

シラバスから見る日本と海外における ソフトウェア工学教育の違い：世界のトップ大学との比較

柏 祐太郎^{1,a)} 近藤 将成^{2,b)} 槇原 絵里奈^{3,c)} 亀井 靖高^{2,d)}

概要：本研究では、日本国内における主要大学とソフトウェア工学における海外トップに位置する大学のソフトウェア工学教育との違いを明らかにする。具体的には、国内大学のシラバスシステムでソフトウェア工学に関する科目を抽出・目視し、どのような技術が教育されているかを調査する。そして、それらの授業がソフトウェア工学知識体系で定義される各知識領域をどの程度網羅しているかを確認する。最後に、海外トップ大学におけるシラバスも同様に調査し、日本のソフトウェア工学教育との比較を通して違いを観察する。国内大学のシラバスを調査した結果、ソフトウェア工学知識体系の知識領域のうち中央値で75.0%が網羅されていることが確認された。しかし、「アーキテクチャ」「運用」「構成管理」などの分野は多くの大学で十分に扱われていないことが判明した。一方、海外のトップ大学では網羅率が中央値84.6%と高く、授業数も日本の約3倍にのぼるなど、より専門的かつ実践的な教育が行われている。ただし、海外トップ大学でも「運用」分野に関しては講義実施率が低く、共通の課題であることが示唆された。

Differences in Software Engineering Education: A Syllabus-Based Comparison Between Japan and Leading Global Universities

1. はじめに

現代社会において、デジタル技術の進展は目覚ましく、我々の生活のあらゆる側面に影響を及ぼす一方、我が国におけるデジタル化の遅れが指摘されている [1]。特に、デジタル赤字と呼ばれる海外のデジタルサービス関連の支出が収入を上回り、国際収支が赤字となる現象が顕著である。デジタル赤字は、経済成長や国際競争力において重大な課題とされ、2023 年にはデジタル関連収支が約 5.5 兆円の赤字を記録した [2]。これは、デジタルサービスの海外依存度が高まっていることを反映している [3]。

デジタル赤字の原因として、日本のソフトウェア開発技術が他国に比べて低いことが考えられる。開発技術の差を埋めるために急速に新しい技術・開発手法の導入が進められているが、未だに世界基準より遅れていると指摘される [4], [5]。例えば、情報処理推進機構が行った DX 動向

調査では、日本における DX の取組みが 2018 年の DX レポート以降進んでいることを指摘した。一方、DX の成果が出ていると回答した割合は日本の 64.3%に対して米国は 89.0%であり、日本は DX の成果を十分に享受できていないと言える [5]。最先端のソフトウェア開発技術を浸透させるには、ソフトウェア工学教育の充実が不可欠である。特に世界では、情報系の学部カリキュラムにおいて、最新の技術や開発手法を取り入れた教育の必要性が謳われている [6]。しかし、国内のソフトウェア工学教育がどのように実施されているかを系統的かつ網羅的に調べた報告はない。

そこで本研究では、日本で実施されているソフトウェア工学教育の全貌の把握と、世界のトップ大学とのソフトウェア工学教育における違いを明らかにする。具体的には、国内における情報系学部を持つ大学のシラバスからソフトウェア工学に関するシラバスを抽出・目視し、どのような技術が教育されているかを調査する。そして、ソフトウェア工学知識体系 [7] の知識領域をどの程度網羅しているかを確認する。最後に、海外トップ大学におけるシラバスも同様に調査し、国内のソフトウェア開発現場を向上させるために、海外トップ大学のソフトウェア教育と異なる点を明らかにし、見習うことができる要素を考察する。

¹ 奈良先端科学技術大学院大学

² 九州大学

³ 立命館大学

^{a)} yutaro.kashiwa@is.naist.jp

^{b)} kondo@ait.kyushu-u.ac.jp

^{c)} makihara@fc.ritsumei.ac.jp

^{d)} kamei@ait.kyushu-u.ac.jp

2. ソフトウェア工学知識体系 (SWEBOK)

ソフトウェア工学知識体系 (SWEBOK: Software Engineering Body of Knowledge) とは、ソフトウェア開発の実務者や研究者が共通して理解すべき知識を体系的にまとめたものであり、IEEE Computer Society によって策定された [7]。計 18 の知識領域が整理され、それぞれの知識領域でソフトウェア工学の実践に必要な基本的な概念や技術を網羅している。また、SWEBOK は定期的に更新されており、2024 年に最新版である ver.4 が出版された [7]。SWEBOK ver.4 には、新たに「Agile」や「DevOps」など、様々な実用的かつ最先端技術が追記されている。本章では、SWEBOK で定義されている 18 の知識領域のうち、後述する本研究の目的に沿った 12 の知識領域について概略を説明する。対象外とした知識領域と理由は 4.3 節で詳述する。

ソフトウェア要求: ソフトウェア要求とはユーザが問題を解決したり特定の目標に達するためにソフトウェアに求められる条件や能力などを指す言葉である。開発者はソフトウェア要求で決められた内容をもとに実装やソフトウェアテストに関する意思決定を行う。ソフトウェア要求を明確化することで、ソフトウェアの正しい動作を定義でき、それ以外の動作を誤った動作として定義できるようになるなど、ソフトウェア開発における重要な基本要素である。

ソフトウェアアーキテクチャ: ソフトウェアアーキテクチャはシステムを構築する要素や要素間の関係性などが設計の原則に従い体系化されたものを指す。開発者はシステムの構成要素や構成要素間の関係性、設計のルール策定など、システムの既存構造を設計し、それに必要なルールやセキュリティなど品質要件を明らかにする。アーキテクチャの作成から評価を通し、システムの持続可能性や保守性などを確保する。

ソフトウェア設計: システム要求を満たすために、ソフトウェアの構成要素とそれらの関係性、ルールなどを定義、詳細化する。開発者はシステム要求を基に設計仕様を考え、システムの構造や振る舞いを決定する。モジュール化、データフロー、使用するデータ構造やアルゴリズム、デザインパターンなどを設計することで、システムの保守性や信頼性を確保する。

ソフトウェア構築: システムの設計を正確にコードへ落とし込み、ユニットテストやデバッグを通じて高品質なソフトウェアを構築する。開発者はコーディングからデバッグ、テスト、評価を一通り行う。また、コードの最適化や効率化なども行い、ソフトウェアのパフォーマンス向上も行い、ソフトウェアの保守性や再利用性を確保する。

ソフトウェアテスト: ソフトウェアテストでは、有限のテストケースセットをテスト対象システム (SUT: System

Under Test) に与えた際に、期待通りの動作であるかを検証することを指す。近年では、開発者がテストコードを記載し、SUT を動作させ、動作が期待通りかを検証する。

ソフトウェア運用: ソフトウェア運用は、アプリケーションやシステムの整合性と安定性を維持しながら、それらを運用することを指す。これらの活動には、本番の運用環境へソフトウェアをデプロイおよび構成することに加え、使用中のアプリケーションの監視と管理が含まれる。アプリケーションが運用開始されると、運用者は発見された欠陥、システムソフトウェア環境やハードウェア機器に対する変更、および新たに発生するユーザー要件を管理する必要がある。

ソフトウェア保守: ソフトウェア保守とは、ソフトウェアが運用段階において新たに発見されるバグ、環境の変化、および、新たな要求といった要素に適切に対応するための活動である。開発者はリリース後の計画を事前に策定するなどしてソフトウェアを保守する。

ソフトウェア構成管理: ソフトウェア構成管理とは、ソフトウェアの構成要素の完全性と正確性を確保するために、ソフトウェアライフサイクル全体にわたって構成管理をするプロセスを指す。開発者は、製品の要素を特定し、変更を管理・制御し、構成情報を検証・記録・報告することで、製品の進化と整合性を制御する必要がある。

ソフトウェアエンジニアリングマネジメント: ソフトウェアエンジニアリングマネジメントは、ソフトウェアプロジェクトにおける計画、見積もり、測定といった作業全体を指す言葉である。ソフトウェアプロダクトやサービスが適切かつ効果的にステークホルダーに提供され、ステークホルダーの利益になることを保証するためのものである。

ソフトウェアエンジニアリングプロセス: ソフトウェアエンジニアリングプロセスとは開発者がソフトウェアを構築・運用するために実施する作業活動である。開発者は設計から運用後のメンテナンスまでの様々な工程において様々な活動 (例: 設計における要件定義) を行っている。

ソフトウェアエンジニアリングモデルと方法: ソフトウェアエンジニアリングにおいてモデルとは、モデル構築および分析のための表記方法および手順を与えることで、問題解決のためのアプローチを提供する。方法はソフトウェアや関連する製品を設計、テスト、検証等をするためのアプローチを提供する。モデルと方法は、ソフトウェアエンジニアリングに構造を与えることでソフトウェアエンジニアリングを体系化および反復可能にする。

ソフトウェア品質: ソフトウェア品質は満たすべき要求を満たしていること、それをどの程度満たしているかで定義される。開発者はソフトウェアの品質を考慮することで、ステークホルダーの期待をソフトウェアが満たしているかを評価できる。

3. Research Questions

以下では、本研究で取り組む2つの Research Questions (RQs) を紹介する。

RQ1: 日本のソフトウェア工学教育はソフトウェア工学知識体系をどの程度網羅しているのか?

1960年代にソフトウェア危機 (Software Crisis)[8] が警鐘されてから約60年、ソフトウェア開発技術は日進月歩で進化してきた。ソフトウェア工学知識体系 (SWEBOK) は、先人が蓄積してきたこれらのソフトウェア工学の知識と技術を体系的に整理したものであり、国際的に認められた知識体系である。日本のソフトウェア工学教育が、ソフトウェア工学知識体系のどの程度を網羅しているかを評価することは、教育と実務におけるギャップを明らかにでき、教育の質を向上させるための具体的な改善点の発見に繋がる。RQ1では国内の情報系教育を行う大学を抽出し、これらの大学で教育するソフトウェア工学がSWEBOKにおける知識領域をどの程度を網羅しているかを調査する。

RQ2: 日本のソフトウェア工学教育は世界のトップ大学におけるソフトウェア工学教育と何が異なるのか?

産業界が求めるソフトウェア技術者の育成のためにも、世界の大学では様々な現代的教育が行われている。例えば、海外における教育の実例として、DevOpsを取り入れたもの[9]や、PBL (Project-Based Learning) にマイクロサービスを取り入れるための授業設計[10]が報告されている。このように海外では最新技術を取り入れた教育カリキュラムが構築されており、ICT先進国を支える人材を輩出し続けている。RQ2では、日本と海外トップ大学のカリキュラムを比較し、日本におけるソフトウェア工学教育が海外のトップ大学のそれとどのように異なるのかを観察する。

4. 調査方法

本節では、本研究を実施するために必要となるシラバスの特定方法および、データ分析方法を説明する。図1に本研究の全体像を示す。

4.1 情報系学部・学科を有する大学の選定

本研究では、まず情報教育を行う学部 (例: 情報工学部) を特定する。プログラミング教育は学部を問わず行われている場合が多いが、本論文はソフトウェア工学に特化した講義・演習を対象とするため、まずは情報教育を扱う学部および大学の特定を行った。

本研究では、情報系の人材確保に向けた文部科学省の取り組みである「大学・高専機能強化支援事業」に注目した。当該取り組みでは、支援1 (学部再編等による特定成長分

野への転換) と支援2 (高度情報専門人材の確保に向けた機能強化) が存在するが、支援1は、第四次産業革命や Society 5.0、データサイエンスブームなどを受け、情報系の人材確保に向けた新学部学科設立を目的としている。これらの大学は2024年末時点で創設準備段階であることが多かったため、分析対象としては不適切と考えた。一方、支援2は既存の情報学部の定員を増員することを目的としている。つまり、支援2に申し込んだ教育機関はすでに情報系学部・学科を有すると考えられるため、支援2の初回公募*1、第2公募*2に応募後採択された大学を対象に調査を行った。ここで、本研究が学部生を対象とする点と、後述するシラバス調査より、以下に当てはまる大学は調査の対象外とした。

- 大学院大学
- 工業高等専門学校
- シラバス検索システムが不在もしくは認証が必要

以上より、合計69校 (国立54校、公立7校、私立8校) を調査対象とする。

4.2 シラバス検索

次に、各大学におけるソフトウェア工学に関する授業のシラバスを検索する。各大学のシラバス検索システムより、以下の条件で授業のシラバスを検索した。

- 学部生を対象にした科目
- 情報系学部を対象とした科目
- 2024年度開講科目
- 講義名に「ソフトウェア」または「ソフトウエア」を含む

講義名から検索できない場合、キーワード検索や全検索などを利用し、検索に該当したシラバスの講義名を一つずつ目視で確認した。該当の授業が本当にソフトウェア工学の各要素を扱っているのかについては、著者のうち1名が調査対象であると判断したのちに、異なる2名の著者で同様の授業シラバスを確認し、調査対象として適切であるかを確認した。なお、タイトルにはソフトウェア工学が含まれるものの、実際の授業内容の過半数がSWEBOKで扱うソフトウェア工学に当てはまらない場合 (例: プログラミングの基礎やハードウェアなど)、調査の対象外とした。

また、RQ2では、海外大学のシラバスを調査するが、シラバスの特定はRQ1と同様の方法で実施する。ただし、すべての学部を調査することは時間的制約により難しいため、CSランキングにおける「Software Engineering」部門におけるトップ100から対象の大学を選定する。*3 なお、

*1 https://www.mext.go.jp/content/20230721-mxt_senmon01-72.pdf

*2 https://www.mext.go.jp/content/20240624-mxt_senmon01-000027827.pdf

*3 <https://csrankings.org/#/fromyear/2014/toyear/2024/index?soft&world>

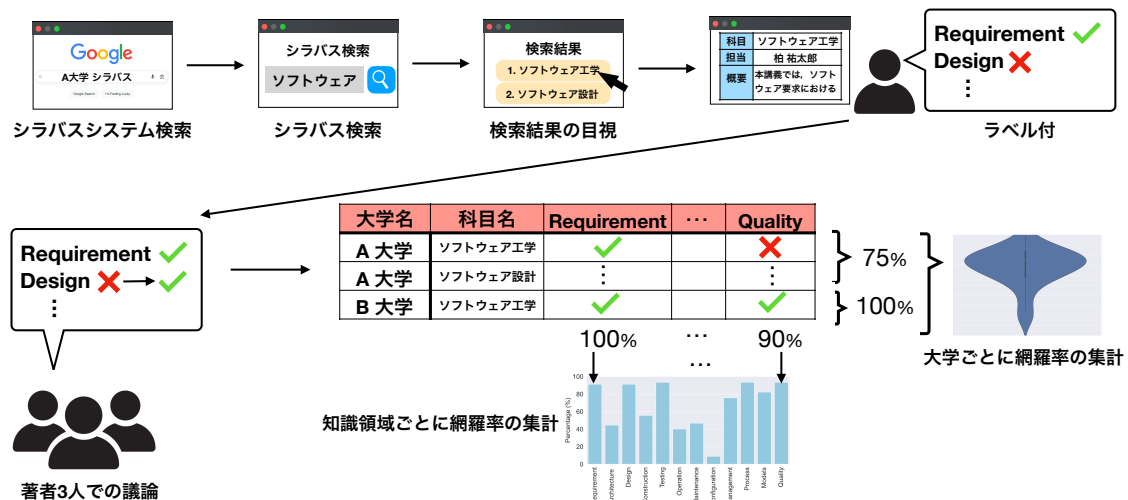


図 1: 本研究の概要

海外トップ大学のシラバスにおいて日本語で検索はできないため、日本語に対応するキーワードとして“Software”を用いた。また海外では 10 月から新学期がスタートする。そのため、最も新しい 2023 年 10 月から 2024 年 9 月までの一年間を検索対象とした。この期間のシラバスが既に閲覧できない場合、最も最新のシラバスから過去 1 年分を対象期間とした。その他の条件は日本と同様のものを利用した。調査の結果、CS ランキングトップ 100 の大学中、52 大学においてシラバスシステムを特定できた。そのうち、内部生のみ検索可能である場合 (5 大学) や Software を科目名に含む授業が該当しなかった大学 (2 大学)、授業の詳細が書かれていないもしくは英語以外の言語のページしか存在しない大学 (26 大学) などを除外した結果、19 大学が対象となった。本調査では、上位 10 大学のシラバスを調査した。対象となった国内大学・海外トップ大学は Online Appendix を参照されたい。^{*4}

4.3 SWEBOK における調査項目

次に、調査対象のシラバスが 2 章で述べた SWEBOK の各要素を満たすかについて目視調査を行った。シラバスにおける授業概要やねらい、各単元のテーマや詳細で出現したキーワードを、どの SWEBOK 知識領域に当てはまるか SWEBOK 本文を参照しながら著者らで目視分類した。例えば、A 大学のソフトウェア工学のシラバスに「デザインパターン」「オブジェクト指向設計」というキーワードが出てきた場合、SWEBOK の知識領域「ソフトウェア設計」にこれらのキーワードが分類される。つまり、A 大学のソフトウェア工学は知識領域「ソフトウェア設計」を含むと判断する。ただし、キーワードによっては複数の知識領域に跨ることがある。例えば、キーワード「ソフトウェアテ

ストと品質保証」の場合、知識領域「ソフトウェアテスト」と「ソフトウェア品質」が該当する。このような場合は、複数の知識領域を含んだ講義と判定する。

また、SWEBOK 13 章以降ではセキュリティやオペレーティングシステムなども扱われている。これらはコンピュータを扱う上で基礎的かつ重要な知識である一方、本研究で取り扱うソフトウェア開発における最新技術や開発手法に関わる教育調査として適していないと判断し、以下に示す 13 章以降の項目は調査の対象外とした。

- Sec.13: Software Security
- Sec.14: Software Engineering Professional Practice
- Sec.15: Software Engineering Economics
- Sec.16: Computing Foundations
- Sec.17: Mathematical Foundations
- Sec.18: Engineering Foundation

4.4 データ分析方法

RQ1: ソフトウェア開発者が学習すべき SWEBOK の知識領域を、各大学におけるソフトウェアに関連する講義がどの程度網羅しているかの割合 (網羅率) を計測する。学部別でなく、大学別に網羅率を測る理由は、どの学部がどの授業を受講できるか把握することが容易でないこと、また、ソフトウェア工学に関する知識領域をどの程度の網羅しているかを大学単位で計測したかったためである。

RQ2: RQ1 と同様に、海外トップ大学におけるソフトウェア工学授業の知識領域の網羅率を算出する。国内大学と海外トップ大学で網羅率の差に統計的有意差が存在するかを確認するために、ノンパラメトリックな検定であるマンホイットニーの U 検定を用いる (有意水準: 0.01)。また、その差が実質的に意味のある差であるかを確認するため、ノンパラメトリック効果量指標である Cliff's delta を用いて効果量を測定する。

^{*4} <https://github.com/pos1/2025-SES-SE-Education-Online-Appendix>

表 1: 各知識領域におけるキーワード (Top 5)

ソフトウェア要求	アーキテクチャ	ソフトウェア設計	ソフトウェア構築	ソフトウェアテスト	ソフトウェア運用
要求分析	アーキテクチャ設計	オブジェクト指向	プログラミング	ソフトウェアテスト	継続的インテグレーション
要求獲得	アーキテクチャ	UML	統合開発環境	システムテスト	デプロイ
要求定義	アーキテクチャパターン	構造化手法	再利用	テストケース設計	運用
要求仕様	機能配置	設計	コーディング	ホワイトボックステスト	リスク管理
ユースケース	機能分割	構造化分析	コーディング規約	ブラックボックステスト	仮想化
ソフトウェア保守	ソフトウェア構成管理	マネジメント	プロセス	モデルと方法	ソフトウェア品質
保守	構成管理	プロジェクト管理	ソフトウェアプロセス	モデリング	ソフトウェア品質
デバッグ	再利用	ソフトウェア見積もり	アジャイル	アジャイル	品質管理
デバッグ	バージョン管理	開発管理	ウォーターフォールモデル	形式手法	品質特性
リファクタリング	ビルドツール	ガントチャート	プロジェクト管理	モデル検査	バグ
保守と進化	ソースコード管理	WBS	ソフトウェア開発手法	スパイラルモデル	品質保証

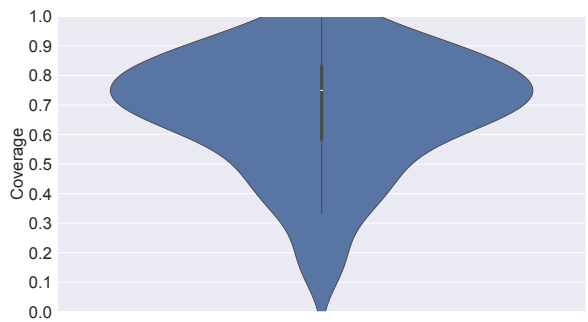


図 2: 国内大学における知識領域の網羅率

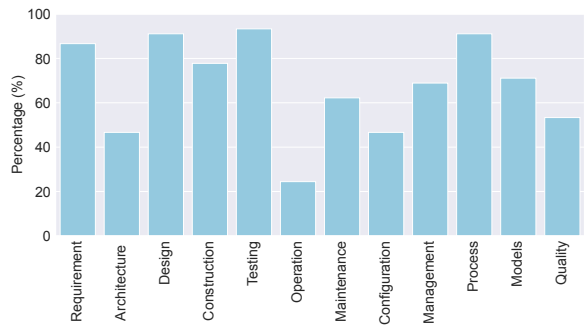


図 3: 国内大学における各知識領域の網羅率

5. 結果

本章では、各 RQ で得られた結果をまとめる。

5.1 RQ1: 国内大学のソフトウェア工学教育における SWEBOK 知識領域の網羅率

国内 69 大学におけるソフトウェア工学に関連する授業を調査した結果を以下に示す。

Findings 1. 69 大学中、45 大学 (65.2%) にソフトウェア工学に関する授業が存在した。

著者の目視調査より、65.2%の大学でソフトウェア工学に関する授業を発見した。表 1 に大学を横断して頻出するキーワードを知識領域ごとに纏める。なお、表中のキーワードはシラバスに対する著者らの目視によって発見されたキーワードを名寄せしたものであることに注意されたい。以下に目視調査によって明らかになった結果を示す。

Findings 2. 国内大学における SWEBOK 知識領域の網羅率は中央値で 75.0%であった。

図 2 に、各大学の SWEBOK の知識領域の網羅率をバイオリンプロットで示す。当該バイオリンプロットでは、外側の層 (つまり色のついた部分) の厚みが、プロットされた値の確率密度を表す。各バイオリンプロットの中央には、暗いボックスが四分位範囲 (IQR) を示し、白い点が中央値を示している。

網羅率の分布を確認すると、中央値が 75.0%であり、半数の大学が大部分を網羅していることがわかる。また、四分位範囲を確認すると、第一四分位値が 58.3%, 第三四分位値が 83.3%, その差 (IQR) は 25.0%程度と比較的小さいことがわかった。一方、網羅する知識領域が 50%未満の大学も 7 大学 (調査大学全体の 10.1%) 存在する。つまり、関連科目がない 24 大学を含めると、44.9%の大学で、ソフトウェア工学の知識領域の大半を網羅する授業がないことがわかる。

Findings 3. 多数の大学で「アーキテクチャ」、「運用」、「構成管理」の講義が不足している可能性がある。

図 3 は各知識領域を教えている大学の割合を示したものである。「要求」、「設計」、「テスト」、「プロセス」は 80%を超える大学で教えられている。一方で、「アーキテクチャ」、「運用」、「構成管理」は他の知識領域と比較して際立って低い。これらの知識領域をカバーしている大学は 50%を下回っており、半数以上の大学で教えられていない。

Answer to RQ1

国内大学におけるソフトウェア工学教育は、中央値で SWEBOK 知識領域の 75.0%と広く網羅されている。一方で、「アーキテクチャ」「運用」「構成管理」など一部の知識領域は、半数以上の大学で十分に教えられておらず、教育内容に偏りが見られる。

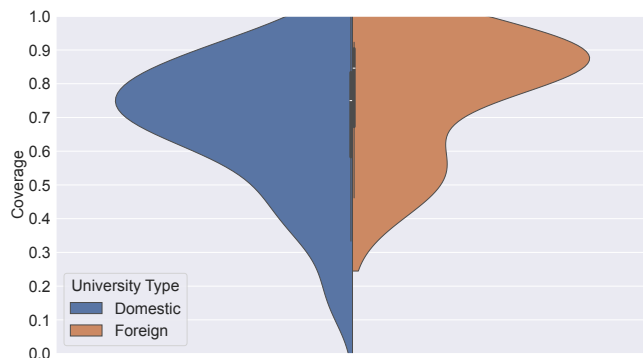


図 4: 海外トップ大学における知識領域の網羅率

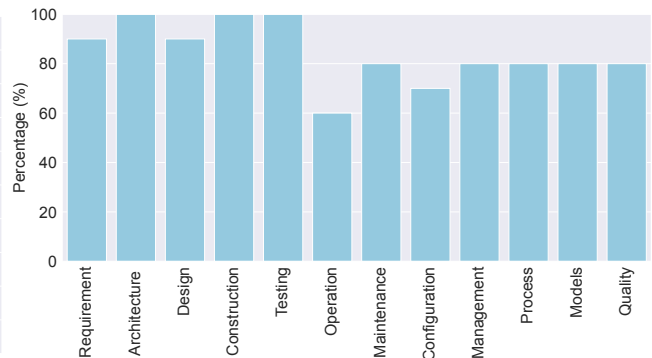


図 5: 各海外トップ大学における各知識領域の網羅率

5.2 RQ2: 海外トップ大学のソフトウェア工学教育における SWEBOK 知識領域の網羅率

調査対象の海外トップ大学におけるシラバスシステムでソフトウェア工学に関連する授業を検索した結果、すべての大学で最低 1 つ以上の授業が存在した。以下に目視調査によって明らかになった結果を示す。

Findings 4. 対象とした海外トップ大学における *SWEBOK* 知識領域の網羅率は中央値で 84.6% で、国内大学の網羅率より 9.6 ポイント高い。

図 4 に、海外トップ大学における *SWEBOK* の知識領域の網羅率の分布を示す。海外トップ大学あたりの網羅率の中央値は 84.6% で、国内大学の 75.0% と比較して 9.6 ポイント高いことがわかる。次に四分位値を確認すると、第 3 四分位値は 90.4% であり、上位 25% の大学では 9 割を超える知識領域を教えていることが示される。四分位範囲を確認すると、23.1% であり（第 1 四分位値は 67.3%）、国内大学の IQR と同様である。ただし、最小値は 46.2% であり、ほぼ全ての大学でほぼ半数以上の知識領域を網羅していた。また、網羅率について統計的検定を実施した結果、国内大学と海外トップ大学の網羅率において統計的有意差を確認できた。また、効果量も “Large” と判定された。

Findings 5. 海外トップ大学でも、「運用」に関する講義を行う大学が比較的少ない。

図 5 に、各知識領域の講義を行う大学の割合を示す。図より、多くの知識領域に関する講義は 80% 以上の大学で実施されており、特にソフトウェア構築やソフトウェアテスト、ソフトウェア品質に関する授業は、ほぼ全ての大学で行われていた。一方、「運用」に関する網羅率は 60%、「構成管理」に関する網羅率は 70% と比較的少ないことがわかる。特に、「運用」に関しては、国内だけでなく、海外にも共通する課題であることが伺える。

Findings 6. 海外トップ大学において 1 大学あたりソフトウェア工学に関わる授業数は日本の 3 倍である。

図 6 に、1 大学あたりで開講されているソフトウェア工学に関わる授業数を示す。日本では 1 校あたり 2 科目が開

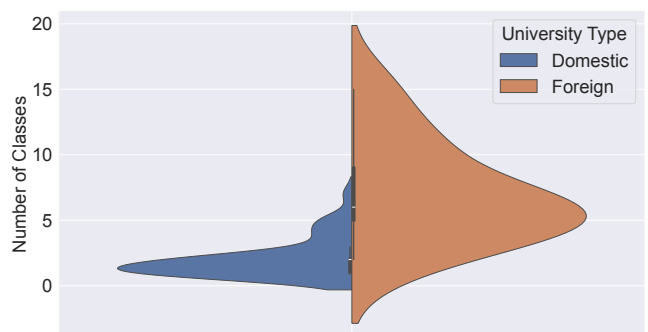


図 6: 国内・海外トップ大学におけるソフトウェア工学に関連する授業数の分布

講されていることに対し、海外トップ大学では 1 校あたり 6 科目であり、その開きは 3 倍にも上る。国内大学の調査対象科目 91 件に対して、科目名に「ソフトウェア工学」が含まれる授業は 52 科目 (57.1%)、海外トップ大学の調査対象科目 135 件に対し、科目名に “Software Engineering” が含まれるものは、47 科目 (34.8%) と大きな開きがある。これは、海外トップ大学ではソフトウェア工学全般の授業より、専門性の高い授業が多いことが示唆される。例えば、海外トップ大学では “Software Quality Assurance” や “Software Testing”, “AI Tools for Software Eng” といった専門的な授業が散見された。日本では専門的科目は修士課程で開講されることが多いが、海外トップ大学では学部生の段階で高度な授業を提供していると同える。

Answer to RQ2

海外トップ大学における *SWEBOK* 知識領域の網羅率は中央値 84.6% と高く、国内大学より 9.6 ポイント高い。一方、「運用」分野の講義実施率は海外でも低く、共通の課題であることが示された。また、国内大学では「ソフトウェア工学」と名のつく授業が多くを占めるが、海外トップ大学では「Software Quality Assurance」や「Software Testing」等、詳細な授業が多く、関連する授業数が日本の 3 倍であった。

6. 考察：国内と海外トップ大学のソフトウェア工学教育において生じる差違の原因

本研究では、国内大学と海外トップ大学のソフトウェア工学に関する講義の内容を目視調査し、その違いを明らかにした。RQ1では、国内大学のSWEBOK知識領域の網羅率は、75.0%であり、「アーキテクチャ」「運用」「構成管理」など一部の知識領域は、半数以上の大学で十分に教えられておらず、教育内容に偏りを確認した。また、これらの網羅率が低い三領域は、RQ2の海外トップ大学との比較においても著しく大きな差を確認された。特に、「運用」については、SWEBOKでもver.4において新たな需要の観点から追加された項目であり、「構成管理」と共に、DevOpsなど、モダンなソフトウェア開発では不可欠な要素である。

これらの問題を解決するためには、比較的新しい技術を授業に反映することが重要だと考えられる。しかし、シラバスの更新が日本の仕組み上容易ではないことなどが、阻害要因となる。例えば、日本の大学においては、授業科目を廃止・新設するには、文部科学省へ科目変更届の提出が必要である[11]。しかし、変更に関わる作業に多くの時間を必要とするため、新たな講義の追加において足枷になっている可能性がある。一方、例えば本調査でも対象となったPurdue Universityでは、新規授業の提案から承認まで約1ヶ月、さらに学内の認証のみで科目新設が可能である。^{*5}これは、海外の大学の多くはセメスター制であるため、新規科目を開講するにあたり早急な審議が必要であることも理由として考えられる。

Takeaway Message 1

「運用」などに関する授業が不足していることを認識し、学生の実践的なスキルを向上させ、産業界のニーズに対応するためにも、新たな授業の導入を検討することが望ましい。

これらは大学内のソフトウェア工学を教える教員数も関係していると考えられる。例えば、今回対象となった国内大学と海外トップ大学において、教員一人当たりのソフトウェア工学授業数の中央値は共に1講座であった。しかし、RQ2で示されたように海外トップ大学では、ソフトウェア工学関連の授業が多い。つまり、海外大学では、多数の教員で、広い範囲のソフトウェア工学領域をカバーしている。実際、大学あたりのソフトウェア工学授業担当者は国内大学が中央値で2人に対して、海外トップ大学は8人と大きく異なっていた。また、国内大学でソフトウェア工学授業担当者が海外大学の中央値（8人）を超える大学は存在しなかった。

^{*5} <https://www.purdue.edu/teaching-learning/academic-proposals/undergraduate/course.html>

この差異が発生する要因としては、ソフトウェア工学部やソフトウェア工学センターなど、ソフトウェア工学に特化した部門の有無などが影響していることが考えられる。例えば、今回調査対象にした海外トップ大学では、アメリカでは、Carnegie Mellon 大学にソフトウェア工学に特化したSoftware Engineering Instituteが存在し、Iowa State大学ではソフトウェア工学プログラムが存在している。今回、調査の対象外ではあるが、カナダではConcordia大学やÉcole Polytechnique de Montréal大学でソフトウェア工学部が存在する。一方、日本では、南山大学のみがソフトウェア工学部を有するなど、少数である。^{*6}日本では研究者が多くの大学に分散しており、詳細なソフトウェア工学を教えるリソースが枯渇し、広く浅いソフトウェア工学の授業を強いられている可能性がある。

Takeaway Message 2

ソフトウェア工学で教える内容が高度化し、その絶対量も多い現代において、従来の限られた時間で専門的かつ深い教育の実現は困難になりつつある。海外トップ大学ではソフトウェア工学関連の授業を担当する教員が数多く存在し、それぞれが専門的な教育を施すことで広い知識領域をカバーしている。一方、日本では1大学あたりのソフトウェア工学教育担当者が少なく、広く浅いソフトウェア工学の授業を実施せざるを得ない状況にある可能性がある。

7. 関連研究

シラバスの分析は、カリキュラムの質保証や見直しに有用であることが既存研究で示されており[12]、複数大学間でシラバスを比較することで、自国の教育上の強みや弱み、改善策のヒントになると指摘される[13]。近年ではシラバスをビッグデータとして分析するOpen Syllabus Projectも存在し、^{*7}国を跨いだカリキュラムの共通性や多様性などが明らかとなった。

ソフトウェア工学教育に注目した研究として、Kitchenhamらはソフトウェア工学のカリキュラムの有用性を調査するため、卒業生を対象にアンケート調査を行った[14]。調査結果では、Webプログラミングや構成管理などのカリキュラムの不足や、過剰に手厚く扱われた分野の存在を指摘し、カリキュラム見直しや、実践的スキルより重視すべきと述べている。RadaidehとAl-Haijaは、ヨルダン科学技術大学におけるソフトウェア工学系の学部で教育される科目が、SWEBOKをどれほど網羅するか調査した[15]。SWEBOKの各章がどれほどカリキュラムでカバーされて

^{*6} 南山大学は、大学・高専機能強化支援事業に採択されていないため、今回の分析対象に含まれていない。

^{*7} <https://www.opensyllabus.org/>

いるかに応じて4つの分類を行ったところ、ソフトウェア構築や構成管理など、合計4つの章が一部しか取り扱われていなかった。

これらの研究では、産業界のニーズと実際に大学で教えられる内容にはギャップが存在すると指摘しており、日本においても同様の指摘が存在する [16]。本研究も同様に、ソフトウェア開発の基礎的な講義（ソフトウェア設計など）は多数の大学で行われている一方、構成管理や DevOps のように、モダンなソフトウェア開発に関しては十分教育されておらず、シラバスの精査や改善が必要であると考えます。

産業界のニーズに対応するための取り組みとして、国外では SE2014 [17]、国内では J07-SE [18] や J17-SE [19] といったカリキュラム標準・モデルが策定されている。しかし、ソフトウェア工学分野におけるカリキュラム標準・モデルは J17-SE が策定された 2018 年の更新が最後である。そこで、本研究では 2024 年に更新されたばかりの SWEBOK ver.4 を利用することで、最新のソフトウェア開発において必要とされる知識をカバーした。

8. 妥当性への脅威

内的妥当性への脅威：本研究において、各科目がどの SWEBOK の知識領域に合致しているかは、著者らの目視分析によって調査している。目視分類の際に著者らの主観が混入している可能性がある。ただし、各自の目視後に、目視者全員で再確認および議論するなど、主観を除去する取り組みを行い、最小限の影響に止めている。また、調査対象の大学の規模は考慮していない。規模が大きな大学では網羅可能な知識領域が増える可能性がある。

構成的概念妥当性への脅威：本研究では各授業のシラバスの内容に、SWEBOK の各項目の内容が存在する場合、網羅していると判定した。そのため、それぞれの項目をどの程度深く教えているかは考慮していない。また、シラバス記載の充実度には差があり、不記載の内容を授業で実施している可能性もあることに注意されたい。

外部妥当性への脅威：本研究では情報教育を行う国内大学を抽出するために、「大学・高専機能強化支援事業」に採択された大学のみを抽出した。これらの大学で得られた結果が、全ての大学で一般化できるとは限らない。また、海外のトップ大学はソフトウェア工学の研究を行うトップレベルの大学から選定した。つまり、比較的充実したソフトウェア工学授業のある大学から結果を得ているため、一般的な海外トップ大学より高く評価されている可能性がある。

9. おわりに

本論文は、日本の大学におけるソフトウェア工学教育の現状を、国際的な標準である SWEBOK (Software Engineering Body of Knowledge) との対応状況や、海外のトッ

プ大学との比較を通じて明らかにすることを目的としている。調査の結果、日本の大学では SWEBOK に定義された知識領域のうち、中央値で 75.0% を網羅しており、全体としては一定の水準に達していることが分かった。しかし、「アーキテクチャ」「運用」「構成管理」といった実務に直結する分野については、半数以上の大学で十分に教えられておらず、教育内容に偏りが見られた。

一方、海外トップ大学では SWEBOK の網羅率が中央値で 84.6% と高く、より多くの知識領域をカバーしているだけでなく、1 大学あたりの授業数も日本の約 3 倍にのぼる。特に海外トップ大学では「Software Testing」や「Software Quality Assurance」など、専門的かつ実践的な授業が提供されており、教育の深さと幅の両面で日本との差が顕著である。ただし、「運用」分野に関しては海外トップ大学でも講義実施率が比較的低く、国際的にも共通する課題である。

謝辞 本研究は、JSPS における科研費 (JP22K18630, JP24K02921, JP24K02923, JP25K03100, 24K20905)、二国間交流事業 (JPJSBP120239929)、JST におけるさががけ (JPMJPR22P3) および ASPIRE (JPMJAP2415)、稲盛財団の稲盛科学研究機構 (InaRIS: Inamori Research Institute for Science) フェローシップの助成を受けた。

参考文献

- [1] 総務省. 令和 5 年版 情報通信白書 | 各国企業のデジタル化の状況. <https://www.soumu.go.jp/johotsusintokei/whitepaper/ja/r05/html/nd24b210.html>, 2023. Last accessed: 2025-07-21.
- [2] 西角直樹, 綿谷謙吾. デジタル赤字拡大は悪いことなのか? 目指すべきは「日本の強み」と「デジタル」の融合. <https://www.mri.co.jp/knowledge/insight/20240425.html>, 2024. Last accessed: 2025-07-21.
- [3] 松瀬滯奈, 齋藤誠, 森下謙太郎. 国際収支統計からみたサービス取引のグローバル化. https://www.boj.or.jp/research/wps_rev/rev_2023/data/rev23j09.pdf, 2023. Last accessed: 2025-07-21.
- [4] 2023 年版 ソフトウェアデリバリー に関する現状調査 - 日本版. Technical report, CircleCI, 2023.
- [5] 独立行政法人情報処理推進機構. Dx 動向 2024 進む取り組み、求められる成果と変革. Technical report, 独立行政法人 情報処理推進機構, 2024.
- [6] M. Kuhrmann, J. Nakatumba-Nabende, R. Pfeiffer, P. Tell, J. Klünder, T. Conte, S. G. MacDonell, and R. Hebig. Walking through the method zoo: Does higher education really meet software industry demands? In *Proc. of ICSE-SEET*, pp. 1–11, 2019.
- [7] H. Washizaki, editor. *Guide to the Software Engineering Body of Knowledge (SWEBOK Guide), Version 4.0*. IEEE Computer Society, 2024.
- [8] P. Naur and B. Randell. *Software Engineering: Report of a conference sponsored by the NATO Science Committee*. Scientific Affairs Division, NATO, 1969.
- [9] H. M. M. Ramirez and C. V. Hurtado. Work in progress: Design and implementation of a microservices architecture for project-based learning of software engineering patterns. In *Proc. of EDUCON*, pp. 1–3, 2023.
- [10] Y. Demchenko, Z. Zhao, J. Surbiryala, S. Koulouzis,

- Z. Shi, X. Liao, and J. Gordiyenko. Teaching devops and cloud based software engineering in university curricula. In *Proc. of eScience*, pp. 548–552, 2019.
- [11] 文部科学省. 変更届出要領. "https://www.mext.go.jp/a_menu/koutou/kyoin/080718_2/001.pdf", 2025. Last accessed: 2025-07-21.
- [12] 渡辺雄貴. 学習成果に基づく授業設計の視点から見たシラバスの内容分析. *大学評価研究*, Vol. 13, pp. 113–122, 2014.
- [13] Z. Liu and T. Jia. A comparison of the chinese and american undergraduate syllabi in regard to their content and style. In *Proc. of ICMES*, pp. 228–234, 2017.
- [14] A. Amiri, M. Banari, and N. Yousefnezhad. An investigation of undergraduate software engineering curriculum: Iranian universities case study. In *Proc. of ICCSE*, 2011.
- [15] M. A. Radaideh and Q. A. Al-Haija. Evaluating the software engineering curriculum at just: A comparative analysis with iee knowledge areas. *IJISAE*, Vol. 12, No. 4, pp. 583–593, 2024.
- [16] 掛下哲郎, 大月美佳. J07 フォローアップ調査：情報系学生に対する要求と達成度の調査と分析. *電気関係学会九州支部連合大会講演論文集*, Vol. 2011, pp. 298–299, 2011.
- [17] Mark Ardis, David Budgen, Gregory W. Hislop, Jeff Offutt, Mark Sebern, and Willem Visser. Se 2014: Curriculum guidelines for undergraduate degree programs in software engineering. *Computer*, Vol. 48, No. 11, pp. 106–109, 2015.
- [18] 阿草清滋, 西康晴, 沢田篤史, 鷲崎弘宜. 情報専門学科カリキュラム標準「j07」：4. ソフトウェアエンジニアリング領域 (j07-se) . *情報処理*, Vol. 49, pp. 743–749, Jul 2008.
- [19] カリキュラム標準 J17：ソフトウェアエンジニアリング領域策定 WG. カリキュラム標準 j17：ソフトウェアエンジニアリング領域 (j17-se) . https://www.ipsj.or.jp/annai/committee/education/j07/ed_j17-SE.html, 2018. Last accessed: 2025-07-22.