Собрание по проблемам управления: сб. трудов ВСПУ. 2014. с.4585-4594. URL: <http://vspu2014/ipu/ru/proceedings/prcdngs/4585.pdf> (дата обращения: 01.05.2022).
5. Intel VTune Amplifier. 2017. URL: <https://software.intel.com/sites/default/files/managed/d7/ba/intel-brief.pdf> (дата обращения: 07.05.2022).
6. Уткин Л.В. Анализ риска и принятие решений при неполной информации. –СПб: Наука. 2007. -404с.
7. Микони С.В. Теория принятия управленческих решений. –СПб: Лань. 2015. -448с.
8. Гласс Р. Факты и заблуждения профессионального программирования. –СПб: Символ-Плюс. 2007. -240с.
9. Глухова Л.В., Казиев В.М., Казиева Б.В. Анализ и синтез сетевых бизнес-структур цифровой экономики. Сб. трудов научно-практической конференции «Право и экономика: прогресс и цифровые технологии» / Под ред. Ю.К.Альтудова, А.Е.Карлика и др. 2019. -с.143-147.
10. Hitchins D. What are the General Principles Applicable to Systems? – INCOSE INSIGHT. 2009. Vol.12. Iss.4. –pp.59–64.

References:

1. Burakov D.P., Kozhombardieva G.I. Use of Bayes formula in evaluation of software quality according to ISO/IEC9126 // Software products and systems. 2019. No.1(32). –pp.34-46.
2. ISO/IEC 25010:2011. Systems and Software Engineering Systems and Software Quality Requirements and Evolution (SQuaRE) – System and software quality models. URL: <https://www.iso.org/standard/35733.html> (accessed date: 02.05.2022).
3. Aachen D, Klaus A, Ternev R. CMMI: An integrated approach to process improvement. Practical introduction to the model. -M.: MFB. 2005. –330p.
4. Zharko E.F. Comparison of software quality models: analytical approach // XII All-Russian. Management Issues Meeting: Sat. of VSPU. 2014. –pp.4585-4594. URL: <http://vspu2014/ipu/ru/proceedings/prcdngs/4585.pdf> (accessed date: 01.05.2022).
5. Intel VTune Amplifier. 2017. URL: <https://software.intel.com/sites/default/files/managed/d7/ba/intel-brief.pdf> (accessed date: 07.05.2022).
6. Utkin L.V. Risk analysis and decision-making in case of incomplete information. -SPb: Science. 2007. -404p.
7. Mikoni S.V. Management decision theory. -SPb: Doe. 2015. -448p.
8. Glass R. Facts and misconceptions of professional programming [translated from English]. -SPb: Symbol-Plus. 2007. -240p.
9. Glukhova LV, Kaziev VM, Kazieva BV Analysis and synthesis of network business structures of the digital economy. Sat. proceedings of the scientific and practical conference "Law and Economics: Progress and Digital Technologies " / Ed. Yu.Altudov, A.Karlik, etc. 2019. -pp.143-147.

10. Hitchins D. What are the General Principles Applicable to Systems? – INCOSE INSIGHT. 2009. Vol.12. Iss.4. –pp.59-64.