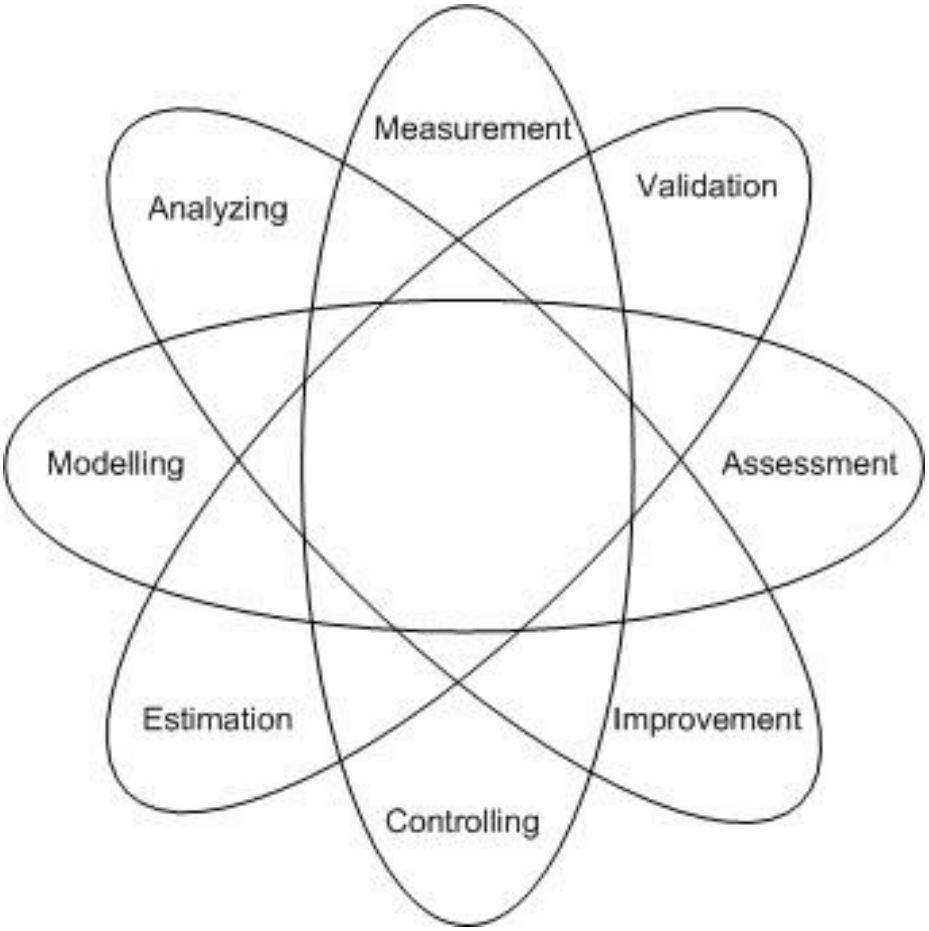
COSMIC

Volume 29, Number 2, October 2024

Software Measurement News

Journal of the Software Measurement Community



Editors:

Alain Abran, Jens Heidrich, Reiner Dumke, Andreas Schmietendorf

SM L@b

Université du Québec
École de technologie supérieure

DASMA

CONTENTS

Announcements 2

Andreas Schmietendorf: Workshop: Herausforderungen Low-Code orientierter KI-Ansätze 2
Reiner R. Dumke: SML@b News 4

Conference Reports 5
Alain Abran: IWSM/Mensura 2024 5

Community Reports 17
Jean-Marc Desharnais: COSMIC News 2024 17
NESMA News 2024 18
Luigi Buglione: GUFPI News 2024 19

News Papers 20

Sandro Hartenstein, Marc-Steven Fischer, Andreas Schmietendorf: Fallstudie zur KI-gestützten Anonymisierung deutschsprachiger Transkripte 20
Reiner R. Dumke, Michael Weiß: Could Software Quality influence Web Usage? 33

New Books on Software Measurement 41

Conferences Addressing Measurement Issues 45

Metrics in the World-Wide Web 48

Editors:

Alain Abran
Professor and Director of the Research Lab. in Software Engineering Management
École de Technologie Supérieure, 1100 Notre-Dame Quest, Montréal, Quebec, H3C 1K3, Canada, `alain.abran@etsmtl.ca`

Jens Heidrich
Professor on Applied Informatics, University of Mainz, Germany
Saarstr. 21. 55122 Mainz , Germany
`jens.heidrich@hs-mainz.de`

Reiner Dumke
Professor on Software Engineering,
University of Magdeburg, Faculty of Informatics, Germany,
`reiner.dumke@t-online.de, http://www.smlab.de`

Andreas Schmietendorf
Professor on Business Informatics, Hochschule für Wirtschaft und Recht
Alt-Friedrichsfelde 60 10315 Berlin, Germany,
`andreas.schmietendorf@hwr-berlin.de`

Editorial Office: University of Magdeburg, FIN, Postfach 4120, 39016 Magdeburg, Germany

Technical Editor: Dagmar Dörge

IFAF Institut für angewandte Forschung Berlin

Herausforderungen Low-Code orientierter KI-Ansätze

Öffentlicher Expertenworkshop in Anlehnung an die Themen des IFAF-Forschungsprojekts TAHAI (TrustAdHocAI – Details siehe QR-Code)

Ort:

*Fraunhofer IESE Kaiserslautern – **12. November 2024***
Fraunhofer-Platz 1, 67663 Kaiserslautern

Vorläufige Agenda der Vormittagssitzung (09:30 bis 12:30 Uhr):

Session 1 (09:30 bis 10:30 Uhr) – **Einführung und Impulsvorträge** (jeweils 10 min.)

Eröffnung des Workshops
Dr. Andreas Jedlitschka (Fraunhofer IESE), Prof. Dr. Jens Heidrich (HS Mainz)

Impuls zu den Hintergründen und Zielen des TAHAI-Projekts
Prof. Dr. Andreas Schmietendorf (HWR Berlin/Uni Magdeburg)

Impuls zu den KI-Ergebnissen im Mediationsdiskurs
Walter Letzel (TU Berlin)

Impuls zu den Herausforderungen des EU Artificial Intelligence Acts
Prof. Dr. Ralf Schnieders (HTW Berlin)

Session 2 (10:30 bis 11:15 Uhr) – **Sicherheitsaspekte von KI-Systemen**

Robuste KI-Systeme entwickeln mit dem Badgers Bad Data Generator
Dr. Julien Siebert (Fraunhofer IESE)

Unsicherheiten in KI-Systemen managen: Transparenz und Zuverlässigkeit durch Uncertainty Wrappers
Dr. Michael Kläs (Fraunhofer IESE)

15 min. Kaffeepause

Session 3 (11:30 bis 12:30 Uhr) – **Ansätze im Forstwirtschaftlichen Diskurs**

KI-basierte Totholzerkennung in der Forstwirtschaft (Online-Beitrag)
Prof. Dr. Erik Rodner (HTW Berlin)

KI-gestützte Förderung nachhaltiger Forstwirtschaft
Dr. Andreas Jedlitschka (Fraunhofer IESE)

60 min. Mittagspause

Vorläufige Agenda der Nachmittagssitzung (13:30 bis 15:00 Uhr):

Moderierte Diskussionsrunde zum Thema KI-Sicherheit:
Sandro Hartenstein (HWR Berlin/Uni Magdeburg)

Umgang mit Anforderungen an die KI-Sicherheit im Diskurs domänenspezifischer Anwendungsfelder. Schwerpunktmäßig soll dabei auf KI-Algorithmen für die Sprachverarbeitung (u.a. LLMs) aber auch die Bilderkennung (u.a. Convolutional Neural Networks) Bezug genommen werden.

Ausblick auf das Folgeprojekt TALCAI - Transfer Applied Low Code AI
Prof. Dr. Andreas Schmietendorf (HWR Berlin/Uni Magdeburg)

Bemerkung:

Die Teilnahme am Workshop ist kostenfrei, dennoch wird um eine Anmeldung (via Email: tahai@hwr-berlin.de) zum Zweck der besseren Organisation gebeten. Ggf. notwendige Änderungen der Agenda sind vorbehalten.

Veranstalter und Unterstützer:

GI-FG - “Measurement & Data Science” - Arbeitskreis ESAPI
https://fg-data-science.gi.de

ceCMG - „Central Europe Computer Measurement Group“
https://cecmg.de/

Ergebnissicherung:

Im Nachgang zum Workshop soll ein korrespondierender Tagungsband publiziert werden. Dieser wird sowohl Beiträge und Ergebnisdiskussionen des Workshops als auch Teilergebnisses des Projekts TAHAI enthalten.

Weiterführenden Informationen im Diskurs der Projektleitung TAHAI:
https://blog.hwr-berlin.de/schmietendorf

SML@b News

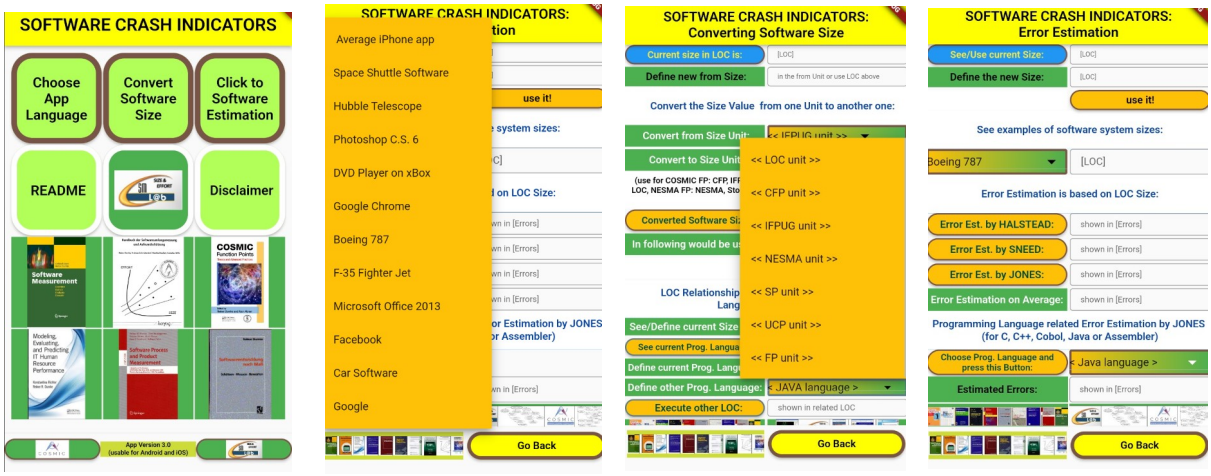
Reiner R. Dumke, University of Magdeburg, Germany

https://www.smlab.de/

The ***SoftwareCrash*** app (see (a)) enables the specification of a size measure for the software (such as function points, lines of code, etc. (see (c)) and shows a selectable error estimation based on empirical error metrics (see (d)). In particular, even very extensive software systems such as Boeing software, Google software, etc. can be selected and the number of errors estimated (see (b). This app can be ordered free at reiner.dumke@t-online.de.

This app is very well suited for the education about properties of software and their potential problems. For iPhone application use their emulator services.

SoftwareCrash Android App



(a) *(b)* *(c)* *(d)*

The wellknown SML@b Apps ***SoftwareLite*** (Software sizing based on the COSMIC FP method) and ***SoftwareExpert*** (COSMIC-based software sizing and software estimation) are available in the Google Play Store anymore.

Conference Report

Alain Abran, ETS Montreal, Canada

IWSM MENSURA

MONTRÉAL, SEPT. 30-OCT. 4, 2024

The Joint Conference of the 33rd International Workshop on Software Measurement (IWSM) and the 18th International Conference on Software Process and Product Measurement (MENSURA), will be held on September 30 – October 4, 2024 in Montréal, Canada.

In following are shown the conference paper abstracts of the IWSM/MENSURA 2024:

Technical Debt Measurement: An Exploratory Literature Review

Donatien Koulla Moulla[1], Ernest Mnkandla[1], Hayatou Oumarou[2] and Thomas Fehlmann[3]

[1)]University of South Africa, The Science Campus, Florida, 1710, South Africa
[2)]University of Maroua, Maroua, P.O. Box 46, Cameroon
[3)]Euro Project Office, Giblenstrasse 50, 8049 Zürich, Switzerland

Abstract

Measuring Technical Debt is important in guiding software development teams to make informed decisions and prioritize refactoring initiatives. This study presents an exploratory literature review of studies published between 2010 and 2023 to investigate the current state of Technical Debt measurement research. Through a set of four research questions, this study identifies the prevalent methodologies, metrics, and obstacles entailed in quantifying Technical Debt. Specifically, this study focuses on what is proposed to be measured through Technical

Debt, the measurement solutions proposed for measuring Technical Debt, and how these approaches categorize and evaluate various aspects of Technical Debt. By scrutinizing the diverse approaches and challenges, this exploratory literature review identifies gaps (and related issues) in Technical Debt measurement research and contributes to a nuanced understanding of Technical Debt measurement practices, offering insights into enhancing software sustainability and maintainability.

Keywords

Technical debt measurement, Technical debt quantification, Technical debt identification, Defect density, Maintainability, Exploratory Literature Review.

Quantum Software Sizing: Contemporary Interpretations and Approaches

Hassan SOUBRA

Ecole Centrale d'Electronique-ECE Lyon, 24 rue Salomon Reinach, 69007 Lyon

Abstract

Conventional software sizing approaches initially centered on metrics like lines of code, gradually transitioning to more refined measurements such as function points. However, these approaches could not be directly applicable to quantum software due to the fundamental differences between classical and quantum computing paradigms. In quantum software sizing, factors such as the number of qubits required, the depth of quantum circuits, the connectivity requirements of qubits, the error rates of quantum gates, and the complexity of the quantum algorithms play crucial roles.

Additionally, considerations such as the choice of quantum programming language, quantum hardware platform, and optimization techniques also impact the overall size estimation. This paper provides an overview of quantum software sizing, highlighting initial exploration and classification of sizing measurement concepts of Quantum Software.

Keywords

Quantum Software, Quantum Software Sizing, Quantum Software Engineering, Metrics, Measurement

Predicting Software Size and Effort from Code Using Natural Language Processing

Samet Tenekeci[1], Hüseyin Ünlü[1], Emre Dikenelli[1], Uğurcan Selçuk[1],

Görkem Kılınç Soylu[2] and Onur Demirörs[1]

[1)]İzmir Institute of Technology, Gülbahçe, İzmir, 35430, Türkiye
[2)]Izmir University of Economics, Balçova, İzmir, 35330, Türkiye

Abstract

Software Size Measurement (SSM) holds a crucial role in software project management by facilitating the acquisition of software size, which serves as the primary input for development effort and schedule estimation. However, many small and medium-sized companies encounter challenges in conducting objective SSM and Software Effort Estimation (SEE) due to resource constraints and a lack of expert workforce. This often leads to inaccurate estimates and projects exceeding planned time and budget. Hence, organizations need to perform objective SSM and SEE with minimal resources and without relying on an expert workforce.

In this research, we introduce two exploratory case studies aimed at predicting the functional size (COSMIC and Event-based size) and effort of software projects from the code using a deep-learning-based NLP model: CodeBERT. For this purpose, we collected and annotated two datasets consisting of 4800 Python and 1100 C# functions. Then, we trained a classification model to predict COSMIC data movements (entry, exit, read, write) and four regression models to predict Event-based size (interaction, communication, process) and effort. Despite utilizing a relatively small dataset for model training, we achieved promising results with an 84.5% accuracy for the COSMIC size, 0.13 normalized mean absolute error (NMAE) for the Event-based size, and 0.18 NMAE for the effort. These findings are particularly insightful as they demonstrate the practical utility of language models in SSM and SEE.

Keywords

Software size measurement, Effort estimation, Artificial intelligence, Natural language processing

Measurement In DevOps Standard: Proposed Improvements

Alexandra LAPOINTE-BOISVERT[1], Sylvie TRUDEL[1] and Jean-Marc DESHARNAIS[2]

[1)]Dept. of Computer Science, UQAM, Montreal, Canada
[2)]ÉTS, Montreal, Canada

Abstract

Context. DevOps is about improving the software delivery performance. Lord Kelvin stated: "*If you can not measure it, you can not improve it.*" For this reason, well-defined measurement is particularly important when using DevOps.

Method. This paper critically examines the measurement process outlined in the ISO/IEC/IEEE-32675 DevOps standard and proposes several improvements based on measurement-related standards (ISO-15939, ISO-25021, VIM, etc.).

Findings. The measures (base or derived) stated in the DevOps standard are neither characterized nor defined, as described in several standards. Some measures given as examples are ambiguous. Also, the DevOps standard claims to follow ISO/IEC/IEEE-12207, but some expected measures are missing (e.g. effort and size). This article proposes a way to define the different measures and explains why having effort and size measures is important in the context of DevOps.

Conclusion. The actual measurement process, as described in ISO/IEC/IEEE-32675, could be improved by better defining the base measures, including effort and size, using a measurement method, defining the derived measures using a measurement function, and avoiding ambiguous measures.

Keywords

ISO-32675, base measures, derived measures, DevOps, software measurement, software measurement process

A Hybrid Framework for COSMIC Measurement: Combining Large Language Models with a Rule-Based System

Safae Laqrichi

Arts et Métiers Campus of Rabat, Technopolis Rabat-Shore, Rabat 11103, Maroc

Abstract

Accurate Functional Size Measurement (FSM) is crucial for effective project management and resource allocation in software development. The COSMIC FSM provides a consistent and standardized method, allowing for objective comparisons and accurate estimation of effort. However, manual FSM is often time-consuming, error-prone, and subject to assumptions and human bias. Automating this process can greatly benefit organizations but introduces challenges associated with handling requirements expressed in Natural Language (NL).

In this paper, we explore the integration of Large Language Models (LLM) with rule-based systems to enhance the automation of the COSMIC Function Point Measurement. Our hybrid framework combines the contextual understanding and adaptability of LLMs, such as GPT-4, with the precision and consistency of rule-based engines crucial for tasks requiring strict adherence to rules and regulations. This approach enables a more robust analysis, processing of NL requirements, and precise application of the COSMIC measurement method.

The effectiveness and feasibility of this approach are demonstrated through a series of experiments conducted on COSMIC public case studies. Our results show a promising F1-score of 0.95 in identifying COSMIC key components using GPT-4, an F1-score of 0.97 in deducing data movements using the rule-based system, and a global error rate of only 6.6% in measuring COSMIC function points using the integrated framework. These results underscore the potential for more reliable and efficient system analysis through the synergistic capabilities of LLMs and rule-based systems.

Keywords

COSMIC FSM, Automation, LLM, GPT-4, ChatGPT, Rule-based-system, Prompting

Exploratory Review of Quantum Computing Software Requirements Specification and their Measurement

Tuna Hacaloglu[1,2], Hassan Soubra[3], Pierre Bourque[1]

[1)]École de Technologie Supérieure, 1100, rue Notre-Dame Ouest, Montréal, Québec, H3C 1K3, Canada
[2)]Atilim University, İncek, Gölbaşı, Ankara, 06830, Türkiye
[3)]École Centrale d'électronique (ECE), Lyon, France

Abstract

Quantum software sets itself apart from classical software owing to its powerful computational abilities rooted in entanglement and superposition. Unlike classical software, quantum software diverges notably across various dimensions, including computational models, hardware architectures, algorithms, deployment platforms, and problem domains. Quantum software is also often not standalone and interacts heavily with classical software, stressing the importance of carefully considering hybridization.

From a software engineering standpoint, researchers generally agree that a different approach is required for quantum software, and they advocate a Quantum Software Development Life Cycle (SDLC). This exploratory study briefly outlines the specifics of quantum software, overviews the proposed approaches regarding the software requirements of quantum software, and then reviews the current alternatives for measuring the functional size of quantum software. This study indicates that only a few papers in the literature discuss the requirements and functional size measurements of quantum software. Their results are also mostly conceptual and have not yet been empirically validated. Functional size measurement using quantum software remains an open area for further research.

Keywords

Software size measurement, quantum software, quantum software requirements

Quantifying the Value of Technical Debt Removal: A Proposed Model

Sylvie Trudel[1], Donatien Koulla Moulla[2], Ria Bakhtiani[3], Thomas Fehlmann[4], Frank Vogelezang[5], Hassan Soubra[6] and Shashank Patil[3]

[1)]Université du Québec à Montréal (UQAM), Montreal, Canada
[2)]University of South Africa, The Science Campus, Florida, 1710, South Africa
[3)]Capgemini, 2 Lal Bahadur Shastri Marg, 400079 Mumbai, India
[4)]Euro Project Office, Giblenstrasse 50, 8049 Zürich, Switzerland
[5)]COSMIC, Canada/The Netherlands
[6)]Ecole Centrale d'Electronique-ECE Lyon, 24 rue Salomon Reinach, 69007 Lyon, France

Abstract

Technical debt hinders many organizations from improving their software to meet user demands. Op- portunity: Therefore, it is important to have a model to quantify the value of technical debt removal. Methodology: The initial model was defined as the result of a brainstorming session with a team of experts. The model has then evolved from existing standards (literature), including functional size measurements (COSMIC).

Results: In this position paper, we propose a model for quantifying the value of technical debt (TD) removal and define angles for operationalizing future research. This value should be quantified into the effort required to remove TD, considering the type of TD. The model includes information to support the prioritization of TD removal activities based on their expected value, in- cluding i) partners and vendors, ii) information and technology, iii) value stream and processes, and iv) organization and people.

Keywords

Technical debt removal, Estimation model, Maintainability, Defect density

Measuring The Architecture Fit Of System Integration Designs

Konrad Nadobny[1,2,3], Andreas Schmietendorf[2,1]

[1)]Otto-von-Guericke-University Magdeburg, Faculty of Computer Science, Universitätsplatz 2, 39106 Magdeburg, Germany
[2)]Berlin School of Economics and Law, Department 2 - Cooperative Studies Business and Technology, Alt-Friedrichsfelde 60, 10315 Berlin, Germany
[3)]Bayer AG, Clinical Digital Innovation, Müllerstr. 178, 13353 Berlin, Germany

Abstract

This paper introduces a measurement model for integration architectures and describes the initial validation based on an industrial case study. The work is motivated by the rising challenge of transforming historically grown legacy system landscapes to reduce technical dept in large enterprises while mitigating limitations of competence sovereignty and sparse availability of integration architecture professionals. To address this challenge, the authors describe an easy-to-use approach to classify integration designs and compare their strengths and weaknesses to give guidance on which integration concepts fit best to the needs of the respective use case. These insights can be used on a the one hand to identify potentially problematic integrations that need to be refactored and on the other hand to provide requirements-based guidance for designing new integrations. With this the authors aim to provide enterprise architects with a tool to quantify the architecture fit of the integrations in their information systems landscape while supporting them in prioritizing their efforts. To define the measurement model, the authors analyse established possibilities to classify system architectures and define a taxonomy for integration architectures based on topologies, paradigms and technologies. After defining the subject of measurements, the authors make use of established standards for software quality measuring and define *Confidentiality*, *Availability*, *Integrity*, *Flexibility*, *Throughput* and *Latency* as the metrics to measure integration designs against non-functional requirements. To generate first empirical insights, the authors apply the measurement model to a real-world integration use case that is being refactored in context of a large industrial digital transformation project. By quantifying the gaps of the current integration design and comparing the results of the measurement model with the real-world observations, the authors gain first insights about the validity of their conceptualized approach. The results of the measurements imply, that the current integration using *File-Imports* has gaps in *Confidentiality*, *Integrity* and *Latency*. Switching to an integration using *Webhooks* or *Change-Data-Capture* would close those gaps and would conceptually result in a higher *Throughput* and a stronger *Integrity* and *Flexibility*. These outcomes are in line with the real-world observations, which implies that the conceptualized measurement model works and is fit-for-purpose. Even though the first results are very positive, the authors also identify weaknesses of the approach, so that further research is required for validate and refine the approach.

Keywords

integration architecture, software measurement, enterprise architecture, system integrations

Accessibility Evaluation Of Five Moroccan E-government Portals

Mohammed Rida Ouaziz, Laila Cheikhi[1], Ali Idri[1] and Alain Abran[2]

[1)]Software Project Management Team, ENSIAS, Mohammed V University in Rabat, Morocco
[2)]Department of Software Engineering Information Technology, École de Technologie Supérieure, Montréal, Canada

Abstract

In the present era, e-government portals are viewed as an effective means of providing online services to citizens. It is imperative for everyone to guarantee equitable access to these services, irrespective of their abilities or limitations. Hence, accessibility is vital for those with disabilities and advantageous for everyone. Web Content Accessibility Guidelines (WCAG) provide a comprehensive set of guidelines, in terms of success criteria, that are widely utilized in various tools to assess the accessibility of websites or portals. By following these guidelines, developers can ensure that their websites are accessible to everyone. This paper reports on a study of the accessibility evaluation of five Moroccan e-government portals using the AChecker tool with respect to WCAG. The aim was to prepare recommendations to improve the overall user experience across Moroccan e-government portals.

The evaluation results analysis showed that the portals have some common as well as different failed success criteria with different numbers of errors per WCAG principle and conformance level. In particular, all the portals exhibited varying degrees of adherence to the four principles, and most of the portals struggled to meet levels A and AA, and none of them achieve perfect conformance across all levels. To address this issue, a set of recommendations is provided for each portal to enhance accessibility. The lessons learned from this study can contribute to more inclusive e-government portals and provide insights for governments and designers to prioritize accessibility enhancements.

Keywords

Accessibility, e-government portals, evaluation tools, WCAG, disabilities, impairments

BotCFP: A Machine Learning based Tool for COSMIC Chatbots Sizing

Rahma Becha[1], Asma Sellami[1], Nadia Bouassida[1], Ali Idri[2,1] and Alain Abran[3,1]

[1)]Mir@cl Laboratory, University of Sfax, ISIMS, BP 242. 3021. Sfax-Tunisia
[2)]Software Project Management research Team, ENSIAS, Mohammed V University in Rabat, Morocco
[3)]Ecole de technologie supérieure, ETS, Montreal Université du Québec

Abstract

Chatbots are becoming more popular due to the number of functionalities they provide, the time savings and rapid responses in real time. Developing a chatbot requires defining a list of functional requirements upfront. Some of these requirements can be derived from other chatbots to discover and provide the required functionality, while the acquisition of other requirements is time-consuming and costly. Applying a standardized functional size measurement method, such as COSMIC Function Points – ISO 19761, to chatbots requirements is helpful in estimating the related project development effort and duration. This paper proposes an automated tool named BotCFP for generating the chatbots' sizes using the use-cases.csv dataset.

Three Machine Learning techniques (Support Vector Machine (SVM), Random Forest (RF), and Gradient Boosting Machine (GBM)) were used to determine the chatbots' sizes based on their functional processes (FP) name using three different text vectorization methods: TF-IDF, Word2Vec, and Bag of words. The best measurement results were provided by Random Forest using TF-IDF text vectorization method, deployed and used as an API in the BotCFP tool. The proposed tool allows users (project managers and developers) to determine the chatbot size from its FPs names before starting the development process.

Keywords

Chatbots, COSMIC Sizing, Machine Learning, TF-IDF, Word2Vec, Bag of Words, API

Software Change Size Measurement: An Exploratory Systematic Mapping Study

Tuna Hacaloglu[1,2], Neslihan Küçükateş Ömüral[3], Görkem Kılınç Soylu[4,5], Onur Demirörs[4]

[1)]École de Technologie Supérieure, 1100, rue Notre-Dame Ouest, Montréal, Québec, H3C 1K3, Canada
[2)]Atilim University, İncek, Gölbaşı, Ankara, 06830, Türkiye
[3)]Middle East Technical University, Üniversiteler, Çankaya, Ankara, 06800, Türkiye
[4)]Izmir Institute of Technology, Gülbahçe, Urla, İzmir, 35430, Türkiye
[5)]Izmir University of Economics, Sakarya Caddesi No:156, Balçova, İzmir, 35330, Türkiye

Abstract

Change in software projects can occur through various channels. Customers may request modifications or new features; appraisal activities such as reviews or testing may uncover issues that necessitate adjustments, or products may need to adapt to changes in their operating environment. Therefore, it is essential to assess these changes explicitly and objectively within the scope of software engineering activities. Specifically, quantifying change by measuring its size is crucial for successful management, as without a meaningful metric, it is impossible to accurately assess its impact on the project's effort, schedule, and cost. This study aims to explore the concept of change in software engineering literature, with a particular emphasis on the methods used to measure its size.

The study reveals that the current literature on this topic is still in its early stages and the measurement and estimation of changes remain challenging throughout both development and maintenance phases. According to the reviewed articles, size is primarily used for effort estimation. Various software artifacts from different stages of the Software Development Life Cycle (SDLC) serve as input for change measurement, highlighting the need for a versatile size measurement applicable across all SDLC phases. Most of the reviewed articles interpret change in the context of maintenance activities. This research sets a benchmark for the status of software size measures for software change and highlights related problems to suggest further research topics.

Keywords

Size measurement, software change, software maintenance

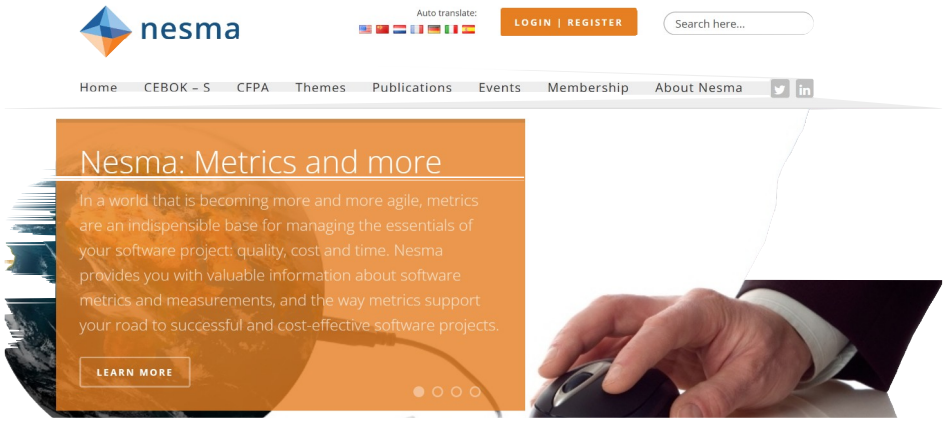
COSMIC News

Jean Marc Desharnais, COSMIC chairman

COSMIC courses are organized in Congo this year extending our software sizing community:

NESMA News

https:// nesma.org



CEBOK – S

Cost Estimation Body of Knowledge - Software

The value of Cost Estimation

One of the key roles of an IT manager is to make decisions and deliver the required needs and business value within the available budget. To make decisions the manager needs the right level of information. For a software project, the required information consist of the scope (although not detailed yet), the budget (for a next iteration) and the duration.

The Cost Estimator is the professional who can provide this information in cooperation with a solution architect who understands the solution and the project manager who understands the delivery approach. Having a professional for the Cost Estimator role ensures that cost estimates become more structured and understandable, more realistic and accurate, and the delivery of expected scope and time-frame more predictable.

The question that comes up regularly is: "Does this mean additional overhead?" The answer is: "No", a software delivery team already performs most of the required activities and provides the required information but because it's not professionalized, the information is often not a sound basis for decision making. The delivery team remains responsible for the estimate, but the Cost Estimator manages the estimation process like a project manager or scrum master manages the delivery or a solution architect manages the solution architecting process.

ICEAA SW will promote the professionalization of Software Cost Estimation and the Software Cost Estimation Body of Knowledge – Software. Members of ICEAA and Nesma will automatically be a member of ICEAA SW.

READ THIS BLOG ABOUT THE NEW FRONTIER IN SOFTWARE COST ESTIMATION

etc.

GUFPI News

Luigi Bulgione, GUFPI chairman

https://www.gufpi-isma.org/news/

etc.

Fallstudie zur KI-gestützten Anonymisierung deutschsprachiger Transkripte

Sandro Hartenstein, Marc-Steven Fischer, Andreas Schmietendorf

Hochschule für Wirtschaft und Recht Berlin
sandro.hartenstein@hwr-berlin.de, marc-steven.fischer@hwr-berlin.de, andreas.schmietendorf@hwr-berlin.de

1 Motivation

Im Diskurs der qualitativen Forschung, aber auch des analytischen Einsatzes von Sprachmodellen der künstlichen Intelligenz (KI) - u.a. Large Language Models (LLMs) - gewinnen transkribierte Gesprächsverläufe zunehmend an Bedeutung. Bei einer Transkription geht es zunächst um eine textliche Sicherung durchgeführter bzw. aufgezeichneter Konversationen. Neben den Wörtern und resultierenden Sätzen werden dabei auch Beziehungen zu den Urhebern (Rollen), ggf. auftretende Emotionen sowie Aspekte der Intonation festgehalten. Bei der Intonation kann es sich z.B. um die Betonung, die Lautstärke, einhergehende Tonhöhen, verwendete Pausen oder auch Schweigephasen handeln. Typische Beispiele für entsprechende Transkripte finden sich bei Gerichtsverhandlungen, medizinischen Konversationen oder bei den hier im Mittelpunkt stehenden Mediationssitzungen.

Derart erzeugte Transkripte implizieren zwangsläufig den Umgang mit personenbezogenen Daten. Es gilt die Privatsphäre und den Datenschutz der beteiligten Personen entsprechend den gesetzlichen Rahmenbedingungen (insbes. Datenschutz-Grundverordnung der Europäischen Union - EU DSGVO) zu gewährleisten. Sollen diese dennoch als Quelldaten für weiterführende Analysen eingesetzt werden, bedarf es der Anonymisierung durchgeführter Transkriptionen. Im Zusammenhang mit einer Anonymisierung geht der Personenbezug verloren bzw. dieser kann nur unter unverhältnismäßig hohem Aufwand wiederhergestellt werden. Insbesondere bei Mediationsanalysen, bei denen oft sensible und persönliche Informationen verarbeitet werden, ist die Anonymisierung ein unverzichtbarer Schritt, um das Vertrauen der Beteiligten zu wahren und die Integrität der Analyse sicherzustellen.

Auch der Prozess zur Anonymisierung kann von den Möglichkeiten des Einsatzes von Methoden der künstlichen Intelligenz profitieren. Dieser Beitrag adressiert diesbezüglich die folgenden Forschungsfragen:

1. Welcher NLP (Natural Language Processing)-Ansatz kann personenbezogene Daten in deutschen Transkriptionen mit ausreichend hoher Genauigkeit identifizieren.

2. Welche qualitativen Eigenschaften liefern vortrainierte KI-Modelle im Zusammenhang mit der Anonymisierung sensibler Transkripte im Diskurs der Mediation?

Die folgenden Abschnitte beschäftigen sich zunächst mit dem fachlichen Bezugsbereich der Anonymisierung und den grundsätzlich für eine Anonymisierung einsetzbaren KI-Ansätzen. Im Weiteren wird auf Details einer durchgeführten Teststellung von Frameworks und Tools zur Anonymisierung von Texten eingegangen. Der letzte Abschnitt fasst die Erkenntnisse zusammen und gibt einen Ausblick auf weiterführende Themen.

2 Grundlegende Aspekte der Anonymisierung

Eine praxisorientierte Arbeit zum Thema der Anonymisierung transkribierter Interviews findet sich unter [9] oder auch [11]. Dem entsprechend unterscheidet die Forschung im Allgemeinen drei Grade einer Anonymisierung:

- Formale Anonymisierung – entfernen direkter Identifizierungsmerkmale (Personennamen, Ortsangaben, …).
- Faktische Anonymisierung – Re-Identifizierung nur mit unverhältnismäßig hohem Aufwand möglich.
- Absolute Anonymisierung – hier ist es unmöglich, auf personenbezogen Daten zurückzuschließen.

Im Zusammenhang mit transkribierten Mediationssitzungen gilt es korrespondierende Entitäten und deren Beziehungen zu erkennen und in einem weiteren Schritt mit Hilfe von Zuordnungstabellen zu pseudonymisieren bzw. durch synthetische Daten zu ersetzen.

Sofern die Zuordnungstabellen vernichtet werden, kann von einer faktischen Anonymisierung ausgegangen werden. Im Einzelnen sehen die Autoren folgende Aspekte:

a. Ersetzen im Transkript auftretender Personennamen durch Pseudonyme oder auch Label bzw. Marken. In einem Transkript zur Mediation beziehen sich diese auf die Medianten (Streitparteien) und beteiligte Mediatoren.

b. Ebenso können die Beteiligten Personen (z.B. eigene Kinder) und deren Namen bzw. eingenommenen Rollen (z.B. Landesminister in …) innerhalb der Mediationssitzungen erwähnen, auch diese sind zu ersetzen.

c. Ersetzt werden sollten auch Bezüge zu Orts- und Straßennamen bzw. spezifischen geografischen Informationen (z.B. Haus am Westufer des Sees…), welche ggf. Rückschlüsse auf Personen zulassen.

d. Weiterhin sind Berufsbezeichnungen respektive einnehmbare Rollen innerhalb von Organisationen (z.B. Präsident…), welche eine Identifikation ermöglichen könnten, zu ersetzen.

e. Letztlich lassen auch Angaben zu ethnischer Verortung oder politischen und religiösen Überzeugungen Rückschlüsse auf einzelne Personen zu. Auch diese sind hinsichtlich der Anonymisierung zu berücksichtigen.

Jede Anonymisierung geht mit einem Informationsverlust einher. Dem entsprechend gilt es im Forschungsdiskurs zu prüfen, inwieweit anonymisierte Transkripte noch die verfolgten Analyseziele unterstützen bzw. diese ggf. auch einschränken. Zur Gewährleistung reproduzierbarer und erklärbarer Ergebnisse sollten durchgeführte Ersetzungen auch ohne die Zuordnungstabellen semantisch nachvollzogen werden können, ohne dabei allerdings auf konkrete Personen schließen zu können. In diesem Zusammenhang hat sich die Protokollierung des Anonymisierungsprozesses als sinnvoll erwiesen.

3 KI-basierte Anonymisierung

Named Entity Recognition (NER) zielt darauf ab, bestimmte Entitäten (Textdaten zu einer realen „Sache“ – häufig Subjekte) zu erkennen und zu klassifizieren. Die Erkennung von personenbezogenen Daten (Personally Identifiable Information – kurz PII) ist ein Teilbereich der NER, der sich auf die Identifizierung und Extraktion sensibler Informationen wie Personennamen, Adressen, Ortsangaben, Telefonnummern und anderen persönlichen Daten aus Texten konzentriert. Dem entsprechend erscheint es sinnvoll, NER-Lösungen für eine automatische Anonymisierung von Texten heranzuziehen.

Der Anonymisierungsprozess kann in folgender Weise abstrahiert werden (vgl. Abbildung 1).

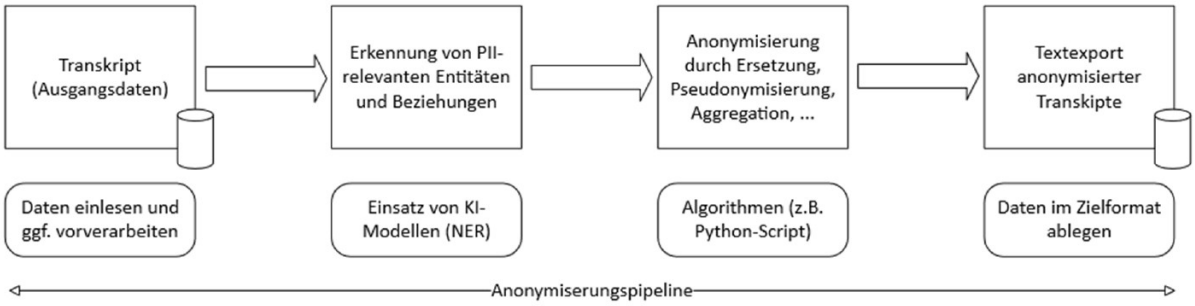


Abbildung 1: *Algorithmisch gesteuerter Prozess zur Anonymisierung*

Der Umgang mit deutschsprachigen Transkriptionen stellt für eine NER-Lösung eine besondere Herausforderung dar. Transkriptionen können Fehler und Abweichungen von der Standardsprache enthalten, was die Leistung von NER-Modellen beeinträchtigen kann. Ebenso können eingesetzte Modelle nur unzureichend auf selten auftretende Personennamen (z.B. bei Migrationshintergrund) trainiert sein und diese deshalb „übersehen“. Darüber hinaus bestehen häufig Schwierigkeiten bei der Identifikation kontextbasierter Entitäten, aber auch beim Umgang mit komplexen Satzstellungen und enthaltenen Füllwörtern, Hüsteln, Pausen und Geräuschen.

Zusammengefasst lassen sich die folgenden Vor- und Nachteile eines NER-Einsatzes aufzeigen:

Vorteile:

- *Effizienzsteigerung:* Automatisierung der Anonymisierungsprozesse und Steigerung der Durchsatzleistung.
- *Verbesserte Genauigkeit:* Einsatz moderner Techniken wie Deep Learning für eine präzisere Erkennung von Entitäten.
- *Flexibilität:* Anpassung an verschiedene Textarten und Fachdomänen durch Kombination eingesetzter Modelle (regelbasierten Systeme, statistischen Modellen oder Deep-Learning-Ansätze).

Nachteile:

- *Fehlerquellen*: NER-Systeme sind nicht fehlerfrei und können Entitäten falsch erkennen oder übersehen.
- *Kontextabhängigkeit:* Die Leistung von NER-Systemen kann stark vom Kontext des jeweiligen Textes abhängen.
- Bezüglich der Implementierung werden wörterbuch- und regelbasierte Ansätze, auch solche auf der Basis von Methoden des maschinellen Lernens unterschieden (vgl. [9] und [11]).

4 Konzeption durchzuführender Teststellungen

Zunächst soll kurz auf die Eigenschaften der zu anonymisierenden Ausgangsdaten, auf die mit Hilfe von Python implementierten KI-Teststellungen und deren Modelle sowie schließlich auf zwei werkzeugbasierte Anonymisierungslösungen eingegangen werden. Im nachfolgenden Kapitel 5 finden sich die Ergebnisse der durchgeführten Teststellungen.

4.1 Verwendete Ausgangsdaten

Bei den zu anonymisierenden Texten wurde auf drei verschiedene komplexe Beispiele zurückgegriffen. Mit Hilfe eines sehr einfachen und hinsichtlich der verwendeten Entitäten leicht verständlichen Demotextes sollte eine Einarbeitung, aber auch ein nachvollziehbarer Anonymisierungsprozess unterstützt werden. Konkret handelt es sich um den folgenden Satz:

> *„Bernd wohnt in Berlin. Max Mustermann arbeitet bei Bosch und wohnt in (Karlsruhe?) in der Auenstrasse 3. Max hat aber auch Verwandte in Berlin.“*

Darüber hinaus erfolgten lokale Tests (geschützter Bereich) mit Hilfe real transkribierter Mediationssitzungen entsprechend den folgenden Ausprägungen:

- Transkript 1: Abbildung einer Mediation mit 6 Sitzungen (1-2 Stunden), woraus ein Transkript von ca. 1,2 Mio. Zeichen (ca. 1,16 MByte) resultiert.

- Transkript 2: Abbildung einer Mediation mit 5 Sitzungen (1-2 Stunden), woraus ein Transkript von ca. 0,6 Mio. Zeichen (ca. 660 KByte) resultiert.

4.2 Eingesetzte NER Frameworks

Wie bereits angesprochen, bietet sich mit dem NER-Ansatz ein Verfahren zur Extraktion von Informationen aus Texten, das automatisch benannte Entitäten wie Personennamen, Ortsnamen und Organisationen identifiziert und klassifiziert. NER-Lösungen lassen sich wörterbuch-, regelbasiert und mit Hilfe von KI- bzw. Machine Learning (ML)-Algorithmen implementieren. Für den Test wurden „spaCy“ und „BERT“ genutzt:

SpaCy

Bei SpaCy handelt es sich um eine in Cythin (Python-Derivat) geschriebene Open-Source-Bibliothek [1]. Unterstützt wird die Entwicklung von NLP-Anwendungen, wobei insbesondere die Satz- oder Entity-Erkennung, Abhängigkeitsanalysen, Wort/Vektor-Transformationen aber auch Tokenization und Lemmatisation ermöglicht werden. Für den Benchmark wurde auf das Modell „de_core_news_lg“ (vgl. [6], [10], [12]) zurückgegriffen.

BERT/ DeBERTa (Transformer-basiertes NLP-Modell)

Bei BERT (Bidirectional Encoder Representations from Transformers) handelt es sich um einen als Open Source zur Verfügung gestellten NLP-Algorithmus (ursprünglich durch Google entwickelt). Das so genannte Transformer-Modell verwendet künstliche neuronale Netze für die Sprachverarbeitung. Die Stärke dieser Sprachmodelle bezieht sich auf das Erkennen von kontextbezogenen Zusammenhängen und Beziehungen zwischen Wörtern. Für die Tests wurde das auf personenbezogene Daten spezialisierte Modell „*deberta-v3-base_finetuned_ai4privacy_v2*“ verwendet, dabei handelt es sich um ein erweitertes BERT-Modell (Decoding-enhanced BERT with disentangled attention).

Weitere Informationen siehe [2], [3], [4], [5] und [13].

4.3 Tool-basierte Ansätze zur Anonymisierung (Privacy Frameworks)

Im Zusammenhang mit der ganzheitlichen Abbildung des Anonymisierungsprozesses finden sich Werkzeuge wie Microsoft Presidio [7] oder OpenRedact [8].

Microsoft Presidio

Zur Erkennung von Entitäten nutzt MS Presidio etablierte NER-Ansätze (spaCy, Transformer) aber auch weitere Erkennungsmethoden. Diese Modelle durchsuchen den Text und identifizieren Personennamen, Ortsnamen, Organisationen und andere relevante Informationen. Anschließend werden die erkannten Entitäten mit einer Wissensbasis namens Presidicis abgeglichen, um ihre Bedeutung zu kontextualisieren und zu präzisieren. Die Wissensbasis enthält Informationen über

verschiedene Entitytypen, wie z. B. die Position einer Person in einem Unternehmen oder die geographische Lage einer Stadt. [7]

OpenRedact

OpenRedact ist ein Tool zur Anonymisierung von deutschsprachigen Texten. Zu anonymisierende Dokumente können entweder über eine Webanwendung oder auch lokal als OnPremise verarbeitet werden. Im Zusammenhang mit den hier verwendeten Transkripten kommt aus Sicherheitsgründen ausschließlich eine lokale Verarbeitung in Frage. Für die Erkennung von Entitäten erfolgt zunächst eine Vorverarbeitung. Leerzeichen und Sonderzeichen werden entfernt und der Text wird in Kleinbuchstaben konvertiert. Danach nutzt OpenRedact Nerwhal (vgl. [14]), ein Framework, das auf spaCy und Stanza basiert, um Personennamen, Ortsnamen, Organisationen und andere sensible Informationen im Text zu erkennen. SpaCy zerlegt den Text in Token, d.h. in einzelne Wörter oder Satzzeichen. Stanza, eine Komponente von Nerwhal, nutzt die Token, um die Entitäten im Text zu erkennen und zu klassifizieren. Stanza verwendet dafür verschiedene Verfahren, z.B. maschinelles Lernen und regelbasierte Mustererkennung.
OpenRedact anonymisiert die erkannten Entitäten automatisch. Die Anonymisierungsmethode kann vom Nutzer konfiguriert werden. Er kann z.B. die Option wählen, die Entitäten durch Platzhalter zu ersetzen oder synthetische Daten zu generieren. Abschließend stellt OpenRedact das anonymisierte Dokument dem Nutzer zur Verfügung. Das Dokument kann in verschiedenen Formaten (pdf, docx, txt) weiterverarbeitet werden. [8]

4.4 Verwendete Testumgebung

Der Prototyp zum Vergleich der Fähigkeiten der unterschiedlichen Anonymisierungsansätze basiert auf Python und wurde in einer Jupyter Notebook-Umgebung entwickelt (vgl. Anlage 2). Entsprechend der Sensibilität der eingesetzten Transkripte erfolgte die Nutzung der Teststellung ausschließlich auf lokalen Rechnern (on-premise). Im Fall der programmiertechnischen Verwendung der NER-Frameworks erfolgte die Anonymisierung mit Hilfe von Zuordnungstabellen unter Einsatz von pseudonymisierten bzw. synthetischen Daten.

Die wichtigsten Qualitätsmerkmale für die Anonymisierung sind eine hohe Erkennungsrate von sensiblen Daten sowie deren klare Trennung vom übrigen Text. Weiterhin soll der Prototyp problemlos mehr als eine Million Zeichen verarbeiten können. Zusätzlich spielen Ressourcenverbrauch und Laufzeit eine Rolle. Die Laufzeit wird dabei auf einem System mit AMD R4850U Prozessor und 32GB RAM getestet. Für die Vergleichstests wurde auf die unter Abschnitt 4.1 bereits beschriebenen Testdaten zurückgegriffen. Mit Hilfe der Tests sollte ermittelt werden, ob sich die eingeführten NER-Ansätze grundsätzlich dafür eignen, lange Transkriptionen zuverlässig lokal zu anonymisieren.

5 Ausgewählte Ergebnisse der prototypischen Tests zur Anonymisierung

Der Abschnitt dokumentiert die prototypische Evaluation der Anonymisierung mit den in Abschnitten 4.2 und 4.3 vorgestellten NER-Lösungen (Frameworks/Tools) unter Verwendung der unter 4.1 aufgezeigten Testdaten. Hierzu wird im ersten Teil ein prototypischer Vergleich der Ansätze aufgezeigt und das beste Framework für den Anwendungsfall konfiguriert und angewendet.

5.1 Berücksichtigte Bewertungskriterien

Die verwendeten Bewertungskriterien wurden sowohl messtechnisch als auch auf der Grundlage eine Selbstbewertung ermittelt. Die gemessenen (ausgezählten) Kriterien beziehen sich auf:

Aspekte der Genauigkeit:

- Entity Recognition (ER): Wie genau erkennt das Modell Entitäten (z.B. Namen, Orte, Organisationen)?
- Entity Linking (EL): Wie gut kann das Modell erkannte Entitäten mit Datensätzen und Knowledge-Graphen abgleichen?
- False Positives (FP): Wie viele falsch positive Entitäten werden erkannt?
- False Negatives (FN): Wie viele Entitäten werden nicht erkannt?

Mit Hilfe einer durch die Testpersonen durchgeführten Selbstbewertung (ordinale Skala von 1 (als schlecht) bis 6 (als gut) wurde auf folgende Sachverhalte eingegangen:

Privacy-Schutz (zusammenfassende Selbstbewertung):

- Minimierung: Entfernt das Modell alle relevanten personenbezogenen Daten (PII)?
- Anonymisierungsgrad: Wie gut werdeSn PII-Entitäten anonymisiert (z.B. durch Maskierung, Ersetzung)?
- Risikobewertung: Berücksichtigt das Modell das Risiko einer Re-Identifizierung?

Effizienz (zusammenfassende Selbstbewertung):

- Geschwindigkeit: Wie schnell kann das Modell Text verarbeiten?
- Speicherverbrauch: Wie viel Speicherplatz benötigt das Modell?
- Skalierbarkeit: Kann das Modell auf große Datenmengen skaliert werden?

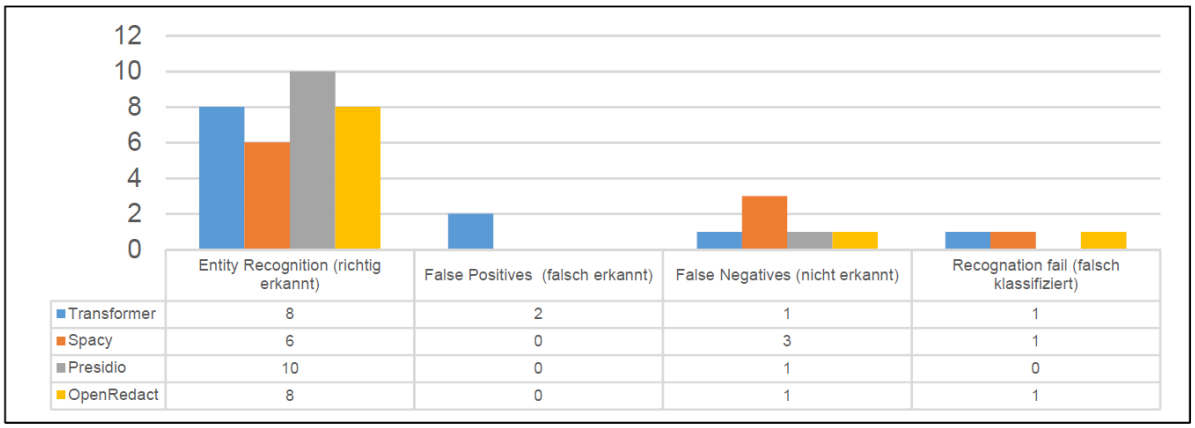
Usability (zusammenfassende Selbstbewertung):

- Integration: Wie einfach lässt sich das Modell in bestehende Systeme integrieren?

- Konfiguration: Wie einfach ist es, das Modell für verschiedene Anwendungsfälle anzupassen?
- Dokumentation: Ist die Dokumentation des Modells klar und verständlich?

5.2 Beispiele für gemessene Ergebnisse (Demotext)

Abbildung 2 bietet einen detaillierten Einblick in die Leistungsfähigkeit der vier untersuchten NER-Ansätze (Transformer - DeBERTa, SpaCy, Presidio und OpenRedact) bei der Identifizierung von Entitäten in Textdaten.



	Entity Recognition (richtig erkannt)	False Positives (falsch erkannt)	False Negatives (nicht erkannt)	Recognation fail (falsch klassifiziert)
Transformer	8	2	1	1
Spacy	6	0	3	1
Presidio	10	0	1	0
OpenRedact	8	0	1	1

Abbildung 2: *Vergleich der Genauigkeit*

Die Analyse der Ergebnisse zeigt deutliche Unterschiede in der Genauigkeit und Zuverlässigkeit der einzelnen Modelle.

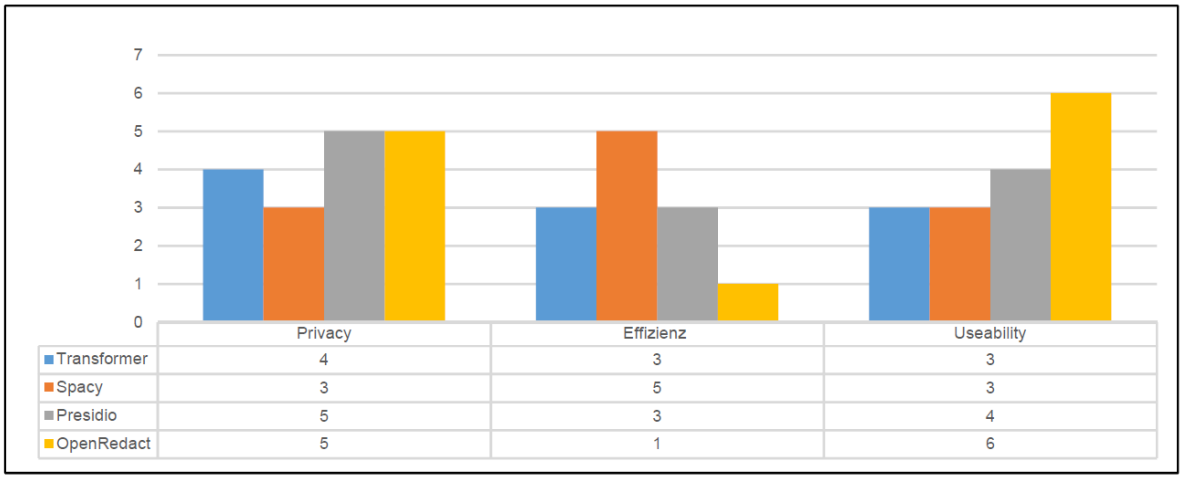
Entitätserkennung: Presidio erwies sich als das leistungsstärkste Framework in Bezug auf die korrekte Identifizierung von Entitäten. Transformer und OpenRedact zeigten ebenfalls eine hohe Trefferquote und lagen dicht hinter Presidio. SpaCy hingegen hatte die größte Schwierigkeit, Entitäten korrekt zu erkennen.

Fehlklassifikationen: Sowohl False Positives als auch False Negatives wurden analysiert. Presidio und OpenRedact zeichneten sich durch die geringste Anzahl an Fehlklassifikationen aus, was auf eine höhere Präzision hinweist. Transformer und SpaCy tendierten dazu, häufiger falsche Positives zu produzieren.

Um eine umfassendere Bewertung der Modelle zu ermöglichen, könnte man die Genauigkeit (Accuracy) berechnen, die den Anteil der korrekt klassifizierten Entitäten an der Gesamtzahl der Entitäten angibt. Allerdings ist diese Metrik in unausgewogenen Datensätzen, in denen beispielsweise viele negative Beispiele vorliegen, oft nicht aussagekräftig genug. Basierend auf den vorliegenden Ergebnissen zeigt sich Presidio als das vielversprechendste Framework für die Entitätserkennung.

Es kombiniert eine hohe Genauigkeit bei der Identifizierung von Entitäten mit einer geringen Anzahl an Fehlklassifikationen. Transformer und OpenRedact bieten ebenfalls solide Ergebnisse, weisen jedoch in einigen Kategorien geringfügig schlechtere Leistungen auf. SpaCy fällt im Vergleich zu den anderen Frameworks deutlich zurück, insbesondere bei der korrekten Erkennung von Entitäten.

5.3 Beispiele für selbstbewertete Ergebnisse (Demotext und Transkripte)



	Privacy	Effizienz	Useability
Transformer	4	3	3
Spacy	3	5	3
Presidio	5	3	4
OpenRedact	5	1	6

Abbildung 3: *Bewertung der Privacy, Effizienz und Usability*

Abbildung 3 zeigt die vergebenen Bewertungen bezüglich der vier im Test berücksichtigten NER-Ansätze in Bezug auf den Datenschutz (privacy), die Leistungsfähigkeit (efficiency) und die Benutzerfreundlichkeit (usability).

Datenschutz: Presidio und OpenRedact zeichnen sich durch einen besonders guten Schutz sensibler Daten aus. Transformer und SpaCy bieten ebenfalls einen akzeptablen Datenschutz, sind jedoch etwas weniger robust in diesem Bereich.

Effizienz: SpaCy ist das mit Abstand effizienteste Framework, gefolgt von transformer-orientierten Ansatz DeBERTa. Presidio und insbesondere OpenRedact sind weniger effizient. Auf diesen Sachverhalt soll im folgenden Abschnitt noch etwas detaillierter eingegangen werden.

Benutzerfreundlichkeit: OpenRedact überzeugt durch seine intuitive Benutzeroberfläche und ist somit das benutzerfreundlichste Framework. Presidio folgt dicht auf und ist ebenfalls einfach zu bedienen. Transformer und SpaCy sind etwas weniger intuitiv, was der programmiertechnischen Verwendung geschuldet ist.

5.4 Betrachtungen zur Effizienz bzw. dem Laufzeitverhalten

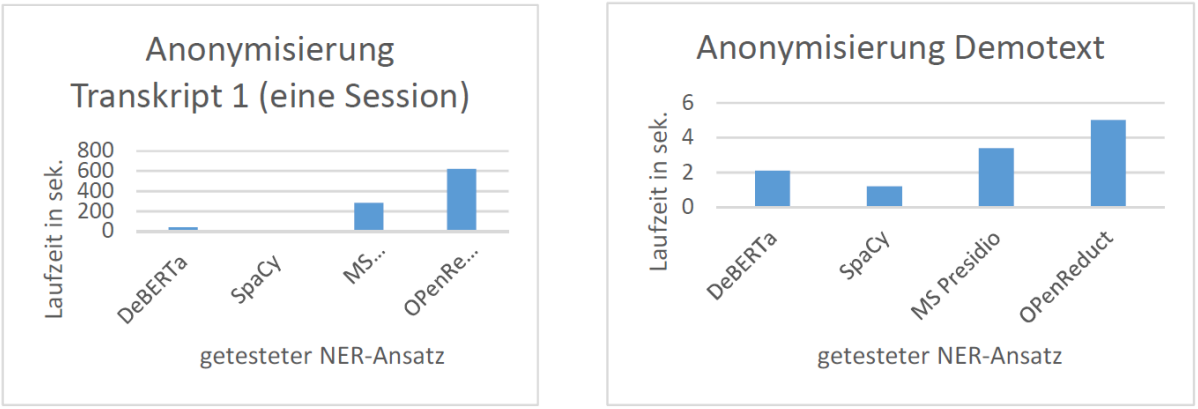


Abbildung 4: *Laufzeitverhalten*

Die Anonymisierung des Demotextes kann selbstverständlich nicht mit der eines realen Transkripts verglichen werden. Entsprechende Laufzeiten fallen aufgrund der unterschiedlichen Komplexitäten deutlich auseinander. Beim der Ermittlung der Laufzeit wurde nur eine Session (1-2 Stunden) einer realen Mediation berücksichtigt, d.h. der Datenumfang lag bei ca. 0,2 Mio. Zeichen bzw. bei 0,2 MByte. Darüber hinaus muss beim Vergleich die Genauigkeit der durchgeführten Anonymisierung berücksichtigt werden. Hier zeigen MS Presidio und OpenReduct deutliche Vorteile. Im Zusammenhang mit dem Transkript 1 (alles Sitzungen) erfolgten bei MS Presidio 1044 Ersetzungen identifizierter Entitäten in 18 Minuten, beim Transkript 2 ergaben sich 569 Ersetzungen innerhalb von 11 Minuten. Beim Einsatz von OpenReduct kam es allerdings wegen der Speicherlimitierung zu einem Abbruch der Anonymisierung (timeout).

Eine Zusammenfassung der Ergebnisse kann in Anlage 1 eingesehen werden.

5.5 Schlussfolgerungen für die Verarbeitung transkribierter Mediationen

MS Presidio wurde im Vergleich als bestes Framework in den untersuchten Kriterien ermittelt. Da transkribierte Mediationssitzungen bezüglich des Datenschutzes als besonders sensibel einzuschätzen sind, sollte trotz der guten Ergebnisse eine manuelle Optimierung der NER-Analyse vorgenommen werden. Als sinnvoll erwiesen hat sich die Verwendung manuell erstellter Listen. Diese sollten auf erlaubte Einträge, die nicht als Entitäten erfasst werden sollen, auf bekannte Namen, die als Personen, bzw. bekannte Orte, die als Lokation erkannt werden sollen, eingehen. In der Anonymisierungsphase gilt es, die gefundenen Entitäten im Text zu verschleiern. Hierzu wurde eine Pseudonymisierung verwendet mit dem Typ und einem Zähler, sodass mehrfach auftretende Entitäten auch nach der Anonymisierung wiederkehren, um den Inhalt möglichst wenig zu verfälschen. Die entstehende Zuordnungstabelle existiert nur temporär während der der Anonymisierung eines Falles und wird am Ende verworfen. Diese ist auch sitzungsübergreifend.

6 Reflektion und Ausblick

In der Untersuchung der einzelnen Ansätze wurde deutlich, dass eine Methode für die Erkennung nicht ausreichend ist. Erst die Frameworks mit mehreren Erkennungsmethoden liefern gute Ergebnisse. Das liegt vermutlich an den Besonderheiten der Mediationstranskriptionen, z.B. Klammern, Pausen und Metainformationen, und auch an der Zielstellung, die Sinnhaftigkeit der Handlungen nicht zu verlieren. Dem entsprechend sollten z.B. nicht alle Orte als personenbezogenen Daten identifiziert werden, sondern nur, wenn sie tatsächlich einen Personenbezug implizieren. Ein Urlaubszielort sollte beispielsweise nicht maskiert werden, eine große Stadt ebenso wenig, eine Kleinstadt oder ein Dorf sollten allerdings maskiert werden.

Entsprechend der prototypischen Implementierungen können die unter Kapitel 1 aufgeworfenen Forschungsfragen in folgender Weise beantwortet werden:

- Die Kombination von mehreren NLP- (insbesondere NER) Ansätzen mit aktuellen vortrainierten Modellen, Listen und RegEx bringt gute Ergebnisse.
- Derzeit ist die Erkennungsqualität in der speziellen Domäne der sensiblen Mediationen noch nicht ausreichend.

Die Ausführungen zur tatsächlichen Anonymisierung der Transkriptionen wurde manuell um bekannte Namen und Orte sowie um eine Erlaubnisliste erweitert. Dies war der pragmatische Weg, um die geforderte Qualität der Anonymisierung zu erreichen. Zukünftig können die Trainingsfunktionalitäten der ML-Komponente in der Erkennungsphase genutzt werden, um die notwendige manuelle Nachfilterung zu minimieren.

7 Quellenverzeichnis

[1] EXPLOSIONAI GMBH: *spaCy : Industrial-Strength Natural Language Processing*. URL https://spacy.io

[2] *deberta-v3-base_finetuned_ai4privacy_v2*. URL https://huggingface.co/Isotonic/deberta-v3-base_finetuned_ai4privacy_v2 – Überprüfungsdatum 2024-09-06

[3] PENGCHENG HE ; XIAODONG LIU ; JIANFENG GAO ; WEIZHU CHEN: DEBERTA: DECODING-ENHANCED BERT WITH DISENTANGLED ATTENTION. IN: INTERNATIONAL CONFERENCE ON LEARNING REPRESENTATIONS, 2021

[4] PENGCHENG HE ; JIANFENG GAO ; WEIZHU CHEN: DEBERTAV3: IMPROVING DEBERTA USING ELECTRASTYLE PRE-TRAINING WITH GRADIENT-DISENTANGLED EMBEDDING SHARING. 2021

[5] ELOV, B.B. ; KHAMROEVA, S.M. ; XUSAINOVA, Z.Y.: THE PIPELINE PROCESSING OF NLP. IN: E3S WEB OF CONFERENCES 413 (2023), S. 3011

[6] SILVA, P. ; GONCALVES, C. ; GODINHO, C. ; ANTUNES, N. ; CURADO, M.: USING NLP AND MACHINE LEARNING

TO DETECT DATA PRIVACY VIOLATIONS. IN: IEEE INFOCOM 2020 - IEEE CONFERENCE ON COMPUTER COMMUNICATIONS WORKSHOPS (INFOCOM WKSHPS). PISCATAWAY, NJ : IEEE, 2020, S. 972–977

[7] MICROSOFT: PRESIDIO : DATA PROTECTION AND DE-IDENTIFICATION SDK. URL HTTPS://MICROSOFT. GITHUB.IO/PRESIDIO/

[8] OPEN KNOWLEDGE FOUNDATION DEUTSCHLAND E. V.: OPENREDACT. URL HTTPS://OPENREDACT.ORG/PROTOTYPEFUND

[9] MEYERMANN, A.; PORZELT, M.: HINWEISE ZUR ANONYMISIERUNG VON QUALITATIVEN DATEN. IN: FORSCHUNGSDATEN BILDUNG INFORMIERT NR.1 (2014), FORSCHUNGSDATENZENTRUM (FDZ) BILDUNG AM DIPF DEUTSCHES INSTITUT FÜR INTERNATIONALE PÄDAGOGISCHE FORSCHUNG (HRSG.). FRANKFURT AM MAIN. ONLINE ABRUFBAR UNTER: HTTPS://WWW.FORSCHUNGSDATEN-BILDUNG.DE/FILES/FDB-INFORMIERT-NR-1.PDF

[10] TRAINED MODELS & PIPELINES, URL HTTPS://SPACY.IO/MODELS/DE

[11] WERNER, C., MEYER, F., & BISCHOF, S. (2023). GRUNDLAGEN, STRATEGIEN UND TECHNIKEN DER ANONYMISIERUNG VON TRANSKRIPTEN IN DER QUALITATIVEN FORSCHUNG: EINE PRAXISORIENTIERTE EINFÜHRUNG. FORUM QUALITATIVE SOZIALFORSCHUNG / FORUM: QUALITATIVE SOCIAL RESEARCH, 24(3). HTTPS://DOI.ORG/10.17169/FQS-24.3.4067

[12] SPACY: DIE OPEN-SOURCE PYTHON-BIBLIOTHEK FÜR NLP, HTTPS://DATASCIENTEST.COM/DE/SPACY-OPENSOURCE-BLIBLIOTHEK, FEBRUAR 2023

[13] LUBER, S.: DEFINITION WAS IST BERT?, HTTPS://WWW.BIGDATA-INSIDER.DE/WAS-IST-BERT-A-1116088/, MAI 2022

[14] NERWHAL: A MULTI-LINGUAL SUITE FOR NAMED-ENTITY RECOGNITION, URL HTTPS://MEDIUM.COM/@OPENREDACT/NERWHAL-A-MULTI-LINGUAL-SUITE-FOR-NAMED-ENTITY-RECOGNITION-D3AC6BEB547

ANLAGE 1 – TABELLARISCHER VERGLEICH

	Transformer Pipeline	SpaCy	MS Presidio	OpenReduct
Pseudonymisierter Demotext mit Markierung für Ungenauigkeiten	<PER 1> wohnt in <LOC 1>. <PER 2> <PER 1>er<PER 3><PER 4> arbeitet bei <ORG 1> und wohnt in <LOC 2> in der Auestrasse 3. <PER 3> hat aber auch Verwandte in <LOC 3>	<PER 1> wohnt in <LOC 1> . <PER 2> <PER 3> arbeitet bei <ORG 1> und wohnt in Karlsruhe in der Auestrasse 3. Max hat aber auch Verwandte in <LOC 2>	<PERSON_2> wohnt in <LOCATION_0>. <PERSON_1> arbeitet bei <ORGANIZATION_0> und wohnt in (<LOCATION_2>?). in der <LOCATION_1> 3. <PERSON_0> hat aber auch Verwandte in <LOCATION_0.>	<PERSON 1> wohnt in <LOCATION 1>. <PERSON 2> arbeitet bei <PERSON 3> und wohnt in (<LOCATION 2>). in der <LOCATION 3> 1 <PERSON 4> hat aber auch Verwandte in <LOCATION 1>.
Genauigkeit (Qualität) Demotext	einige false Positives	einige nicht erkannte Entitäten	Sehr genau	Sehr genau
Laufzeit Demotext	2.1s	1.2s	3.4	5s
Laufzeit des realen Transcripts zur Mediation (incl. read and write file) - nur eine Session	40.3s	4.1s	282.0s	622s
Genauigkeit Mediation ZeroShot (qualitativen Fehleranalyse)	einige False Positives	einige False Negatives	Sehr genau	Sehr genau
Transkript 1 (alle Sessions)			1044 Ersetzungen in 18min	Fail (timeout)
Transkript 2 (alle Sessions)			569 Ersetzungen in 11min	Fail (timeout)

ANLAGE 2 – EXEMPLARISCHER AUSZUG ZUR PYTHON-IMPLEMENTIERUNG

Die Python basierte Implementierung mehrerer Prototypen zur Anonymisierung erfolgte mit Hilfe von jeweils ca. 500 LoC. Im Detail wurden folgende Funktionen für den Prozess der Anonymisierung umgesetzt:

- Vorbereiten der Ausgangsdaten - Steuerzeichen entfernen, UTF8 (rtf-Format)
- PII Analyse und Anonymisierung
  - Laden des Transkripts
  - Konfiguration der NER-Analyse
  - Laden des Anonymizers und Konfiguration
  - Ausführen der NER-Analyse und Anonymisierung
- Speichern der anonymisierten Transkriptionen

Die folgende Abbildung zeigt einen kleinen Ausschnitt der Implementierung, wobei es insbesondere um das Laden des Transkripts und die Konfiguration der NER-Analyse (erste Zeilen der Implementierung) geht.

PII Analyse und Anonymisierung

Die Analyse erfolt mit den Frameworks Spacy, Transformer und MS Presidio mit verschiedenen Identifikationsmethoden. Zudem werden Manuell Identifizierte Eigennamen und Orte als Liste hinzugefügt, da Testreihen bestimmte Eigenamen unzuverlässig erkannt haben. Hier ist Finetuning und Training der Modelle möglich, allerdings nicht Fokus dieser Untersuchung.

```
import os
text_list = []
input_dir = r'ppm\F5'

for i in range(1, 2):  # Annahme: _S1.txt bis _S6.txt
    filename = f'_S{i}.txt'
    input_file_path = os.path.join(input_dir, filename)
    print(f"Transkript : {input_file_path} geladen")
    with open(input_file_path, 'r', encoding='utf-8') as file:
        text = file.read()
        text_list.append(text)
```

```
Transkript : ppm\F5\_S1.txt geladen
```

Konfiguration der Analyse NER

```
import spacy
from presidio_analyzer import AnalyzerEngine, RecognizerRegistry, PatternRecognizer
from presidio_analyzer.nlp_engine import NlpEngineProvider
from transformers import pipeline

from presidio_anonymizer import OperatorConfig
from presidio_anonymizer.operators import Operator, OperatorType
from typing import Dict


# SpaCy-Model laden
SPACY_MODEL = "de_core_news_lg"
nlp = spacy.load(SPACY_MODEL)


# Transformer-Model laden
TRANSFORMER_MODEL = "domischwimmbeck/bert-base-german-cased-fine-tuned-ner"
nlp_transformer = pipeline("ner", model=TRANSFORMER_MODEL)
```

Could Software Quality influence Web Usage?

- a Web Site Analysis based Evaluation -

Reiner R. Dumke, Michael Weiß

SML@b Magdeburg, Germany, https://softmeasure.de

Abstract

The following article is dedicated to a classic metrics application for determining web quality, for example, and its possible effects on the real application or use of the respective web contents.

It is a nostalgic look at the quality of software systems under changing user requirements and interests. During the 'inventory' of our measurement tools that are still valid or working today, we came across a very powerful and modern tool of its time (intelligent multitasking of complex website evaluations), which still works reliably today.

We first carried out the same measurements and evaluations as 20 years ago and then turned our attention to the question of whether such classic web system quality metrics are still relevant for the use of these systems today.

On the one hand, we show the obvious decline in website quality in the classic sense and, on the other hand, the special interests that are dominated not least by the all-dominant click rate metric.

1 Some Historical Aspects of Metrics Applications

In my experience, the 70s and 80s were mainly characterised by constructive software engineering, in particular by universal programming languages, generators and general macro technology and generative tools. My academic qualifications also focused on this area [Dumke 1988].

As early as the end of the 1970s and into the 1990s, analytical consideration as empirical software engineering with the evaluation of the complexity and efficiency of programming and the application of corresponding metrics or measures for all possible program components and paradigms came to the focus ([Zuse 1998], [Sneed 2017]).

On this basis, entire measurement and quality processes were developed, which characterised the IT world as methodologies and standards ([Abran 2010], [Bundschuh 2008]). On this approaches of extensive evaluations and measurement

databases, fundamental quality assurance for extensive software systems is therefore state of the art [Jones 2010].

Today, measurement-based development is essentially installed for controlled system development, which is essential, especially in times of agile ad hoc development [Ebert 2007].

2 The Rediscovery of our Web Measurement Suite

During the development of our SML@b, a large number of measurement tools were implemented ([Dumke 2003], [Dumke 2014]):

- for determining complexity metrics for various programming languages (Pascal, COBOL, PL/1, Prolog, Java, Smalltalk, Kotlin, Dart, etc.),
- as a set-up of measurement services, such as the Java measurement service, where anyone can measure their Java program and which, for example, shows the simple number of Java programming elements, such as methods (approx. 20000) or thousands of Java classes (so who knows Java as a whole?),
- or agent-based measurement tools for the knowledge-based analysis of web technologies or also for the quality of web pages or web content of all kinds according to a multiprocessing principle.

We rediscovered a remarkable complex tool for analyzing and evaluating websites, which was created as part of a diploma thesis.

This diploma thesis deals with the development of a Web Quality Assurance System which enables various user groups to compare web pages and to evaluate their quality [Weiß 2004].

Therefore aspects of web quality are examined from the point of view of different interest groups. It is necessary to use an integrated system which can be customized for the requirements of respective users and allows for a comfortable retrieval of information from web pages.

The following illustration shows the website of this tool in its functional overview.

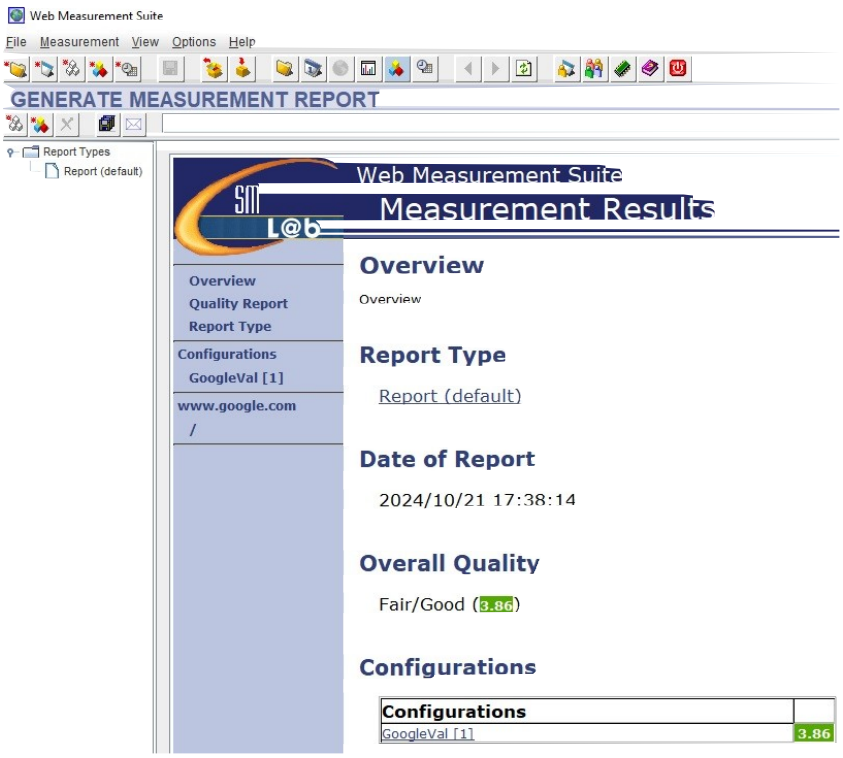


Fig. 1: *Working screen of our Web Measurement Suite*

A total of 52 software metrics and measures were defined for the evaluation of the following properties of web content:

- the performance of a web page (response time and their accessibility),
- the complexity of the web structure (number of internal and external links and their call errors, etc.),
- the number of absolute and relative web structuring for the integrated components such as images, documents and so-called frames,
- the page size of the HTML source code as well as information on nesting and linguistic diversity,
- the complexity of the web content according to the nesting depth of tables, frames and the like.

For the quality assessment, a total of 75 mostly ordinary scaled quality measures were determined on the basis of these metrics and their measured values, the scaling of which was based on the web usage experience at the time, which was certainly subjective but generally perceived as useful.

The concept therefore consists of determining the measured value and calculating a single quality score for the respective website with different analyzed and adjustable link depths (from 1 as best and 10 as worst). This principle is therefore the same as for the ranking of a scientific degree with the associated advantages and disadvantages.

One result at the time (exactly 20 years ago) was the following table. It primarily analyzed typical web applications and the company's own web presence. Please note that Google was one of many web search engines and Facebook and Amazon were still in the development phase.

Web Source	Quality	Good (in percent)	Bad (in percent)	Fair (in percent)
Wikipedia	*2.61 (Good)*	*81*	*5*	*14*
University of Magdeburg	*3.61 (Good)*	*79*	*10*	*11*
W3C	*4.02 (Fair/Good)*	*73*	*14*	*13*
Germany Government	*5.62 (Fair)*	*55*	*18*	*27*
University of Stanford	*6.31 (Fair)*	*61*	*19*	*20*
eBay	*6.55 (Fair/Poor)*	*54*	*29*	*17*
Amazon	*6.77 (Fair/Poor)*	*55*	*27*	*18*

Tab. 1: *Web Site Evaluation of chosen Web Contents at 2004*

The interest in whether such tools are still running or working today ultimately led to the following table in which we have included a few more websites in the evaluation.

Web Source	Quality	Good (in percent)	Bad (in percent)	Fair (in percent)
GI-FG Data Science & Measurement	*5.82 (Fair)*	*67*	*17*	*16*
University of Magdeburg	*6.04 (Fair)*	*57*	*23*	*20*
SML@b	*6.13 (Fair)*	*65*	*26*	*9*
W3C	*6.45 (Fair/Poor)*	*58*	*28*	*14*
Wikipedia	*7.19 (Poor)*	*56*	*31*	*13*
COSMIC	*7.58 (Poor)*	*53*	*32*	*15*
CECMG	*7.71 (Poor)*	*48*	*35*	*17*
Amazon	*7.71 (Poor)*	*57*	*32*	*11*
University of Stanford	*7.75 (Poor)*	*45*	*43*	*12*
eBay	*7.78 (Poor)*	*53*	*33*	*14*
Germany Government	*7.81 (Poor)*	*41*	*42*	*17*

Tab. 2: *Web Site Evaluation of chosen Web Contents at 2024*

The reasons for such a quality loss can be the following general aspects:

- the maintenance of websites or web content is often carried out by auxiliary staff or only temporary employees (especially in higher education),
- wellknown: an essential feature of maintenance is the constant increase in the entropy of software systems,
- Finally, the evaluation model of our measurement tool may also be inappropriate due to constant technological change and expansion.

3 Current Quality Measurements vs. Usage Analysis in the Web

During the general search or observation on the web, we found the following analysis on the frequency of use of web services [Study 2024].

Web Source	Billion of clicks per month
Google	*131.2*
Youtube	*71.43*
Facebook	*12.97*
Wikipedia	*6.93*
Instagram	*6.5*
Reddit	*5.84*
Bing	*4.77*
Twitter (x.com)	*4.35*
WhatsApp	*3.83*
Yahoo	*3.46*
Amazon	*3.23*
ChatGPT	*3.1*
Tiktok	*2.61*

Tab. 3: *Ranking of Web Usage find in the (Web) Literature*

The exact values are less important here than the ranking itself. Obviously, the search for information is very dominant. The fact that the search for information can of course also just mean curiosity or the search for variety is of no further interest or search for knowledge. But, the psychological, social and demographic context is not our area of expertise and would be not considered here furthermore.

Finally, the simple application of our Measurement Suite software is shown in the following table.

Web Source	Quality	Good (in percent)	Bad (in percent)	Fair (in percent)
Twitter (x.com)	*2.8 (Good)*	*91*	*5*	*4*
Google	*3.81 (Good)*	*79*	*8*	*13*
Bing	*7.01 (Poor)*	*66*	*29*	*5*
Wikipedia	*7.19 (Poor)*	*56*	*31*	*13*
Amazon	*7.71 (Poor)*	*57*	*32*	*11*
WhatsApp	*7.8 (Poor)*	*55*	*40*	*5*
Facebook	*7.82 (Poor)*	*51*	*40*	*9*
Instagram	*7.85 (Poor)*	*55*	*42*	*3*
Reddit	*7.92 (Poor)*	*49*	*41*	*10*
Yahoo	*7.93 (Poor)*	*49*	*42*	*9*
Tiktok	*8.06 (Poor)*	*54*	*37*	*5*
ChatGPT	*8.08 (Poor)*	*54*	*37*	*9*
Youtube	*8.38 (Poor)*	*47*	*46*	*7*

Tab. 4: *Web Site Evaluation based of our Web Measurement Suite*

Here, too, the generally poor quality of most websites is evident on the basis of our classic web evaluation. The reasons for this - as already indicated above - are as follows:

- The maintenance of the websites cannot be carried out with the classic meticulousness due to the extraordinarily large volume in the shortest possible time,
- The personnel for website maintenance can only be indirectly connected to the main processes of the company (e.g. as an external maintenance service).
- As mentioned above, the measurement and evaluation system of our tool can no longer fully meet the current requirements and technologies,
- The fact that the main metric on the web is actually only the click rate can also have a very specific impact on user behavior,
- Last but not least: the measurement itself can of course lead to very specific measurement values on the basis of the used private provider.

The following figure shows a direct comparison of frequency of use vs. classic software quality.

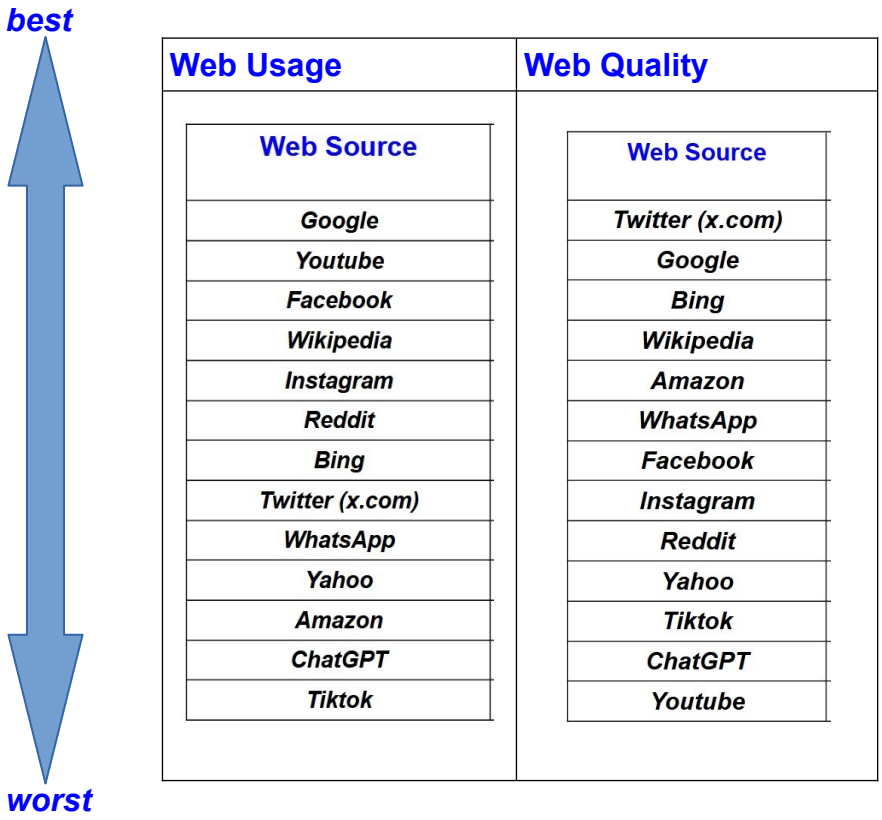


Fig. 2: *Relationship between Web Usage and Web Quality Evaluations*

4 Conclusions

The above question about the connection between (classic) software quality and frequency of use naturally implies a wide variety of interpretations.

Of course, the main criterion for software quality is user satisfaction. However, the particular interests of the various user groups, their motivation and, last but not least, the available options and conditions for using the web also play an important role.

On the one hand, web tools are used for the design of websites for a wide range of applications, which generate and optimize the web presence professionally for the assurance of a basic quality.

But on the other hand, the dominance of a click rate metric with its special features of remuneration and the associated effects of real, initiated and purchased clicks is also an important aspect of reasons for a larger web usage.

From the perspective of software quality assurance, such considerations are definitely interesting. But, what would an AI think about it?

5 References

[Abran 2010] Abran, A.: *Software Metrics and Software Metrology.* John Wiley & Sons, 2010

[Bundschuh 2008] Bundschuh, M.; Dekkers, C.: *The IT Measurement Compendium.* Springer Publ., 2008

[Dumke 1988] Dumke, R.: *Makroprogrammierung.* Fachbuchverlag, Leipzig, 1988

[Dumke 2003] Dumke, R.; Lother, M.; Wille, Z.; Zbrog, F.: *Web Engineering.* Pearson Education Publ., Munich 2003

[Dumke 2014] Dumke, R., Schmietendorf, A., Seufert, M., Wille, C.: *Handbuch der Softwareumfangsmessung und Aufwandschätzung.* Logos Verlag Berlin, 2014

[Ebert 2007] Ebert, C.; Dumke, R.: *Software Measurement – Establish, Extract, Evaluate, Execute.* Springer Publ., 2007

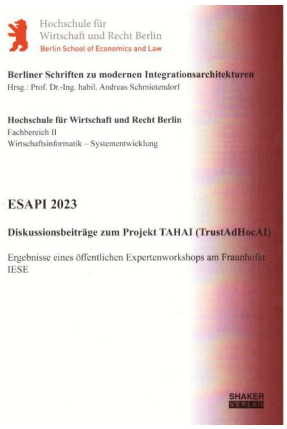
[Jones 2010] Jones, C.: *Software Engineering Best Practices – Lessons from Successful Projects in the Top Companies.* McGraw-Hill Publ., 2010

[Sneed 2017] Sneed, H.: *Endstation Wien – 45 Jahre Projekterfahrungen in der deutschsprachigen IT-Welt.* BoD Norderstedt, 2017

[Study 2024] https://de.statista.com/statistik/daten/studie/1458250/umfrage/meistbesuchte-websites-weiltweit-nach-anzahl-der-besucher/

[Weiß 2004] Weiß, M.: Konzeptiom und prototypische Implementation einer agentenbasierten Web-Qualitätssicherung, University of Magdeburg, Faculty of Informatics, 2004

[Zuse 1998] Zuse, H.: *A Framework of Software Measurement*. de Gruyter Publ., Berlin, 1998

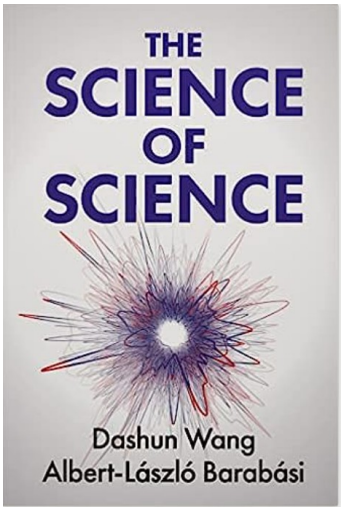


Andreas Schmietendorf, Jens Heidrich:

ESAPI 2023 – Diskussionsbeiträge zum Projekt TAHAI (TrustAdHocAI)

Shaker Verlag, Aachen, 2023

Das vorliegende Buch fasst die Beiträge und Diskussionen des 7. Workshop zur Bewertung von service-basierten APIs zusammen und ist in der Buchreihe der Schriften zu modernen Integrationsarchitekturen erschienen. Es ist im Kontext des sogenannten TAHAI-Projektes eingeordnet.

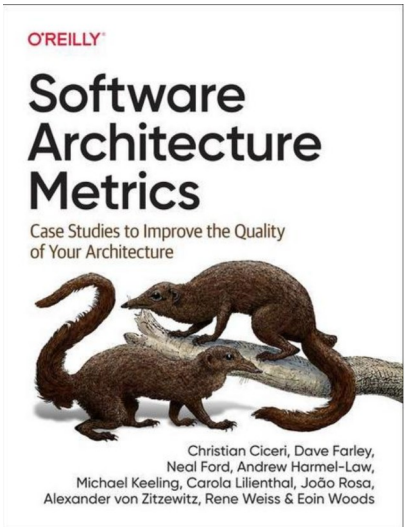


D. Wang, A.-L. Barabási:

The Science of Science Big Data, Metrics, and Impact

Cambridge University Press, 2021

„Big data analysis and quantitative tools help identify success and failure within the discipline. Areas in the 'science of science' that are ripe for further research are explored, and the implications this could have for future technological and innovative work are examined. With anecdotes and detailed, easy-to-follow explanations of the research, this book is accessible to all scientists, policy makers, and administrators with an interest in the wider scientific enterprise.“

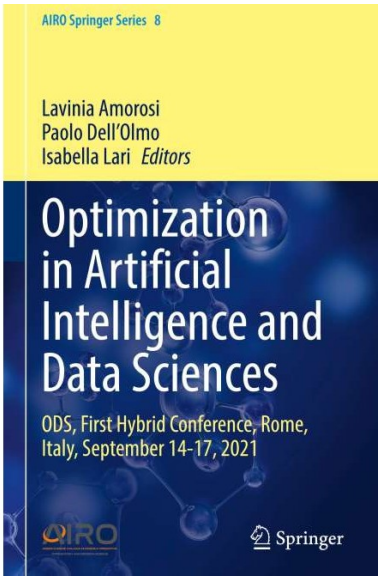


C. Cicero et al.:

Software Architecture Metrics

O'Reilly Publ, May 2022

„This isn't a book about theory. It's more about practice and implementation, about what has already been tried and worked. Detecting software architectural issues early is crucial for the success of your software: it helps mitigate the risk of poor performance and lowers the cost of repairing those issues. Written by practitioners for software architects and software developers eager to explore successful case studies, this guide will help you learn more about decision and measurement effectiveness.“

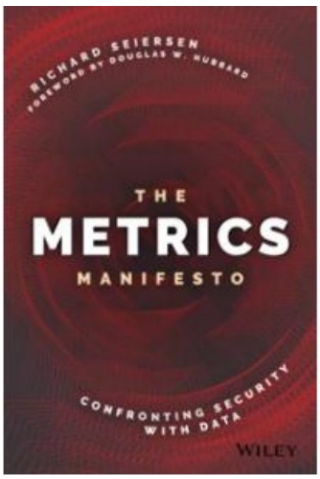


L. Amorosi, P. Dell'Olmo, I. Lari

Optimization in Artificial Intelligence and Data Science

Springer Publ. Berlin, Heidelberg, 2022

„The book offers new and original contributions on different methodological optimization topics, from Support Vector Machines to Game Theory Network Models, from Mathematical Programming to Heuristic Algorithms, and Optimization Methods for a number of emerging problems from Truck and Drone delivery to Risk Assessment, from Power Networks Design to Portfolio Optimization. The articles in the book can give a significant edge to the general themes of sustainability and pollution reduction, distributive logistics, healthcare management in pandemic scenarios and clinical trials, distributed computing, scheduling, and many others."

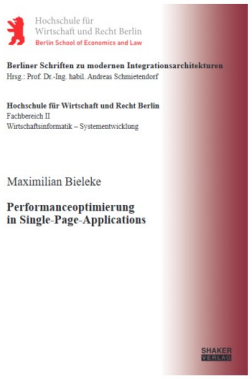


Seiersen, R.:

The Metrics Manifesto: Confronting Security with Data

John Wiley Publ., 2022, ISBN 978-1-119-51536-4

The Metrics Manifesto considers security with data delivers an examination of security metrics with R, the popular opensource programming language and software development environment for statistical computing. This insightful and upto-date guide offers readers a practical focus on applied measurement that can prove or disprove the efficacy of information security measures taken by a firm. The book's detailed chapters combine topics like security, predictive analytics, and R programming to present an authoritative and innovative approach to security metrics

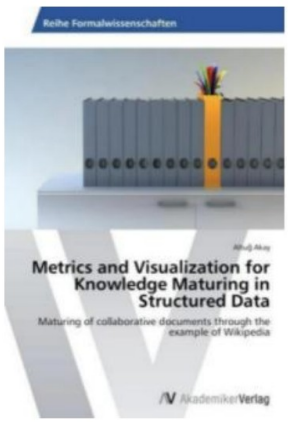


Maxemilian Bieleke:

Performanceoptimierung in Single-Page Application

Shaker-Verlag, Aachen, 2021, ISBN 978-3-8440-8315-6

Das vorliegende Buch beschreibt die Effizienz von Web-Applikationen hinsichtlich deren Performance in ausgewählten Anwendungs-bereichen.



Akay, A.:

Metrics and Visualization for Knowledge Maturing in Structured Data

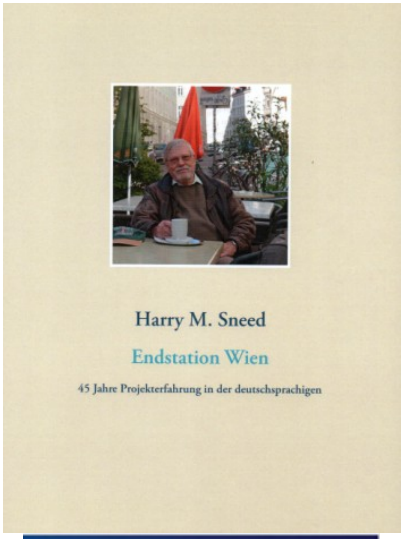
Akademiker-Verlag, 2021

This book considers the maturing of information in collaborative environments such as Wikis, intranet documents or documents in cloud is investigated via four metrics. After the definition and calculation of the metrics, the results are visualized in graphical format. Therefore, the readers can see the evolution of the metrics within the time, but also the relations of metrics with each other.

Harry Sneed:

Endstation Wien

45 Jahre Projekterfahrungen in der deutschsprachigen IT-Welt



BoD Norderstedt, 2017, 328 S.
ISBN 978-3-7448-8364-1

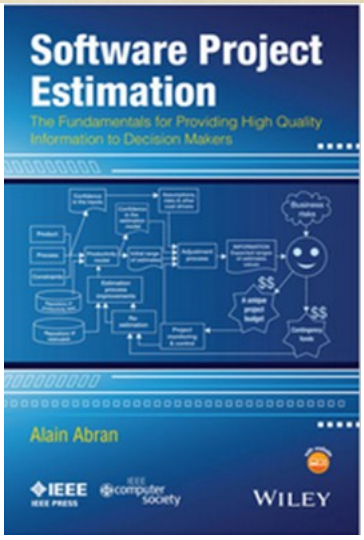
Dieses Buch beschreibt nahezu die gesamte Tätigkeit von Harry Sneed in der IT-Welt, von den Anfängen der Großrechner mit den COBOL und PL/1-Programmen bis hin zu den aktuellen und modernen Ansätzen Service-orientierter Technologien und Systemen.

Dieses Buch fasst vor allem die umfangreichen Erfahrungen zu Wartungs-, Migrations- und Testprojekten zusammen, die auch für die Beherrschung aktueller und moderner Software-Anwendungen, von unschätzbarem Wert sind.

Abran, A.:

Software Project Estimation: The Fundamentals for Providing

High Quality Information to Decision Makers



Wiley IEEE Computer Society Press, 2015 (288 pages), ISBN 978-1-118-95408-9

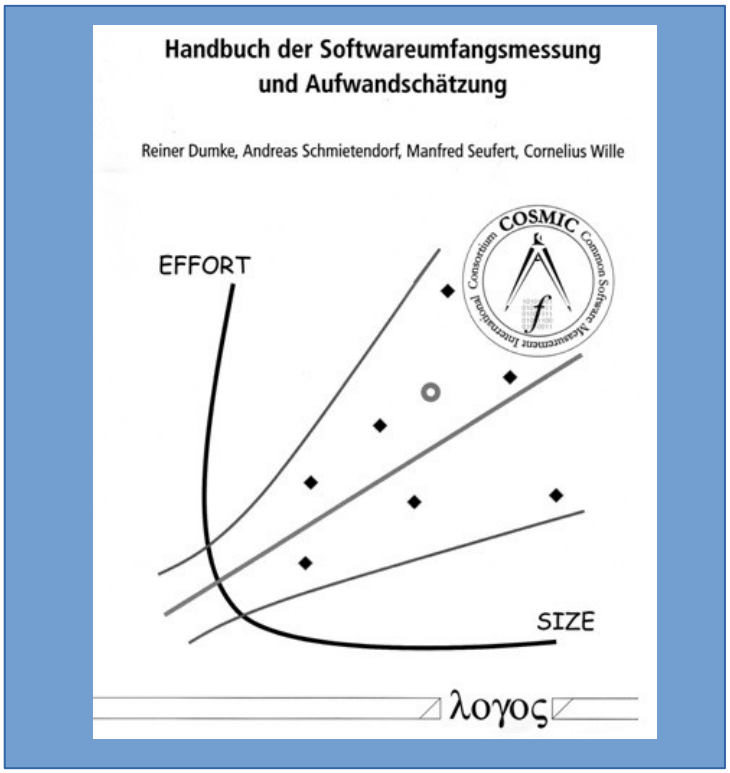
This book introduces theoretical concepts to explain the fundamentals of the design and evaluation of software estimation models. It provides software professionals with vital information on the best software management software out there. End-of-chapter exercises, Over 100 figures illustrating the concepts presented throughout the book, Examples incorporated with industry data.

Please remember:

Dumke, R., Schmietendorf, A., Seufert, M., Wille, C.:

Handbuch der Softwareumfangsmessung und Aufwandschätzung

Logos Verlag, Berlin, 2014 (570 Seiten), ISBN 978-3-8325-3784-5



This book shows an overview about the current software size measurement and estimation approaches and methods. The essential part in this book gives a complete description of the ***COSMIC measurement method****, their application for different systems like embedded and business software and their use for cost and effort estimation based on this modern ISO size measurement standard.*

Software Measurement & Data Analysis Addressed Conferences

September 2024

EuroAsiaSPI² 2024:
European Systems & Software Process Improvement and Innovation Conference
September 4 - 6, 2024, Munich, Germany
see: https://conference.eurospi.net/index.php/en/

QEST 2024:
International Conference on Quantitative Evaluation of Systems
September 9 - 13, 2024, Calgary, Canada
see: https://www.qest.org

data2day 2024:
Konferenz für Data Scientists, Data Engineers und Data Teans
September 18 - 19, 2024, Heidelberg, Germany
see: https://www.data2day.de/

ISSTA 2024:
International Symposium on Software Testing and Analysis
September 16 - 20, 2024, Vienna, Austria
see: https://2024.issta.org

Data Science Workshop 2024:
Young Scientists an early-stage research in Data Science Workshop 2024
September 26, 2024, Wiesbaden, Germany
see: https://informatik2024.gi.de/

KI4SE 2024:
KI for Software Engineering
September 28, 2024, Wiesbaden, Germany
see: https://informatik2024.gi.de/

October 2024

ICSEA 2024:
International Conference on Software Engineering Advances
September 29 – October 3, 2024, Venice, Italy
see:
https://www.iaria.org/conferences2024/ICSEA24.html

IWSM/MENSURA 2024:
The Join Conference of the 33nd International Workshop on Software Measurement and the 18th International Confernce on Software Process and Product Measurement
September 30 – October 4, 2024, Montreal, Canada
see: https://www.iwsm-mensura.org/

ICSME 2024:	**International Conference on Software Maintenance and Evolution** October 6 – 11, 2024, Flagstaff, AZ, USA **see: https://conf.researchr.org/home/icsme-2024**
API 2024:	**API Conference 2024** October, 2024, Berlin, Germany **see: https://apiconference.net/berlin-de/**
ESEIW 2024:	**Empirical Software Engineering International Week** October 20 - 25, 2024, Barcelona, Spain **see: https://conf.researchr.org/home/eseiw-2024**
ESEM 2024:	**Conference on Empirical Software Engineering and Measurement** October 20 - 25, 2024, Barcelona, Spain **see: https://conf.researchr.org/home/esem-2024**
ASQT 2024:	**Arbeitskonferenz Softwarequalität, Test und Innovation** *--- currently in planning ---* **see: http://www.asqt.org/**
ODSC West 2024:	**Open Data Science Conference East** October 29 - 31, 2024, Burlingame, CA, USA **see: https://www.odsc.com**
ASE 2024:	**Automated Software Engineering** October 27 – November 1, 2024, Sacramento, USA **see: https://conf.researchr.org/home/ase-2024**

November 2024

SEFM 2024:	**International Conference on Software Engineering and Formal Methods** *--- in the planning phase ---* **see: https://sefm-conference.github.io/**

December 2024

PROFES 2024:	**International Conference on Product Focused Software Process Improvement** Dezember 2 - 4, 2024, Tartu, Estonia **see: https://conf.researchr.com/home//profes-2024/**
IEEE ICDM 2024:	**IEEE International Conference on Data Mining** Dezember 9- 12 , 2024, Abu Dhabi, UAE **see: https://18.133.154.107/**
BCD 2024:	**International Conference on Big Data, Cloud Computing, and Data Science Engineering** December 12 - 14, 2024, Danang, Vietnam

see: https://acisinternational.org/conferences/bcd-2024-winter/

IEEE International Conference on Big Data

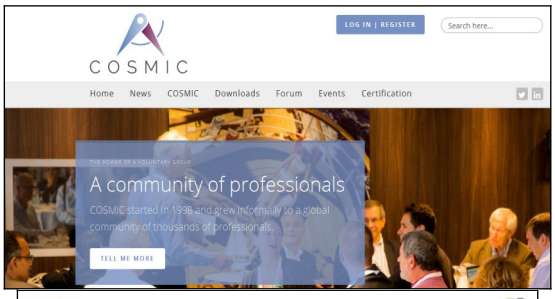
Big Data 2024: December 15-18, 2024. Washington, USA

see: https://bigdataieee.org/BigData2024/

see also:

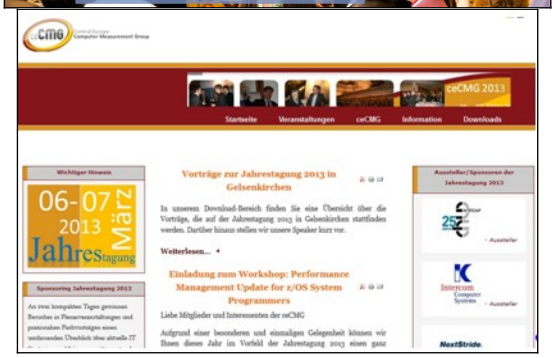
- https://www.acm.org/conferences
- https://www.ieee.org/conferences_events/index.html

COMMUNITIES



Common Software Measurement International Consortium (COSMIC)

http://cosmic-sizing.org



Central Europe Computer Measurement Group (ceCMG)

http://www.cecmg.de

Metrics Association's International Network (MAIN)

http://*www.mai-net.org*

Netherlands Software Metrics users Association (NESMA)

http://www.nesma.org/

GI-Fachgruppe Measurements & Data Science

https://fg-data-science.gi.de

(Measurement News Online)

International Software Benchmarking Standard Group (ISBSG)

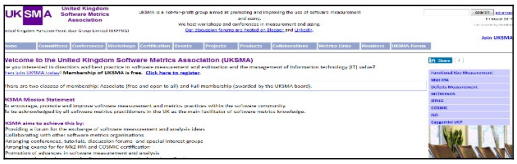
https://www.isbsg.org

Finnish Software Measurement Association (FISMA)

http://www.fisma.fi/in-english/

Asociacion Espanola de Metricas de Software

http://www.aemes.org/



United Kingdom Software Metrics Association (UKSMA)

http://www.uksma.co.uk

Gruppo Utenti Function Point Italia - Italian Software Metrics Association (GUFPI - ISMA)

http://www.gufpi-isma.org

Anwenderkonferenz Software-qualität und Test (ASQT)

http://www.asqt.org

MEASUREMENT SERVICES

Software Measurement Laboratory (SML@b)

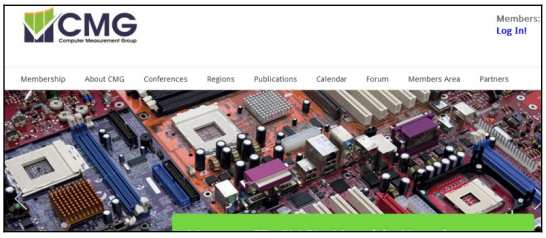
http://www.smlab.de

International Function Point Users Group (IFPUG)

http:www.ifpug.org

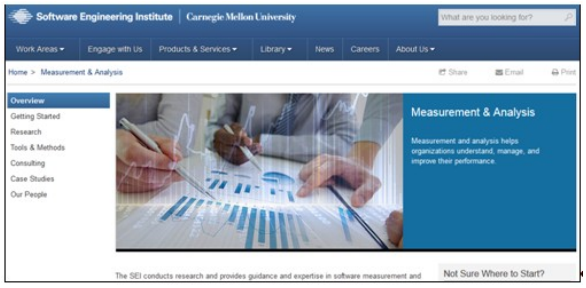
Practical Software & Systems Measurement

www.psmsc.com/:



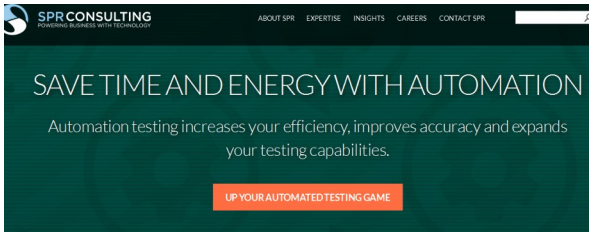
Computer Measurement Group (CMG)

http://www.cmg.org



Software Engineering Institute (SEI)

www.sei.cmu.edu/measurement/



Software Productivity Research (SPR)

http://www.spr.com/

McCabe & Associates

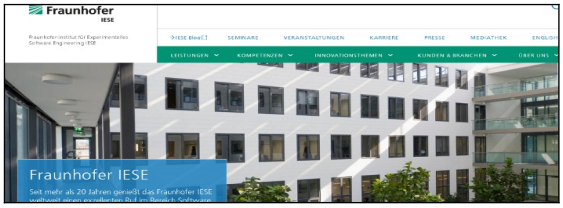
http://www.mccabe.com

SQS Gesellschaft für Software-Qualitätssicherung

http://www.sqs.de

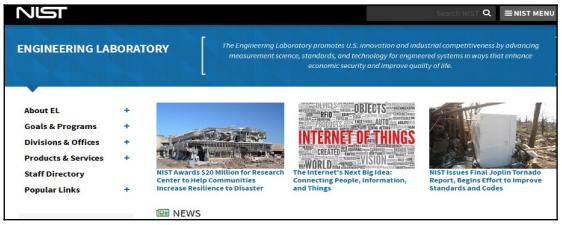
Quantitative Software Management (QSM)

http://www.qsm.com/



Fraunhofer Institute for Experimental Software Engineering (IESE)

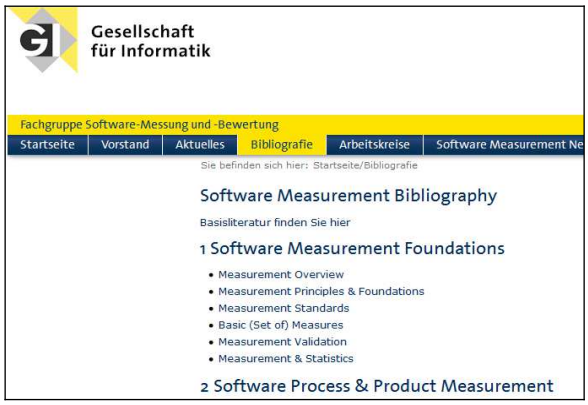
https://www.iese.fraunhofer.de/



National Institute of Standards and Technology (NIST)

https://www.nist.gov/el

SOFTWARE MEASUREMENT INFORMATION



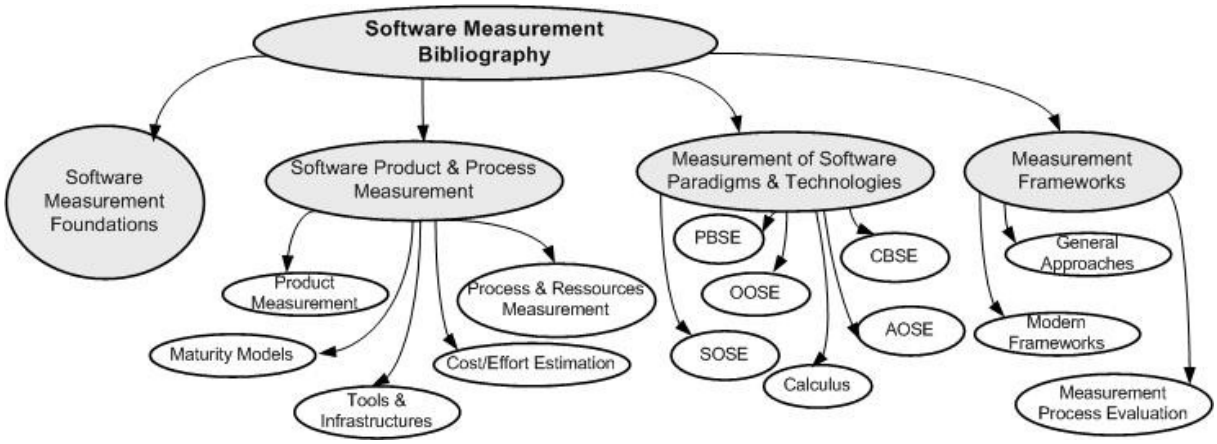
Software Measurement Bibliography

See our overview about software metrics and measurement in the Bibliography at

https://fg-metriken.gi.de/bibliographie/

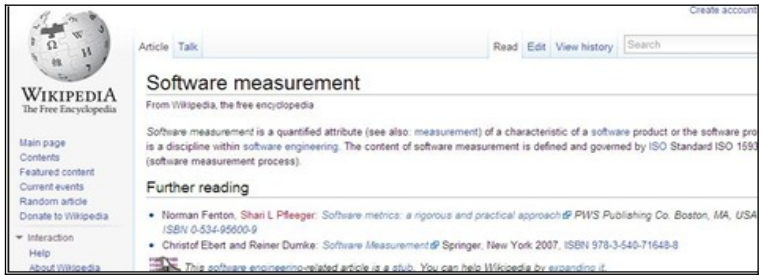
including any hundreds of books and papers

Bibliography Structure:



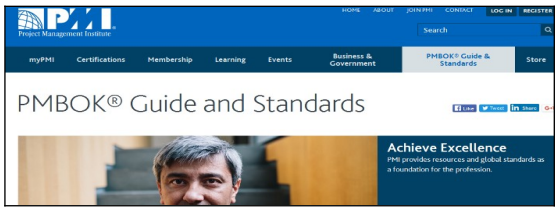
Software Measurement & Wikipedia

Help to qualify the software measurement knowledge and intentions in the world wide web:



Software Engineering Body of Knowledge (SWEBOK)

http://*www.swebok.org*



Project Management Body of Knowledge (PMBOK)

http://www.pmbook.org

SOFTWARE MEASUREMENT NEWS

VOLUME 29 2024 NUMBER 2

CONTENTS

Announcements **2**

Andreas Schmietendorf: Workshop: Herausforderungen Low-Code orientierter KI-Ansätze *2*
Reiner R. Dumke: SML@b News *4*

Conference Reports **5**

Alain Abran: IWSM/Mensura 2024 ..... *5*

Community Reports **17**

Jean-Marc Desharnais: COSMIC News 2024 ..... *17*
NESMA News 2024 ..... *18*
Luigi Buglione: GUFPI News 2024 ..... *19*
.....

News Papers **20**

Sandro Hartenstein, Marc-Steven Fischer, Andreas Schmietendorf: Fallstudie zur KI-gestützten Anonymisierung deutschsprachiger Transkripte *20*
Reiner R. Dumke, Michael Weiß: Could Software Quality influence Web Usage? *33*

New Books on Software Measurement **41**

Conferences Addressing Measurement Issues **45**

Metrics in the World-Wide Web **48**

ISSN 1867-9196