

Ähnlichkeitsbasiertes Retrieval von BPMN-2.0-Modellen

Maximilian Pfister, Florian Fuchs und Ralph Bergmann

Universität Trier, Wirtschaftsinformatik II,
Universitätsring, 54286 Trier, Germany
[s4mapfis] [s4flfuch] [bergmann]@uni-trier.de,
<http://www.wi2.uni-trier.de>

Zusammenfassung. Business Process Model and Notation (BPMN) 2.0 gilt als einer der vielversprechendsten Standards zur Prozessmodellierung, jedoch ist die Modellierung auch für Experten mit großem Aufwand verbunden, sodass heute versucht wird, die Modellierung durch Wiederverwendung von Prozessmodellen zu unterstützen. Hierzu werden Repositories von qualitativ hochwertigen Geschäftsprozessmodellen aufgebaut, aus denen wiederverwendbare Modelle ausgewählt werden können. In dieser Arbeit wird ein Ansatz aus dem prozessorientierten Fallbasierten Schließen für das ähnlichkeitsbasierte Retrieval von BPMN-2.0-Modellen erweitert und empirisch erprobt. Anhand eines existierenden BPMN-Repositories wird überprüft, welchen Mehrwert hierbei die Einbeziehung der Semantik in die Ähnlichkeitsberechnung gegenüber einer rein lexikalischen Bewertung erbringt.

1 Einleitung

Modellierung und Analyse von Geschäftsprozessen sind wichtige Aufgaben für moderne Unternehmen. Es existieren heute verschiedene Notationen zur Prozessmodellierung. Nach Allweyer [3] unterscheidet man folgende Standards: XPD (XML Process Definition Language) und BPEL (Business Process Execution Language) sind Prozessausführungssprachen, die hauptsächlich zur Definition von automatisierten Prozessen genutzt werden. EPC (Event-driven Process Chain) ist eine Notation mit einem stärkeren Fokus auf Geschäftsprozessmodellierung. Diese Notation ist jedoch kein offener Standard. Bei der Business Process Model and Notation (BPMN) steht die Modellierung und Visualisierung von Geschäftsprozessen im Vordergrund. “Die Notation wird sowohl für fachliche Prozessmodelle [(Geschäftsprozesse), Anm. d. Verf.] verwendet als auch für detaillierte Ablaufspezifikationen [(Workflows), Anm. d. Verf.], die durch Process Engines ausgeführt werden” [2].

BPMN hat in der aktuellen Version 2.0 eine weitere Verbreitung erfahren, was dazu geführt hat, dass umfangreichere Prozessrepositorien in diesem Format entstehen [11]. Die Wiederverwendung von Prozessmodellen stellt eine vielversprechende Möglichkeit dar, um die Produktivität von Prozessgestaltern zu steigern, den Designprozess zu beschleunigen und Redundanzen in Prozessdatenbanken zu vermeiden [10]. Um ein Retrieval von gespeicherten Geschäftsprozessen zu ermöglichen, ist es notwendig, Ähnlichkeitsmerkmale für Prozesse zu definieren, um diese vergleichbar zu machen.

Idealerweise wird das Prozessmodell aus dem Repository ausgesucht, das die Anforderungen der aktuellen Suchanfrage am besten erfüllt, sodass anschließend nur noch wenige Anpassungen vorgenommen werden müssen, um ein geeignetes Prozessmodell zu erhalten. Es werden neue Methoden zur Ähnlichkeitsberechnung und zum Retrieval von Prozessmodellen benötigt [4], um Experten dahingehend zu unterstützen, dass sie nicht nur strukturell, sondern auch semantisch verwandte Prozesse finden können. Prozessorientiertes Fallbasiertes Schließen stellt einen Ansatz dar, um die Wiederverwendung von Prozessmodellen zu ermöglichen [14].

Fallbasiertes Schließen (Case-based Reasoning, CBR) [1,6,15] ist eine Technik, die erfahrungsbasiertes Problemlösen unterstützt. Neue Probleme werden dadurch gelöst, dass die Lösungen ähnlicher Probleme aus einer Fallbasis abgerufen und angepasst werden. Prozessorientiertes Fallbasiertes Schließen (Process-oriented Case-based Reasoning, POCBR) befasst sich mit der Verbindung von CBR mit prozessorientierten Informationssystemen [14]. POCBR ermöglicht die Erstellung neuer Prozessmodelle durch Auswahl (Retrieval) von bestehenden, zu einer Anfrage ähnlichen, Prozessmodellen aus der Fallbasis (Repository). Hierbei geht man davon aus, dass als Anfrage bereits ein partielles Prozessmodell erstellt wurde, das einige wichtige Bestandteile enthält, aber noch nicht vollständig alle notwendigen Prozesselemente zu einem konsistenten Prozessmodell verbindet. Bergmann und Gil [7] haben einen Ansatz zum ähnlichkeitsbasierten Retrieval für das POCBR entwickelt, der auf blockorientierten und datenflussbasierten Workflows basiert. Dieses Paper hat zum Ziel, diesen Ansatz auf BPMN 2.0 zu übertragen und dadurch auszuweiten. Hierzu ist es notwendig, sowohl die BPMN-Elemente als auch die Charakteristik von BPMN als graphenorientierte Modellierungssprache bei der Ähnlichkeitsberechnung zu berücksichtigen. Die Ähnlichkeitsberechnung für Prozessmodelle erfordert darüber hinaus ein lokales Ähnlichkeitsmaß zum Vergleich von Knoten und Kantenlabels der Prozessmodelle. Dies ist für bestehende BPMN-Modelle schwierig, da diese Labels überwiegend aus Freitexten bestehen. Ein sorgfältig modelliertes Ähnlichkeitsmaß ist entscheidend für eine gute Retrievalqualität, jedoch ist dessen Entwicklung mit großem Aufwand verbunden. In dieser Arbeit werden daher die Ergebnisse eines einfachen lexikalischen Ähnlichkeitsmaßes mit denen eines semantischen Ähnlichkeitsmaßes im Rahmen einer empirischen Untersuchung verglichen.

2 Block- vs. graphenorientierte Prozessmodelle

Blockorientierte Modellierungssprachen definieren den Kontrollfluss über strukturierte Aktivitäten, durch deren Verschachtelung komplexe Abläufe mit Sequenzen, Alternativen und Schleifen erstellt werden können [12]. Ein Beispiel für blockorientierte Modellierungssprachen ist BPEL [12]. Graphenorientierte Modellierungssprachen wie BPMN definieren den Kontrollfluss über Kanten, die die zeitlichen und logischen Abhängigkeiten zwischen Knoten repräsentieren [12]. Knoten können Start- und Endzustände sowie Aktivitäten oder Konnektoren sein. Konnektoren lassen sich wiederum in *AND*, *OR* und *XOR* unterteilen, welche den Kontrollfluss verzweigen (*SPLIT*) oder vereinigen (*JOIN*) können. Graphenorientierte Modellierungssprachen gelten als ausdrucksstärker als blockorientierte, da sie Abfolgen von Aktivitäten durch die Verkettung von Knoten mit gerichteten Kanten darstellen können [13]. Dies ermöglicht beispielsweise die Kon-

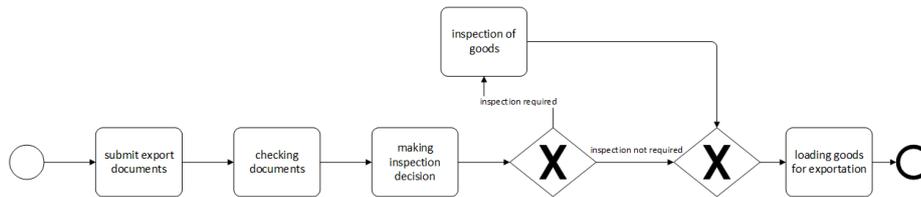


Abb. 1. BPMN-Prozessmodell aus der BPMAI

struktion von zyklischen Kantenfolgen mit mehreren Ein- und Ausstiegspunkten, was in blockorientierten Sprachen nicht realisierbar ist [12]. Abbildung 1 zeigt ein BPMN-Prozessmodell aus der BPM Academic Initiative (BPMAI) [11]. Das Prozessmodell beschreibt einen Exportvorgang im Güterverkehr. Das Beispiel zeigt die Steuerung des Kontrollflusses über die Kanten des Prozessmodells (“inspection required” und “inspection not required”). BPMN besitzt verschiedene grafische Elemente, welche sich in *Flow Objects*, *Connecting Objects*, *Pools/Swimlanes* und *Artifacts* unterteilen [3]. *Flow Objects* sind die Knoten des Prozessmodells; dazu zählen Aktivitäten, Konnektoren und Ereignisse. Aktivitäten werden als Rechteck dargestellt und beschreiben die Aufgaben eines Geschäftsprozesses. Konnektoren werden als ein auf der Spitze stehendes Quadrat dargestellt. Ereignisse (wie z. B. Start des Exportprozesses) haben unterschiedliche Symbole und werden kreisförmig dargestellt. Da das Konzept von Bergmann und Gil [7] ebenfalls graphenbasiert ist und somit die Definition von Knoten und Kanten unterstützt, lässt sich die grundlegende Struktur von BPMN Diagrammen ohne Probleme abbilden.

3 Repräsentation von BPMN-Modellen als Prozessgraphen

Wir beschreiben nun eine graphbasierte Repräsentation von BPMN-Modellen, die sich am Konzept von Bergmann und Gil [7] orientiert, jedoch einige BPMN 2.0-spezifische Erweiterungen hinzufügt. Ein Geschäftsprozess wird als semantisch annotierter gerichteter Graph dargestellt. Ein solcher *semantischer Prozessgraph* W ist ein Quadrupel $W = (N, E, S, T)$, bei dem N die Knoten und $E \subseteq N \times N$ die Kanten des Prozessmodells repräsentieren. $S : N \cup E \rightarrow \Sigma$ assoziiert zu jedem Knoten und jeder Kante eine semantische Beschreibung aus einer semantischen Metadatensprache Σ . Die Verwendung einer bestimmten Sprache für Σ ist nicht festgelegt; einzige Bedingung ist, dass ein Ähnlichkeitsmaß für diese Sprache erstellt werden kann. In dieser Arbeit wird für Σ eine Taxonomie der Knoten und Kantenbeschreibungen verwendet. $T : N \cup E \rightarrow \Omega$ weist jedem Knoten und jeder Kante einen der folgenden Typen aus Ω zu:

- Jeder Prozess besitzt genau einen *Prozessknoten*. Dieser enthält allgemeine Informationen, die den gesamten Prozess betreffen, wie z. B. eine semantische Beschreibung in Form von Tags, einer Einordnung in eine Ontologie oder Kennzahlen zur Performance, Qualität und Zuverlässigkeit des Prozessmodells.
- Jede Aufgabe eines Prozesses wird durch einen *Aufgabenknoten* repräsentiert, dessen Aufgabe durch die semantische Beschreibung näher spezifiziert wird. Der Aufgabenknoten bietet außerdem die Möglichkeit, einen Adressaten zur Ausführung

der Aufgabe festzulegen. Dies ist eine Übertragung der Rollenverteilung, die in BPMN üblicherweise über Pools und Lanes realisiert wird.

- Datenobjekte werden durch *Datenknoten* repräsentiert. Die semantische Beschreibung dieser Knoten dient der Klassifizierung, z. B. durch eine Datentypenontologie. Datenmodellierung steht jedoch nicht im Mittelpunkt von BPMN 2.0, was sich auch an der überschaubaren Anzahl von Datenobjekten (einfaches Datenobjekt, Datenlistenobjekt, Dateninput, Datenoutput, Datenspeicher) zeigt. Außerdem werden die Zuordnung von Verantwortlichkeiten repräsentiert, um Pools und Lanes zu berücksichtigen.
- Kontrollflussobjekte, wie zum Beispiel das XOR-Gateway oder das Parallele Gateway, werden durch *Kontrollflussknoten* repräsentiert.

Zusätzlich sind folgende Kantentypen möglich:

- Der Prozess-Knoten ist mit jedem anderen Knoten mit einer *Part-Of-Kante* verbunden. Die semantische Beschreibung einer solchen Kante beschreibt die Rolle des betreffenden Knotens im Prozessmodell.
- Der Kontrollfluss zwischen Aufgabenobjekten wird durch *Kontrollflusskanten* repräsentiert. Kontrollflusskanten verbinden entweder zwei Aufgabenknoten oder einen Aufgabenknoten mit einem Kontrollflussknoten. Kontrollflusskanten legen die Reihenfolge fest, in der Knoten ausgeführt werden. Der Startknoten einer Kante muss immer vor dem Endknoten der Kante ausgeführt werden.
- Der Datenfluss zwischen Aufgaben- und Datenobjekten wird durch *Datenflusskanten* repräsentiert. In BPMN 2.0 existieren verschiedene Elemente, um Datenfluss zu modellieren, wie zum Beispiel gerichtete und beidseitige Assoziation und Nachrichtenfluss. Die semantische Beschreibung der Kante legt fest, um welches Element es sich handelt. Für gerichtete Datenflüsse gilt: Ist der Startknoten der Kante ein Datenknoten und der Endknoten ein Aufgabenknoten, liegt ein lesender Zugriff vor, im umgekehrten Fall ist es ein schreibender Zugriff.

Abbildung 2 zeigt das Prozessmodell aus dem vorherigen Abschnitt, dargestellt als Prozessmodellgraph. Aufgabenknoten sind eckig dargestellt, Kontrollflussknoten haben eine ovale Form, *n1* ist der Prozess-Knoten.

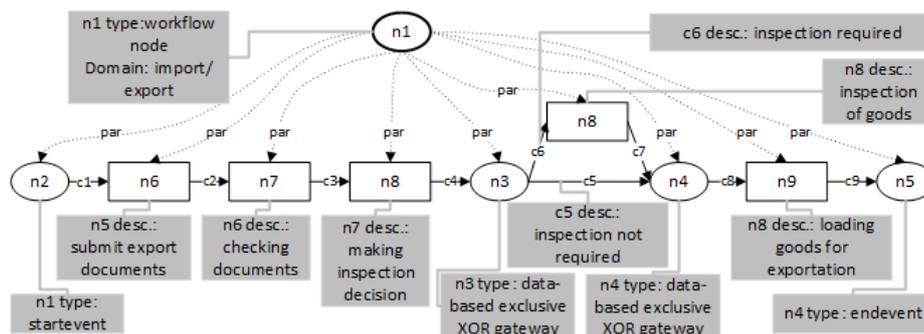


Abb. 2. Beispielhafter Prozessgraph

4 Ähnlichkeit von BPMN-Modellen

Eine Fallbasis (Repository), der Kern eines POCBR-Ansatzes, entspricht einer Menge von Prozessmodellgraphen $CB = \{CW_1, \dots, CW_n\}$, die dieselbe semantische Metadatensprache Σ besitzen. Auch eine Anfrage ist ein solcher Prozessmodellgraph, der aber nicht vollständig spezifiziert sein muss, d. h. es können nur einige Aktivitäten oder Datenknoten angegeben werden; auch die Verknüpfung dieser Kanten muss nicht vollständig sein oder kann auch ganz fehlen. Aufgabe des Retrievals im POCBR ist es nun, die ähnlichsten Prozessmodellgraphen zur Anfrage zu finden.

Um die Ähnlichkeit eines Prozesses aus der Fallbasis (Case) $CW = (N_c, E_c, S_c, T_c)$ zu einer Anfrage (Query) $QW = (N_q, E_q, S_q, T_q)$ beurteilen zu können, müssen sowohl die einzelnen Prozesselemente als auch die Verknüpfungsstruktur berücksichtigt werden. Wir verwenden in dieser Arbeit das Ähnlichkeitsmodell von Bergmann und Gil [7], das eine Erweiterung des Lokal/Global-Ansatzes für strukturelles CBR [8,6] darstellt. Das lokale Ähnlichkeitsmaß bestimmt die Ähnlichkeit zwischen zwei Knoten oder Kanten. Die globale Ähnlichkeit wird durch eine Aggregationsfunktion berechnet, die die lokalen Ähnlichkeitswerte unter Berücksichtigung eines geeigneten Mappings zwischen Query und Prozessmodell zusammenfasst.

In dieser Arbeit werden zwei Ansätze zur Berechnung der lokalen Ähnlichkeiten verglichen. Zum einen wird eine lexikalische Ähnlichkeitsfunktion auf Basis der Levenshtein-Distanz der Knoten- und Kantenbeschreibungen verwendet und zum anderen kommt die semantische Ähnlichkeitsfunktion, basierend auf den semantischen Beschreibungen der Knoten und Kanten, zum Einsatz. Die lexikalische Ähnlichkeit ist domänenunabhängig, wohingegen die semantische Ähnlichkeit für den jeweiligen Gegenstandsbereich spezifisch modelliert werden muss. In dieser Arbeit wird für die Knoten und Kanten jeweils eine Taxonomie der Knoten- und Kantenbeschreibungen (Ontologie) genutzt, die händisch für den untersuchten Teilbereich des Repositories modelliert wurden. Abbildung 3 zeigt einen Teilausschnitt der Knotentaxonomie. Die Ähnlichkeit basiert nun wie folgt auf dieser Taxonomie: Die Ähnlichkeit zweier Beschreibungen hat den Wert 1, wenn diese identisch sind; ansonsten wird die Ähnlichkeit dadurch bestimmt, dass der gemeinsame Oberknoten in der Taxonomie ermittelt wird. Dieser wird mit einem festen Ähnlichkeitswert annotiert, der diese Ähnlichkeit auf einfache Weise repräsentiert (z. B. $sim(\text{“start investigation”}, \text{“x-ray inspection”}) = 0.8$, siehe [5]). Liegen zwei Knoten unterschiedlichen Typs vor, so ist die Ähnlichkeit grundsätzlich 0.

Bei der Berechnung der Kantenähnlichkeit werden nicht nur die semantischen Beschreibung der betrachteten Kanten berücksichtigt, sondern auch die Ähnlichkeit der Knoten, die diese verbinden. Zwei Kontrollflusskanten sollten nur ähnlich sein, wenn diese auch ähnliche Aufgabenknoten als Start- und Endknoten verbinden. Die Funktion $F_E(S_e, S_l, S_r)$ ist eine Aggregationsfunktion, die die semantische Ähnlichkeit der Kanten S_e sowie der Startknoten S_l und Endknoten S_r zu einem Ähnlichkeitswert zusammenfasst. Für F_E wird folgende Funktion verwendet: $F_E(S_e, S_l, S_r) = S_e \cdot 0,5 \cdot (S_l + S_r)$. Die Kantenähnlichkeit und die Ähnlichkeiten der verbundenen Knoten gehen also jeweils zur Hälfte in die Berechnung der aggregierten Kantenähnlichkeit ein.

Die Grundlage der Ähnlichkeitsberechnung zwischen einer Query QW und einem Prozessmodell CW bildet ein zulässiges Mapping m [7]. Ein zulässiges Mapping ist

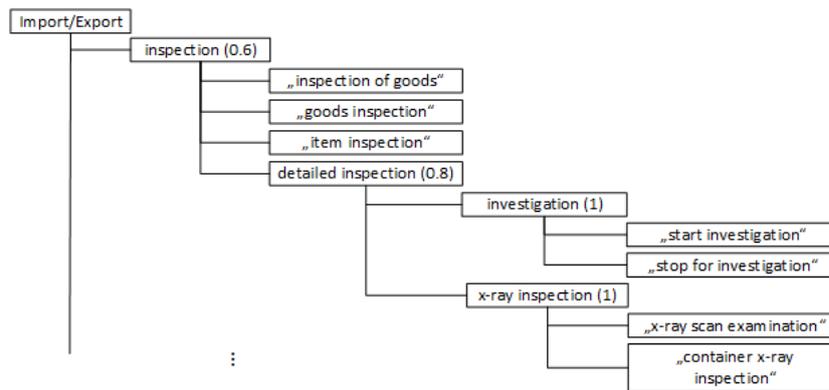


Abb. 3. Teilausschnitt der Import/Export-Taxonomie

eine typerhaltende, partielle Abbildung $m : N_q \cup E_q \rightarrow N_c \cup E_c$. Ein solches Mapping muss die Bedingungen erfüllen, dass ein Knoten oder eine Kante aus QW jeweils nur auf einen Knoten oder eine Kante aus CW abgebildet wird. Außerdem können Kanten nur zugewiesen werden, wenn die von der Kante verbundenen Knoten auch entsprechend als Start- und Zielknoten der Zielkante zugewiesen wurden. Für ein gegebenes Mapping können somit die Elemente der Query auf die Elemente im Prozessmodell abgebildet werden, so dass nun eine Ähnlichkeitsberechnung der zugeordneten Elemente möglich ist. Die resultierenden lokalen Ähnlichkeitswerte können dann zur Gesamtähnlichkeit aggregiert werden. Um die Ähnlichkeit zu bestimmen, ist somit die Kenntnis eines Mappings erforderlich. Diese kann gemäß des von Bergmann und Gil [7] beschriebenen heuristischen Suchverfahrens so bestimmt werden, dass die resultierende Ähnlichkeit maximiert wird.

5 Empirische Evaluation

Ziel der Evaluation ist es herauszufinden, ob das Ranking von Prozessmodellen durch die beschriebene Ähnlichkeitsberechnung vergleichbare Ergebnisse liefert, wie das Ranking von Experten. Des Weiteren werden die Ergebnisse des semantischen Ähnlichkeitsmaßes mit denen des lexikalischen Ähnlichkeitsmaßes verglichen. Zur Erstellung der Fallbasis wurden die Prozessmodelldatenbanken der BPMMAI [11] analysiert. Um eine einheitliche Fallbasis von hoher Qualität zu erreichen wurden nur Prozessmodelle betrachtet, die eine detaillierte Knotenbeschreibung in englischer Sprache beinhalten und vollständig sind (Start- und Endpunkt sind vorhanden, es gibt keine offenen Abzweigungen). Eine detaillierte Beschreibung der Aktivitätsknoten ist von großer Bedeutung, da die Ähnlichkeitsberechnung auf diesen aufbaut. Die gefundenen Prozessmodelle wurden nach Domänen gruppiert. Die Fallbasis besteht aus den ausgewählten Domänen *Import/Export*, *Abrechnungen* und *Bestellprozesse*, wobei jede Domäne 15-20 Prozessmodelle enthält.

Zu jeder Domäne wurden fünf plausible Suchanfragen händisch formuliert. Die Anfragen sind meist kleine, leicht abgeänderte Teilausschnitte der Prozessmodelle aus der

Fallbasis. Zu jeder dieser Anfragen wurden von Experten (Studierende der Wirtschaftsinformatik) die fünf ähnlichsten Prozessmodelle aus der Fallbasis ausgesucht und nach Ähnlichkeit sortiert. Die Ranglisten der Experten wurden mit den Ergebnissen der Ähnlichkeitsberechnungen verglichen. Die Bewertung der Ähnlichkeit erfolgt mit Hilfe der Evaluationskriterien *Korrektheit* und *Vollständigkeit* nach der Ranglistenberechnung von Cheng et al. [9]. Der Wert für die Korrektheit liegt im Intervall $[-1,1]$. Sofern beide Ranglisten übereinstimmen, ist der Wert 1. Wenn beide Ranglisten gegensätzlich sind, liegt der Wert bei -1. Der Wert für die Vollständigkeit bewegt sich im Intervall $[0,1]$ und gibt an, wie viele Ordnungen des Expertenrankings durch das Ähnlichkeitsmaß ebenfalls geordnet wurden. Die Tabelle 1 zeigt die Ergebnisse der Evaluation für die ausgewählten Domänen. Die dargestellten Ergebnisse für Korrektheit und Vollständigkeit sind jeweils das arithmetische Mittel aus den Einzelergebnissen der fünf Anfragen pro Domäne.

	Import/Export		Bestellungen		Abrechnungen	
Ähnlichkeitsmaß	Korr.	Voll.	Korr.	Voll.	Korr.	Voll.
lexikalisch	0,662	1,000	0,596	1,000	0,351	1,000
semantisch	0,751	1,000	0,796	0,793	0,770	0,860

Tabelle 1. Evaluation der Ähnlichkeitsmaße

Die Korrektheit des semantischen Ähnlichkeitsmaßes liegt in allen Domänen über dem des lexikalischen Maßes, am deutlichsten in der Domäne *Abrechnungen*, in der das lexikalische Ähnlichkeitsmaß mit einem Wert von 0.351 kaum zu brauchbaren Ergebnissen führt. Die Vollständigkeit ist hingegen beim lexikalischen Ähnlichkeitsmaß in allen Domänen maximal, was darin begründet ist, dass aus dem Levenshtein-Vergleich von zwei Knotenlabels sehr differenzierte Ähnlichkeitswerte resultieren, wohingegen beim semantischen Ähnlichkeitsmaß nur die in der Taxonomie hinterlegten Ähnlichkeitswerte vorkommen. Somit bewertet das lexikalische Maß nur selten zwei Prozessmodelle in der Fallbasis mit der gleichen Ähnlichkeit, sodass grundsätzlich zwischen allen Fällen eine Ordnungsrelation besteht. Auch wenn diese nicht korrekt ist, führt dies zu einer Vollständigkeit von 1. Beim semantischen Ähnlichkeitsmaß kommt es hingegen vor, dass zwei Prozessmodelle als gleich ähnlich eingestuft werden, obwohl Experten dies nicht so bewerten. Ein stärker differenzierendes semantisches Ähnlichkeitsmaß könnte die Ergebnisse für die Vollständigkeit verbessern. Insgesamt ist zur Einschätzung des praktischen Nutzens aber die Korrektheit ein wichtigeres Kriterium als die Vollständigkeit. Bei Verwendung des semantischen Ähnlichkeitsmaßes wird daher eine deutlich bessere Übereinstimmung mit der Einschätzung der Experten erzielt.

6 Fazit und Ausblick

Diese Arbeit setzt an der aktuellen Problemstellung an, Prozessdatenbanken mittels Anfragen nach ähnlichen Prozessmodellen zu durchsuchen, um die Modellierung von neuen Prozessmodellen zu erleichtern. Der vorgestellte Ansatz auf Basis von prozessorientiertem Fallbasierten Schließen eignet sich zur Ähnlichkeitsberechnung und zum Retrieval von Prozessmodellen im BPMN-2.0-Format. Die Einbeziehung der Semantik

von Prozessmodellen hat sich als gute Möglichkeit erwiesen, um die Retrievalqualität des Systems gegenüber einer rein syntaktischen Auswertung zu steigern. Dies wird jedoch erkauft durch den höheren Aufwand zur Modellierung der Ähnlichkeitsmaße mittels Taxonomien. Die manuelle Erstellung von Taxonomien könnte sich bei großen Repositorien auf Grund des Modellierungsaufwands als impraktikabel erweisen. Als Ansatzpunkt für weitere Forschungsarbeiten bietet es sich daher an, alternative Verfahren zu untersuchen, die keine manuelle Modellierung erfordern, um die semantischen Ähnlichkeiten zwischen Labels in Prozessmodellen zu bestimmen. Es könnte beispielsweise ein semantisches Vektorraummodell zur Ähnlichkeitsberechnung genutzt werden, das zusätzlich durch Fachtexte domänenspezifisch angereichert werden könnte.

Literatur

1. Aamodt, A., Plaza, E.: Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI communications* 7(1), 39–59 (1994)
2. Allweyer, T.: BPMN-Prozessmodelle und Unternehmensarchitekturen. Untersuchung von Ansätzen zur Methodenintegration und ihrer Umsetzung in aktuellen Modellierungstools. Forschungsbericht, Hochschule Kaiserslautern, <http://www.kurze-prozesse.de/blog/wp-content/uploads/2014/11/BPMNProzessmodelle- und-Unternehmensarchitekturen.pdf>
3. Allweyer, T.: BPMN 2.0: introduction to the standard for business process modeling. BoD–Books on Demand (2010)
4. Becker, M., Laue, R.: A comparative survey of business process similarity measures. *Computers in Industry* 63(2), 148 – 167 (2012)
5. Bergmann, R.: On the Use of Taxonomies for Representing Case Features and Local Similarity Measures. In: Gierl, L., Lenz, M. (eds.) *Proceedings of the 6th German Workshop on Case-Based Reasoning (GWCBR'98)* (1998)
6. Bergmann, R.: *Experience management: foundations, development methodology, and internet-based applications*. Springer-Verlag (2002)
7. Bergmann, R., Gil, Y.: Similarity assessment and efficient retrieval of semantic workflows. *Information Systems* 40, 115–127 (2014)
8. Burkhard, H.D., Richter, M.M.: On the notion of similarity in case based reasoning and fuzzy theory. In: *Soft computing in case based reasoning*, pp. 29–45. Springer (2001)
9. Cheng, W., Rademaker, M., De Baets, B., Hüllermeier, E.: Predicting partial orders: ranking with abstention. In: *Machine Learning and Knowledge Discovery in Databases*, pp. 215–230. Springer (2010)
10. Koschmider, A., Fellmann, M., Schoknecht, A., Oberweis, A.: Analysis of process model reuse: Where are we now, where should we go from here? *Decision Support Systems* 66, 9 – 19 (2014)
11. Kunze, M., Berger, P., Weske, M.: BPM academic initiative - fostering empirical research. In: *Proceedings Demonstration Track - 10th International Conference on Business Process Management*. CEUR Workshop Proceedings, vol. 940, pp. 1–5. CEUR-WS.org (2012)
12. Mendling, J., Lassen, K.B., Zdun, U.: On the transformation of control flow between block-oriented and graph-oriented process modeling languages. *International Journal of Business Process Integration and Management* 3(2), 96–108 (2006)
13. Mendling, J., Reijers, H.A., van der Aalst, W.M.: Seven process modeling guidelines (7pmg). *Information and Software Technology* 52(2), 127–136 (2010)
14. Minor, M., Montani, S., Recio-Garcia, J.A.: Process-oriented case-based reasoning. *Information Systems* 40, 103 – 105 (2014)
15. Richter, M.M., Weber, R.O.: *Case-Based Reasoning - A Textbook*. Springer (2013)