Master's Thesis

IU International University of Applied Sciences

M.Sc. Data Science 120 ECTS

Opportunities for Cognitive Modeling in Software Project Management: The Value of Applying Cognitive Science to Conceptual Models

Aaron K. Althauser

Enrolment number: 92014910

1410 NE Campus Parkway

Seattle, WA 98195

Supervisor: Prof. Dr. Markus C. Hemmer

Date of submission: 3rd September 2023

Abstract

Purpose. To highlight and formalize the interconnected relationships between cognitive modeling (CM), software project management (SPM), and behavioral and cognitive science.
Value. The study builds on the existing research efforts of SPM modeling by extracting an interdisciplinary cognitive approach and applying it to preexisting conceptual models.
Methods. Systematic Literature Review (SLR), text mining, clustering
Key findings. CM techniques have the capacity to reproduce a wide scope of cognitive processes, many of which are useful to take into consideration in the modeling and simulation of human-computer interactive behavior. While there exists separated bodies of research within each of the study domains, there is an incongruity amongst the modeling community between the semantic distinctions of cognitive and computational modeling.
Conclusion. There is a strong relationship between CM, SPM, HCI, and cognitive science that is under-represented in current cognitive science and PM research. The implementation of cognitive theories into models that replicate SPM processes is a key area of future cognitive and behavioral science studies.

Keywords: cognitive modeling, software project management, behavioral science, cognitive science, human-computer interaction

Table of Contents

List of Figures

List of Tables

List of Abbreviations

BPM	Business Process Model
CM	Cognitive Modeling
EPIC	Executive Process-Interactive Control
EUT	Expected Utility Theory
FCM	Fuzzy Cognitive Map
GOF	Goodness of Fit
HCI	Human-Computer Interaction
ICCM	International Conference for Cognitive Modeling
IS	Information System
ITSM	IT Service Management
kNN	K-Nearest Neighbors
MADM	Multi Attribute Decision Making
MHP	Model Human Processor
ML	Machine Learning
OR	Operational Research
PM	Project Management
RQ	Research Question
RUT	Random Utility Theory
SD	Software Development
SDLC	Software Development Life Cycle
SE	Software Engineering

SLR	Systematic Literature Review
SPM	Software Project Management
SPMNet	Software Project Management Net
TF-IDF	Term Frequency-Inverse Document Frequency

1 Introduction

Guided by research in the multidisciplinary field of cognitive science, experts have been able to apply theoretical conclusions to the optimization of tasks within the professional and academic fields both; and in doing so, have advanced the understanding of computational and cognitive psychology beyond the theoretical and into the applied science realm. The comprehension of the dynamic aspects of human cognition and behavior realized in the management of software projects and processes is one of the worthwhile applications of this knowledge. This paper will act as an exploration into the interconnection between the topics of cognitive modeling (CM), cognitive and behavioral psychology, and software project management (SPM).

The study of cognition provides many challenges; human cognition being a multidimensional system with both complex and simple, deliberate and automatic, fast and slow mental processing. By synthesizing various theoretical approaches and mechanisms of cognition into proposed unified structures, called cognitive architectures, computational cognitive models allow for the methodical and thoroughly testable reproductions of human cognitive processes. Starting from fundamental conceptual models of cognition, interrelated cognitive mechanisms can be modeled and applied to specific tasks and the behavioral outputs of the tasks can be tested, verified, and fed back into the conceptual understanding of the models.

Supplemental to CM research, cognitive psychology studies the core processes behind language and knowledge formation, attention, decision-making, problem-solving, memory, and learning, to understand how individuals process and perceive stimuli in their environment, utilize and acquire information, and how they think, reason, and respond to inputs that influence behavior. The disciplines of neuroscience, neuropsychology, and computer science, i.e., computational psychology, are all included in CM to provide an understanding of how cognitive mechanisms affect human action.

Behavioral psychology complements cognitive psychology by exploring human behavior as an external, observable output. In perspective to the study of internal processes, behavioral psychology relies on the social and external aspects. This research scope will examine behavioral psychology from the lens of human-computer interaction (HCI), which focuses on the behavior and cognition that results from humans' interactions and collaborations with digital tools to perform tasks and complete goals. A narrower lens in behavioral psychology will also be employed in this research scope, one that observes the patterns of behavior stemming from common biases and heuristics, specifically in the context of SPM.

Outside of cognitive and behavioral sciences, software project managers play a vital role in controlling and delivering software projects. SPM contains processes for managing tasks throughout the software development life cycle (SDLC) which includes design, coding, testing, and deployment. Specific processes in SPM are based on communication and resource control: planning, scheduling, budgeting, risk management, and quality assurance. The role of a software project manager is intertwined with cognitive science in terms of managers having to utilize complex cognition for decision-making, problem-solving, and information management, while behavioral psychology helps to explain some of the biases and heuristics common to management behavior. Modeling SPM processes allows an organization to document and analyze any bottlenecks or inefficiencies in the business' workflow to streamline performance. Process models, similar in structure to cognitive models, are visual representations of the flow of various activities and synchronous exchanges between employees with inversely defined roles within an organization; these will also be examined using the predefined lenses and scope to investigate their relation to cognitive models.

In the remainder of this chapter, the goal will be to find the cohesion between topics shown in Figure 1 by forming research questions that will explain their inter-connections. Figure 2 shows which topics each RQ includes. To combine these topics in a fluid and all-encompassing approach, a systematic literature review (SLR) was completed. Extracted data from comprehensive literature searches were synthesized and analyzed to provide a bridge of understanding between these significant research fields.

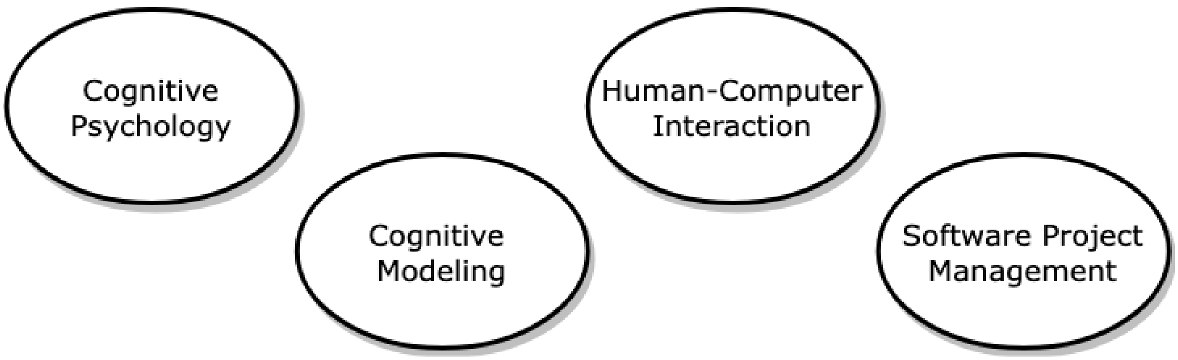

Figure 1. Research domains

1.1 Target Audience

A multi-disciplinary approach will support findings from each area of focus in undiscovered ways. For researchers specifically examining cognition, realizing the cognitive aspects of SPM behavior helps to further develop cognitive theories, and likewise for researchers of SPM. While this study is directed to researchers looking to bridge the knowledge gap between software development, project management (PM), cognitive science, and HCI, it may also have practical implications for business owners and project managers who want to optimize their approach to projects by introducing and considering cognitive science in their business processes.

1.2 Objectives

The following research questions were used to structure and guide thematic concepts of the SLR.

RQ1. Is human cognition reproducible using CM methods?

RQ1.1. What is the scope of cognition that is reproducible?

Representing cognition in a computational and symbolic way allows researchers to test and refine cognitive theories with observable outcomes. To advance the comprehension and guide future research in cognitive science, it is crucial to grasp the extent to which CM techniques encapsulate human cognitive processes.

RQ2. Is HCI behavior replicable using CM methods?

Measuring the precision of models that attempt to explain psychological phenomena requires human behavior as a target variable to predict or assess. While cognitive models focus almost exclusively on cognitive processes, there also exists user models that target individual behavior. It is necessary for researchers to understand the feasibilities for which CM can be incorporated into HCI behavioral models.

RQ3. What are the preconditions of applying CM techniques to SPM?

RQ3.1. What distinguishes SPM from other project management fields?

SPM entails both automatic cognitive processes such as attention, memory, and perception, and complex cognitive processes such as decision-making, problem solving, and knowledge management. Tasks within the SPM field require a specific scope of automatic and complex cognitive processes which has not been the focus of many research studies. It is important to first understand the scope of behavior and cognition that is characteristic of SPM tasks to then synthesize how CM techniques can benefit the completion and analysis of these tasks.

RQ4. What are the thresholds for SPM to be modeled in relation to the preconditions identified by current cognitive science research?

RQ4.1. What elements of preexisting models of project management can be applied to model SPM?

RQ4.2. What are the factors that lead to accurate or inaccurate models of SPM?

Research has suggested that CM techniques can benefit the modeling of a variety of tasks in different domains. Likewise, process and network models have been implemented in project management research, but without the inclusion of cognitive and behavioral preconditions such as biases and heuristics or proposed thresholds of human cognitive abilities. An important metric in any kind of modeling research, apart from the precision and accuracy of models to describe the phenomena in which they are modeling, is the validity and testability of the models developed.

A SLR in the scope of the above topics has the expectation of bridging the gap of knowledge and research that currently exists between SPM and cognitive science, to supply a benchmark of understanding that can be further scaled by supporting research.

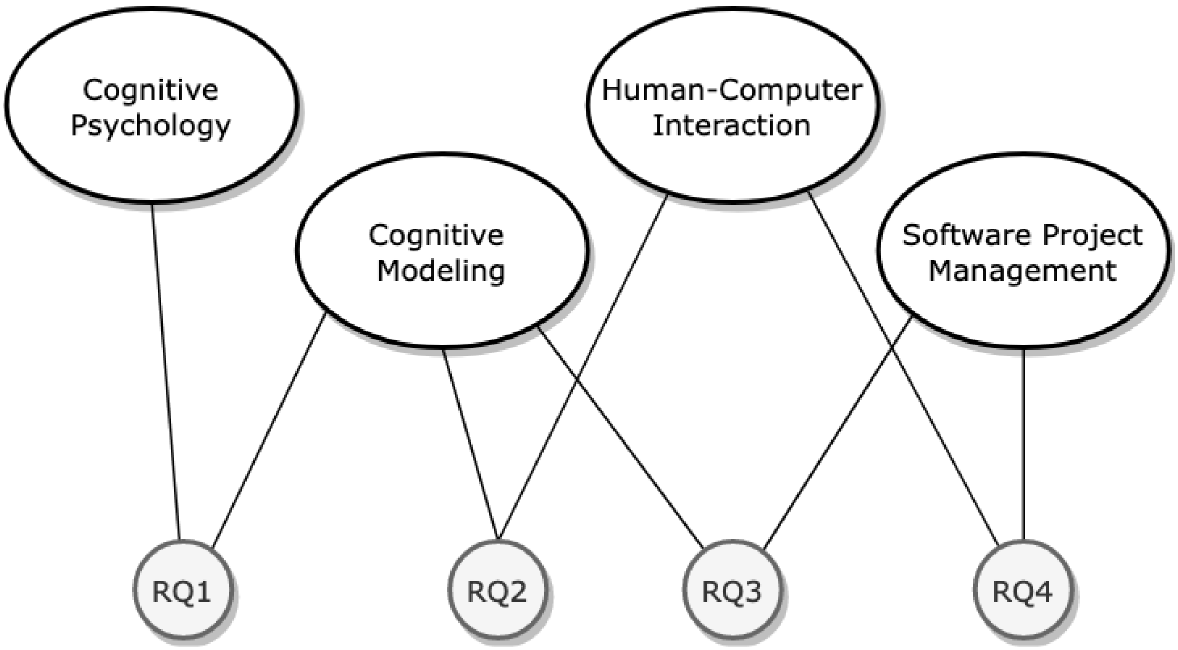

Figure 2. Research domains connected by RQs

1.3 Scope and Constraints

By expanding the research scope of CM into the SPM field, cognitive science researchers have the possibility to study cognitive processes as they appear in the context of software development and potentially gain new insights on biases and behaviors related to the specific area of focus; while researchers in the field of SPM may uncover insights about human cognition that could help recognize underlying processes, motivations, and behaviors. Linking the fields of SPM and cognitive science adds value in each respective area by applying knowledge and conclusions from multiple disciplines into one conceptualized approach. Although each research domain has corresponding themes, it is necessary to refine the overall scope to sharpen the focus of the SLR. Accordingly, there are a few areas that need to be precisely defined.

Cognition

When researching cognition, it is essential to provide a comprehensive review of the various cognitive mechanisms and its errors and variables both. Cognitive science researchers must decide how comprehensively to cover the entirety of the processes responsible for knowledge acquisition, perception, learning and memory, among others, or to apply a narrow-focused scope and examine one cognitive feature exclusively. There are advantages and disadvantages to both approaches: the main advantage of applying research resources to one or a few cognitive aspects is that it provides a thorough examination of the specific set of processes as they relate to the whole cognitive system; however, it must be considered that cognition is a complex and multi-layered entity that controls different features in a modular and dynamic way, making it equally important to understand both the behavioral effects of one cognitive process as it is to understand how the whole of cognition affects behavior.

In this research context, the scope and definition of cognition will be classified as the human mechanisms that control the processes of decision-making, memory formation and recall, learning, attention, and problem-solving. While the extent of understanding cognition is not limited to these processes, they will be the main focuses as they have a direct association with the conceptual knowledge of the SPM field.

Project Management

Considering the limited focused research on SPM in the domain of CM and HCI, at times it may be necessary to compare tasks between different fields of project management, not just in the software sector. Although the specific focus is on SPM, the lack of specified research in this domain and the similarities between cognitive processes in general PM tasks may be identified. This will be reflected in the inclusion/exclusion criteria of the SLR, as many cognitive aspects (such as biases and heuristics) are prevalent in managerial decision-making that do not need to be directly attached to either PM or SPM behavior. Finally, the differences of PM and SPM fields will be further examined in the SLR to add further knowledge to the underlying cognitive processes.

Human-Computer Interaction

While CM has been used to simulate many components of HCI,[1] SPM processes (e.g., visualization of project dashboards) involve the interaction between humans and user interfaces (UIs). In the chosen scope of research, HCI is a critical component to consider – but this study will only the include relevant aspects of HCI to SPM and CM without analyzing the vast collection of HCI research. The main disadvantage of focusing on HCI to study cognitive processes is that much of the individual cognitive processing takes place outside of the computer interaction aspects – that is, in HCI research individuals are often seen as the 'agents' who are interacting with UIs and their behavior is studied accordingly – but this study will approach it from the other way around, focusing on the internal processing that results from *cognitive agents* and their interactions with digital interfaces. Thus, the usefulness of CM as a technique to replicate and measure these behavioral aspects will be sufficiently highlighted.

Affect

SPM processes can be defined with much subjectivity; however, this study does not include observations of emotions or feelings involved in SPM processes or cognitive states, but rather examines the processing time for individuals to carry out these processes. Put another way, the research objectives are limited to studying the automatic behavior necessary for performing SPM processes or the behaviors resulting from the combination of cognitive structures that facilitate individuals to perform SPM processes. Left out of the scope is the coverage of emotions, feelings, beliefs, and mental states that are studied in other sections of psychology.

Other Related Topics

Other topics found frequently throughout the SLR process, that are closely related to the research topics but were intentionally excluded from the selected scope, are as follows.

- Perception, acceptance, or perceived usefulness of PM technology
- Cultural, emotional, or personalities theories in PM
- Perceived project complexity
- Neural bases of cognition
- Implementing cognitive models (e.g., which coding languages used to develop models, building models into network distributions, etc.)
- Affective computing
- Team cognition

[1] (Beimel & Kedmi-Shahar, 2019)

Furthermore, the field of cognitive engineering, along with HCI, is concerned with the design of UIs that best support users within task constraints while focusing on aspects of cognitive psychology that aid in these designs. This study is not as much concerned with designs, or even incorporating cognitive theories into new designs as cognitive engineering posits, but more about the understanding of how to effectively work with tools that are already existing, and how to approach these tools (through modeling) in methods considerate of the effects of machines on human factors, and vice versa. By similar means, studies about design paradigms focused on building IT systems (e.g., systems engineering) were excluded from the full study.

1.4 Assumptions

There are several key assumptions that this study will undertake to establish a comprehensive view of the research topics. First, CM, although it can be in of itself defined by models that reproduce cognitive processes both computationally and symbolically, will include cognitive architectures in its definition. Accordingly, certain research questions (RQ) that reference 'CM techniques' also refer to the use of cognitive architectures as one of its techniques; albeit, cognitive architectures will receive their respective attention in both the Cognitive Modeling and Discussion sections. Secondly, the study of human behavior and performance will be constrained to HCI behavior. This decision is twofold: only observing digital and online SPM activities allows for a more precise focus on the cognitive processes that software project managers enact in activities such as planning, monitoring, and decision-making, and it also allows the research scope to separate these behaviors from the social and environmental variables that affect project managers' cognitive processing.

1.5 Related Studies

While the scope of CM covers many fields, little attention has been supplied specifically to the processes of SPM or have they researched in-depth how CM can be applied to understand and efficiently model the characteristics of SPM processes. Valiente combined software engineering and IT Service Management (ITSM) into UML diagram conceptual models using ontology-based rules as a modeling language;[2] both Case & Stylios and Tlili & Chikhi, in separate studies, used FCM to

[2] (Valiente, 2012)

model project management and SPM risk, respectively;[3,4] Gruhn & Laue studied the effects of cognitive complexity on business process models;[5] Jarecki et al. proposed a framework for building cognitive process models that account for information abstraction with behavioral predictions;[6] Chernova et al. took a cognitive approach to SPM, using cognitive mapping to describe IT project control tasks;[7] and Mair et al. explored the relationship between cognitive processes and personality of software project managers, only to find that modeling problem-solving and estimation is a narrowly-focused task and cannot be generalized to other contexts.[8]

Systematic Reviews

Considering the lack of published research, there is vast potential for a comprehensive SLR that includes the subjects of CM and SPM within its scope. No knowledge of an existing SLR in the precise domain of these defined fields has been found, but there are several formal reviews that include at least one of the topics of interest.

The relationship between cognitive biases and software engineering (SE) have been reviewed in multiple studies: Mohanani et al. performed a mapping study of the systematic errors that are introduced by cognitive biases in SE research;[9] Anu et al. examined how requirements engineering and gathering – a human-centric SE activity – is prone to cognitive error;[10] and Fleischmann et al. reviewed the relation of human cognition and decision-making in information systems (IS) research, observing cognitive biases as the mitigating variable.[11]

Cognition and PM have been linked together in systematic reviews: Fernandes & Vils examined these topics with the intention to further the understanding of cognition;[12] and Stingl conducted a systematic review of behavioral decision making in PM that argued for theoretical homogeneity.

Cognitive Psychology has been included in these reviews as well: Mair et al. completed an SLR on the fundamentals and cognitive processes of problem-solving

[3] (Case & Stylios, 2016)
[4] (Tlili & Chikhi, 2021)
[5] (Gruhn & Laue, 2006)
[6] (Jarecki et al., 2020)
[7] (Chernova et al., 2022)
[8] (Mair et al., 2012)
[9] (Mohanani et al., 2018)
[10] (Anu et al., 2018)
[11] (Fleischmann et al., 2014)
[12] (Fernandes & Vils, 2022)

through the lens of cognitive psychology;[13] Koch et al. examined the affective, behavioral, and cognitive outcomes of agile PM in a SLR and meta-analysis;[14] and Zugal reviewed the state of research between cognitive psychology and Business Process Models (BPMs).[15] While these studies are all useful for developing a benchmark of research, they are lacking a specific focus either on the fields of software, PM, or cognitive science.

1.6 Document Structure

The remainder of this paper will be structured as so: first, the methods and procedures for carrying out the SLR will be detailed in Research Design; then, a background of related topics and their relevance and rationale to be synthesized in the study are labeled in Cognitive Modeling; next, findings from selected studies, data extraction and synthesis, and cluster analysis will be presented in Results; finally, a breakdown of the findings will be discussed in the Discussion, followed by a critical reflection of the study and future research directions in the Conclusion.

---

[13] (Mair et al., 2009)
[14] (Koch et al., 2023)
[15] (Zugal, 2013)

2 Research Design

The study is conducted using an exploratory research strategy. The justification for this approach is due to the research objectives and lack of formal literature connecting the chosen fields. Also, the study takes a post-positivist approach, as most studies that employ computational techniques measure behavioral outputs and cognitive processing time in milliseconds but will critically evaluate the use and understanding of the cognitive approach to modeling SPM.

A SLR is used to collect and combine studies from multiple disciplines. Using a SLR is essential for the chosen subjects as there is no formal review found to support the linkage between fields. The rationale for conducting a systematic review of the connections between these fields is that it establishes a research baseline for future contributions. This SLR is structured in accordance with the guidelines of the 2020 update of the Preferred Reporting Items for Systematic reviews and Meta-Analyses (PRISMA) statement.[16] Following the PRISMA checklist as closely as applicable to this study (p. **Error! Bookmark not defined.**), the remainder of this section will detail the protocol for executing the SLR in terms of planning, conducting, and reporting the review in a methodical and accepted manner for the scientific community.

2.1 Eligibility Criteria

In this study, citations were only accepted within a 30-year period, published from April 1993 to April 2023. The reason for this timeframe for inclusion is because the field of psychology is far older than the current state of technology today; it took a considerable amount of time for the two areas to cohabitate, with the fields not receiving full attention until in the early 1990s with researchers like Kieras and Sun.[17,18] For exceptional cases where originating theories or models needed referenced, they were included. Only studies conducted in English language were included. Only (accepted or submitted) journal articles, conference proceedings, reports, and other grey literature such as theses/dissertations were included. News articles, magazines, blog posts, and similar informal literature was excluded. Furthermore, any sources that were not within the scope of cognitive/behavioral psychology, SPM, or CM were excluded. Only sources that did not fit any exclusion criteria and fit all inclusion criteria were included. Table 1 shows the inclusion and exclusion criteria used.

---

16 (Page et al., 2021)

17 (Kieras & Meyer, 1997)

18 (Sun, 2006)

Table 1. Inclusion/Exclusion criteria

Include	Exclude
Published between April 1993 and April 2023	Published outside search period
Language	Not conducted in English
Source type (article, book, conference, grey literature)	Not an accepted source type
Within topic scope	Outside of topic scope
Full-text available	Full-text not available

2.2 Information Sources

A combination of register, journals, websites, and databases was used to identify studies. Conference proceedings from the International Conference for Cognitive Modeling (ICCM) were thoroughly scanned for the years 2010-2022. Epistemonikos is a database consisting of systematic reviews in a wide range of subjects and was used to scan for relevant systematic reviews.[19] IU Library aggregates several databases and was therefore used to perform a wide search of literature, which covered most of the recognized electronic sources that were identified to be useful to SLRs in the software engineering discipline: [20]

- IEEExplore[21]
- ACM Digital Library[22]
- ScienceDirect[23]

Also included on the list of recommended sources is Google Scholar,[24] which was used in a separate search.

2.3 Search Strategy

To provide a blanket search inclusive of all topics, general keywords were used with wide-scoping Boolean expressions. Filters and updated keywords were then used to narrow the scope of search results to account for subject accuracy within the study. Because of the limited scope of this paper to solely focus on SPM, and to not cover all the different areas of project management – namely, construction, industrial, and healthcare project management – respective subjects to exclude were

[19] https://www.epistemonikos.org
[20] (Brereton et al., 2007)
[21] https://ieeexplore.ieee.org
[22] https://dl.acm.org
[23] https://www.sciencedirect.com
[24] https://scholar.google.com

added to keyword searches to further improve searches. Table 2 clearly shows each information source along with its type, name, and the date on which it was accessed.

Table 2. Information sources

Type	Name	Search Date
Register	IU Library	04/04/2023
Journal	Cognitive Computation	04/04/2023
Journal	Cognition, Technology & Work	04/04/2023
Web	Google Scholar	12/04/2023
Database	Epistemonikos	14/04/2023
Web	ICCM Conference Proceedings	25/04/2023

Search results with astronomical numbers returned were first refined by updating the search string to delimit subjects. For example, to distinguish *software project management* from *construction project management*, the keywords 'construction' and 'industry' were selected to be found within each source's title and full text, and was delimited from the results. Afterwards, filters were applied, such as 'sources', 'publications', and 'subjects'. The full search strings, along with how each search was updated and the number of results returned, are detailed in Table 3.

2.4 Selection Process

Sources were passed through a series of screening processes by one independent reviewer (the author).[25] Certain researchers suggested a 'Single screening with text mining'[26] approach that leverages machine learning (ML) to increase screening accuracy. That approach is used in this paper during the title/abstract screening phase, in a multi-stage process detailed in the remainder of this section.

1. *Data preprocessing*. Search results from multiple sources were cleaned and merged into one file using Python for export into screening tools. Code was written to automatically find and insert missing DOIs and abstracts from each article (see Data Cleaning), while code was adapted from an existing repository[27] and customized to convert formats (see Springer Link CSV to Bibtex and Springer Link CSV to Bibtex Parser).

---

[25] Having an independent researcher screening sources is known to be a 'conservative' approach in terms of cost and labor involved, as there is a tradeoff between the biases of only one decision-maker and the advantages of not having coding disputes that need resolved.

[26] (Shemilt et al., 2016)

[27] https://github.com/0xFORK/csv2bib

2. *Initial screening*. Cleaned search results were uploaded into Abstrackr[28] and screened on title and abstract.[29] A prediction percentage (shown in Figure 22) and a 'hard' prediction of true/false is attached to each citation after sufficient studies have been screened.
3. *Text mining*. The full dataset from Abstrackr was downloaded and added into Orange[30] for analysis. Predictions made by Abstrackr were downloaded in a separate file and joined into the full dataset (Figure 3 and Figure 4). Citations that were screened for relevancy (n=675) were separated into a training set to train a classification model in Orange. Keyword extraction utilized the term frequency-inverse document frequency (TF-IDF) score, a statistic that measures how important a word is to a document. The process and results from this stage are shown in Figure 5 and Figure 24.
4. *Document classification*. Using the extracted keywords, a classification model was built using the k-nearest neighbor (kNN) technique (Figure 6; for model validation scores, see Figure 23). This organized the full dataset into clusters based on word score (presence and count) weighted by the extracted keywords. This step was important for synthesizing correlations in research domains as the analysis that follows highlights key gaps and trends in the relationships between research topics.
5. *Screen on Title & Abstract*. Results from Orange were uploaded to EPPI-Reviewer[31]. Citations that were screened as 'include' were automatically sent to the next phase of screening, while those screened as 'exclude' were excluded from the study. The remaining studies that did not undergo an initial title/abstract screen, but were passed through Abstrackr's prediction algorithm, were accepted. All remaining search results from other sources were uploaded to EPPI-Reviewer and screened.
6. *Screen on Full Text*. The next phase of screening was a two-step process of retrieval and inclusion/exclusion: First, for each citation the full text PDF document had to be retrieved digitally. If the full text was not available to the researcher via institutional access, open-source, or freely-available text download, it was excluded under the exclusion criteria of 'Exclude – full text not available'. Once the full text was retrieved, it was further analyzed to be judged for inclusion/exclusion. Full texts of the studies were analyzed

[28] https://abstrackr.cebm.brown.edu

[29] Abstrackr is a tool that uses a ML algorithm to expedite the screening process by making a prediction on the relevancy of unscreened citations.

[30] https://orangedatamining.com

[31] https://eppi.ioe.ac.uk/EPPIReviewer-Web/home

based on the inclusion/exclusion criteria and either marked for data extraction if included or removed from the full study if excluded.

2.5 Data Extraction Process

Relevant text in each study was assigned to coding tools that were grouped by research question. The process included highlighting the appropriate text and clicking which coding tool/research question it belonged to. No automation tools were used. Data was assigned to coding tools for the purpose of further synthesis and classification. For this purpose, other variables that were not exclusive to research questions, but to the SLR, were created and assigned. These variables include:

- *Extracted*. Studies that were fully passed through the data extraction phase and required no further analysis.
- *Cross-referenced*. To mark that a study's references have been fully examined.
- *Reference framework.* To mark a study as one that provides a referenceable framework for building conceptual models.
- *Cross-reference*. To mark individual text passages relevant to the SLR requiring a citation check.
- *Research needs.* To document mentions of further research requirements.
- *Main point*. To highlight the main findings of each study.

If a study was given the code 'Cross reference' (either pointing to a specific reference or to the general study), the next step was to perform backward snowball sampling to cross-check all relevant references that may not have been found in the initial searches.[32] This approach included sub-processes of scanning each marked study's reference list for title and journal, locating and screening on full text, and applying inclusion/exclusion criteria based on study scope.

2.6 Synthesis Methods

All studies coded in data extraction were exported as reports into Excel. Each row of the report included all extracted data from each study, while the columns organized all data by research question. Synthesis was performed using pivot tables and visualizations inside Excel. Also, EPPI-Reviewer's reporting function includes a 'quick question report' that allows narrowly focused reports to be formed based on ad hoc analysis intentions of the reviewer; this function was used to synthesize results by forming shorter reports of multiple research questions for further analysis.

[32] (Wohlin, 2014)

To further explain this process, an example is where a report is formed with RQ1 and RQ2 selected, and only the studies with extracted data in those two categories would be included in the report, allowing the reviewer to more efficiently synthesize research that covers only RQ1 and RQ2.

Text mining of title and abstract was performed after data extraction using the Lingo3G[33] clustering API included with EPPI-Reviewer. Clustering documents after extraction as opposed to before allowed for more efficient cross-tabulation of included studies and clusters that more accurately defined the research scope. Clusters were visualized in tables and graphs to aid synthesis.

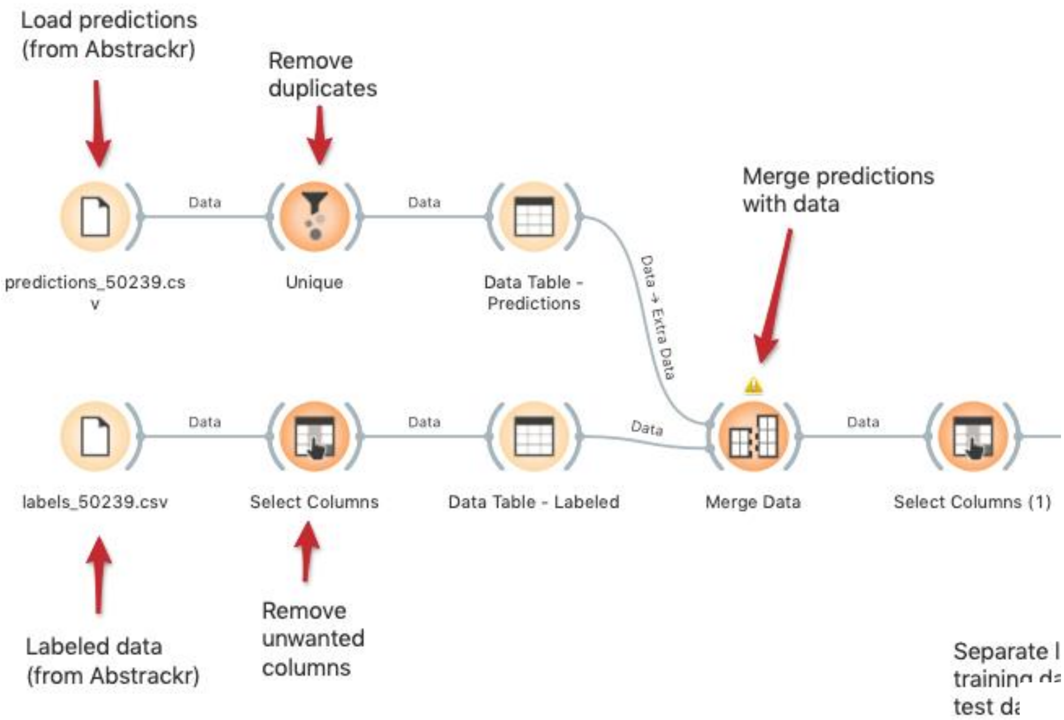

Figure 3. Dataset building process in Orange (a)

[33] https://www.carrotsearch.com/lingo3g/

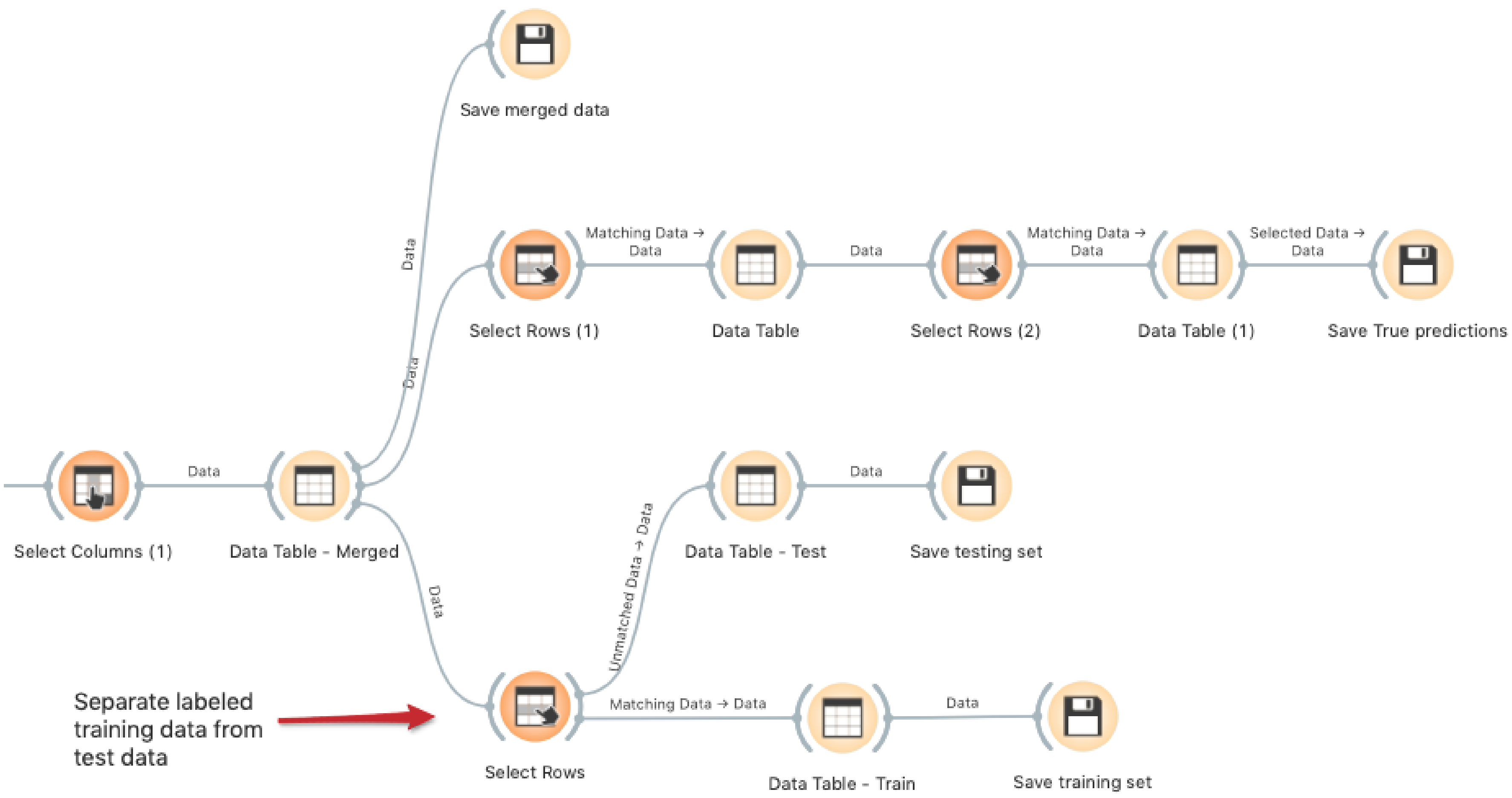

Figure 4. Dataset building process in Orange (b)

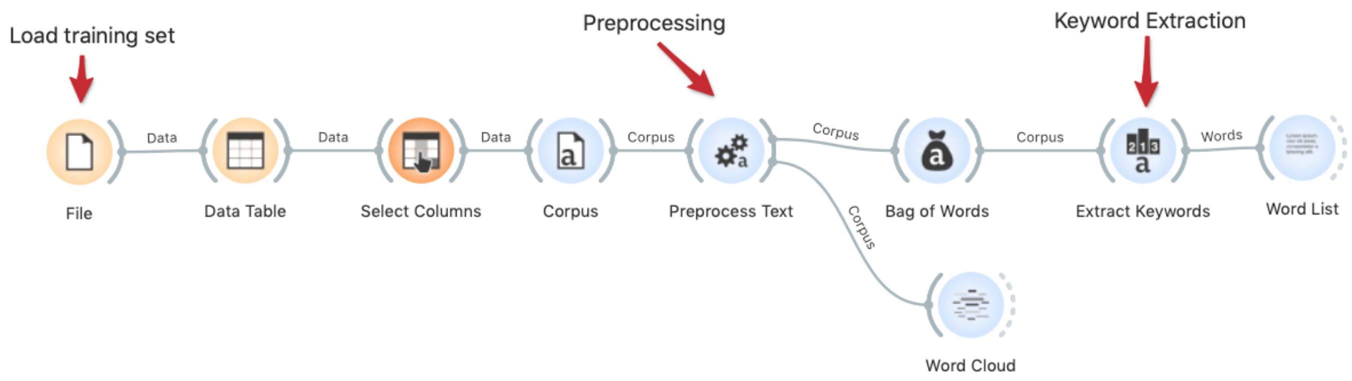

Figure 5. Keyword extraction process in Orange

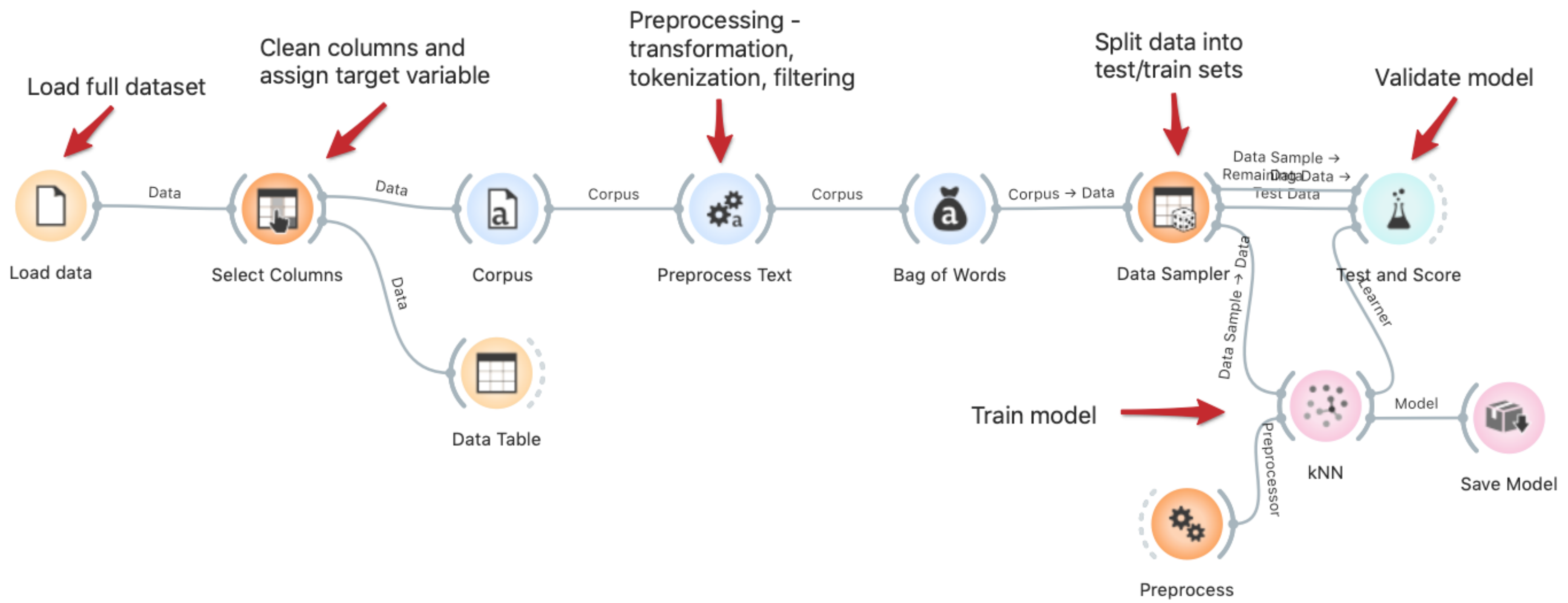

Figure 6. Model building process in Orange

3 Cognitive Modeling

Cognitive models explain different cognitive processes along with their relationships and exchanges between one another in a symbolic, replicable, and visual way. CM has been used in research to model a variety of tasks from a wide range of disciplines, and to test and formulate new hypotheses on behavioral outputs of the subjects being researched. CM is especially significant in research settings due to its capacity to provide clarity on the psychological plausibility of cognitive theories.[34]

While cognitive models can vary in definition from overly symbolic and merely representative to mathematical, researchers apply CM to test and increment knowledge in cognitive science using *computational models*, which in this study will be defined as cognitive models reduced to algorithmic components. The evaluation process of cognitive theories using computational modeling can be seen at a high level in Figure 7.

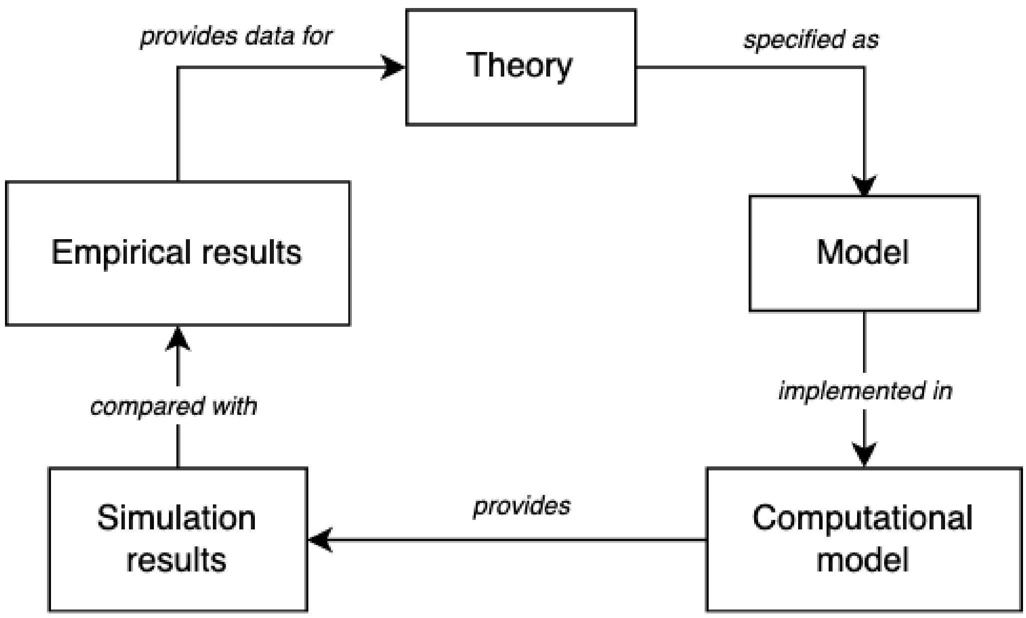

Figure 7. A conceptual scope for computational model development.[35] First, theories are formally developed into models that describe its theoretical boundaries; then, models are made replicable by reducing them to computations and algorithmic components; next, the computational models are used to simulate human behavior and the results of simulations are compared with empirical observations; finally, the results are used to further refine the theory, and the cycle begins anew.

Although there is much overlap in the definitions and applications of *computational* and *cognitive* models, the same considerations are taken in their development. The

[34] (Gigerenzer & Brighton, 2009, p. 128)

[35] (reproduced from Emond & West, 2003, p. 530)

conceptual scope of cognitive model development highlights the considerations that researchers should take when developing cognitive models. Figure 8 provides a visual overview of these considerations, separating the decision domains into *information*, *mental events*, and *behavior*. The information domain is where the researcher specifies the scope of the model by deciding which real-world phenomena the model should cover. The mental events domain is the intermediate stage where the model's inputs are transformed; where the researchers should gauge and refine the model based on its compatibility, separability, and testability. The behavior domain is where the model's outputs should be evaluated for predictive accuracy.

A similar conceptual framework for cognitive model development (Figure 9) shows how the previous scope can fit inside the *domain suitability* group. In this framework, Schürmann shows the interplay of the cost-benefit decisions researchers must make after domain analysis, and how this evaluation process is iterative between research questions, resources, and model design.[36] To fully encapsulate cognitive processes into conceptual, computational, and cognitive models, it is necessary to first provide a background knowledge of cognition to understand the scope of cognitive theories that can be modeled.

3.1 Cognition

The various cognitive processes focused on in this study have, at times, distinct approaches for both modeling and theorizing; however, let's consider two analogous perspectives of cognition to provide a high-level understanding.

First, Simon comprehensibly illuminated cognition in his analogy of an ant walking on a beach: a person perceiving an ant taking a complex path across a beach may attribute the complexity of the path taken by the ant to the intelligent behavior of the ant, when in fact the complexity only came from the terrain which the ant was traversing.[37] Human cognition can then be thought to have similar associations as the ant has with its environment; a makeup of simple internal mechanisms that respond to complex stimuli. Simon's analogy illustrates how complex behavior, whether in humans or ants, is intricately related to the complexity of their cognitive knowledge.

Second, Card et al. describe an analogy of the human mind as an 'information-processing system'.[38] Here, it is conceivable to think as a computer engineer, who

[36] (Schürmann & Beckerle, 2020)

[37] (Simon, 2019, p. 64)

[38] (Card et al., 1983, p. 24)

would describe a computer – at the system level as opposed to component level – to be the interconnection of memory, processors, and their parameters. Thinking of cognition in this same way allows one to envision cognition as a complete system. In this perspective, cognition is the collection of processors that manage different mechanisms like memory, perception, and attention; where the input is information, and the output is complex behavior.

In this light, in the following section each main cognitive process included in the study will be discussed so a formal definition of each can be established. These cognitive components are then connected in a modular manner, akin to an information-processor, to define cognition as one all-encompassing system of simple parts that outputs complex behavior, as Simon's ants.

Decision-Making

The highly dynamic process of decision-making considers that decisions are not binary events but invariably affected by incoming information. In this understanding, decisions can be influenced by individual feedback from stimuli in ones' visual receptive field and task environment to produce behavior that requires adaptation to conditions or events. Such feedback on a cognitive level requires actions to be taken to achieve a goal, where subsequent decisions are made in response to prior actions or events; all under constraints of time and fluctuating environments that do not allow for prolonged mental processing before deciding.[39]

Researchers have identified two disparate modes of decision-making: *intuitive* and *analytical*.[40] Intuitive decisions are made based on imbedded contextual pattern recognition, and analytical decisions are those that generally abide by symbolic rules. Analytical decision-making is therefore more measurable in an academic context (using computational models), as behavior and task performance can be evaluated and recorded while simultaneously modifying the symbolic rules that make-up the decisions. Intuitive decisions are highly complex to study, as it is challenging to artificially reproduce human pattern recognition of situations, which turns the cognitive process into a 'black box' where each module cannot be properly understood.[40]

---

[39] (Prezenski et al., 2017, p. 2)

[40] (Kennedy & Patterson, 2012, pp. 15–16)

Memory

The processes of memory and learning are directly intertwined and can be thought to include perception as a vital component. Memories are mostly induced as bottom-up perceptual processes;[41] however, at a low-level, information gets stored into long-term memory after these perceptual processes, and is recalled from working memory in schema referred to as 'chunks' that reduce the mental effort involved in retrieval.[42] Aside from working and long-term memory, researchers define 'external memory'[43] as a process of memory storage outside of cognitive processes (i.e., information written on a document such as a blackboard or sticky note), which will be an important concept to note as they relate not only to process models but to specific SPM behaviors.[44]

Memory can also be divided into two different systems, *declarative* and *non-declarative*.[45] Declarative memory entails consciously recalling facts and events, while non-declarative (or procedural) memory entails the dynamic extraction of elements and patterns from detached events, also supportive of the development of skill-based motor abilities.[46] This type of processing can be encapsulated in the instance theory of automatization that details the automatic processing of information after routine practice of behavior in a consistent environment, where cognitive mechanisms called 'instances' represent separate units of memory retrieval.[47]

Attention

Attention is a cognitive process affected by individuals' evaluation of the visual world. It is a vital human cognitive process because it filters incoming sensory information based on relevance.[48] Visual cognition limits the volume of information and rate at which it can be accessed from visual short-term memory, therefore making attention a capacity-limited process.[49]

---

[41] (Laird et al., 2010, p. 4)
[42] (Newell, 1994)
[43] (Sweller, 1988)
[44] (Zugal et al., 2011)
[45] (Squire, 2004, p. 171)
[46] (Patterson et al., 2013, p. 333)
[47] (Logan, 1988)
[48] (Kotseruba & Tsotsos, 2020, p. 37)
[49] (Marois, 2005, pg. 296)

In neuroscience, attention is studied with eye-tracking technology and attentional models.[50] On a neurocomputational level, visual attention is a facilitation of the coordinated retinal and foveal movements that elicit an extraction of background and foreground objects from individuals' direct and peripheral perceptive fields.[51]

One study saw researchers observe how information processing involved in browsing behavior involves a user's attention to every meaningful area of the screen in a minimal amount of processing time, followed by the decision to move to another area depending how their goals are attained.[52] In the context of SPM, individuals' attention processes when scanning websites and interacting with technology (i.e., mobile and web applications) is especially relevant.

Problem-Solving

Problem-solving refers to the interactive processes of memory, perception, and attention to find solutions and achieve goals. The act of solving problems starts with identification in the initial state, where actions are then taken to reach an intended objective or goal state, and further actions to reach the solution are taken to complete the goal.[53] The space of all possible actions that can be taken combined with the information processing boundaries of the problem-solver describe both well and ill-defined problems.[54]

The three main processes of problem-solving identified by cognitive psychology are *inference*, *search*, and *recognition*, with the latter two mainly being used in object or pattern recognition and other problems of low complexity.[55] While many conceptual models can exceed the complexity of problems that search and recognition cover,[56] human cognition seems to have developed critical shortcuts in solving these problems that minimize the effort involved (see: Biases, Heuristics). Due to the unbounded variety of strategies, the flexibility of the human cognitive system to use different complex algorithms and heuristics, and due to the multitude of tasks and solutions that make-up any problem space, [57] deciding which rules to use to model problem-solving computationally is a challenging task.

[50] (Borji & Itti, 2012)
[51] (Borji & Itti, 2012, p. 185)
[52] (Chanceaux, 2014, pg. 1)
[53] (Mair et al., 2009, pg. 1)
[54] (Hayes, 1978)
[55] (Larkin & Simon, 1987)
[56] (Zugal et al., 2011, pg. 2)
[57] (Howes et al., 2009, p. 717)

Complex Cognition

When thinking in terms of the complexity of cognitive processes, complex cognition relates those mental activities that interact and symbiotically make-up higher-order processes such as problem-solving and decision-making.[58] Mental activities that are almost automatic, like perception and attention, can be seen as basic cognitive processes that fundamentally create higher complex cognitive structures. For example, backcountry skiing requires dynamic decision-making based on mental models that respond to feedback from responses to stimuli in a skier's environment.[59]

In this way, complex cognition can be defined simply as the mental processing when information is being derived from other information. Many tasks in SPM involve complex cognition to complete, as project managers are constantly making decisions about resource allocation, analyzing, or strategizing based on available information from project databases.

3.1.1 Existing Theories

Theories of cognition hypothesize how knowledge is stored, accessed, and used in the higher functioning of problem-solving and decision-making to try at 'reverse engineering the mind'[60]. The types of models that exist range from probabilistic to rational, and from bottom-up (connectionist) to top-down (probabilistic), while the main theories describe cognition in multiple ways of conscious versus unconscious decision-making behavior.

Dual Process Theory

An important theory of information processing, reasoning, and decision-making comes from Kahneman's interpretation of the fundamental Dual Process Theory, which states there are two systems of thinking (S-I & S-II) that undertake dissimilar actions to control thought processes in decision-making: S-I is the intuitive system that acts unconsciously, effortless, quickly, and emotionally, while S-II is the reasoning system that acts slow, consciously, and rationally.[61,62] Because human cognitive processes are traditionally thought to be rational by nature, it is common to believe that by modeling rational S-II processes researchers can come to an approximate computation of reasoning and decision-making.

[58] (Knauff & Wolf, 2010)
[59] (Prezenski et al., 2017, p. 1)
[60] (Griffiths et al., 2010)
[61] (Kahneman, 2017)
[62] (Almomani et al., 2021, p. 1)

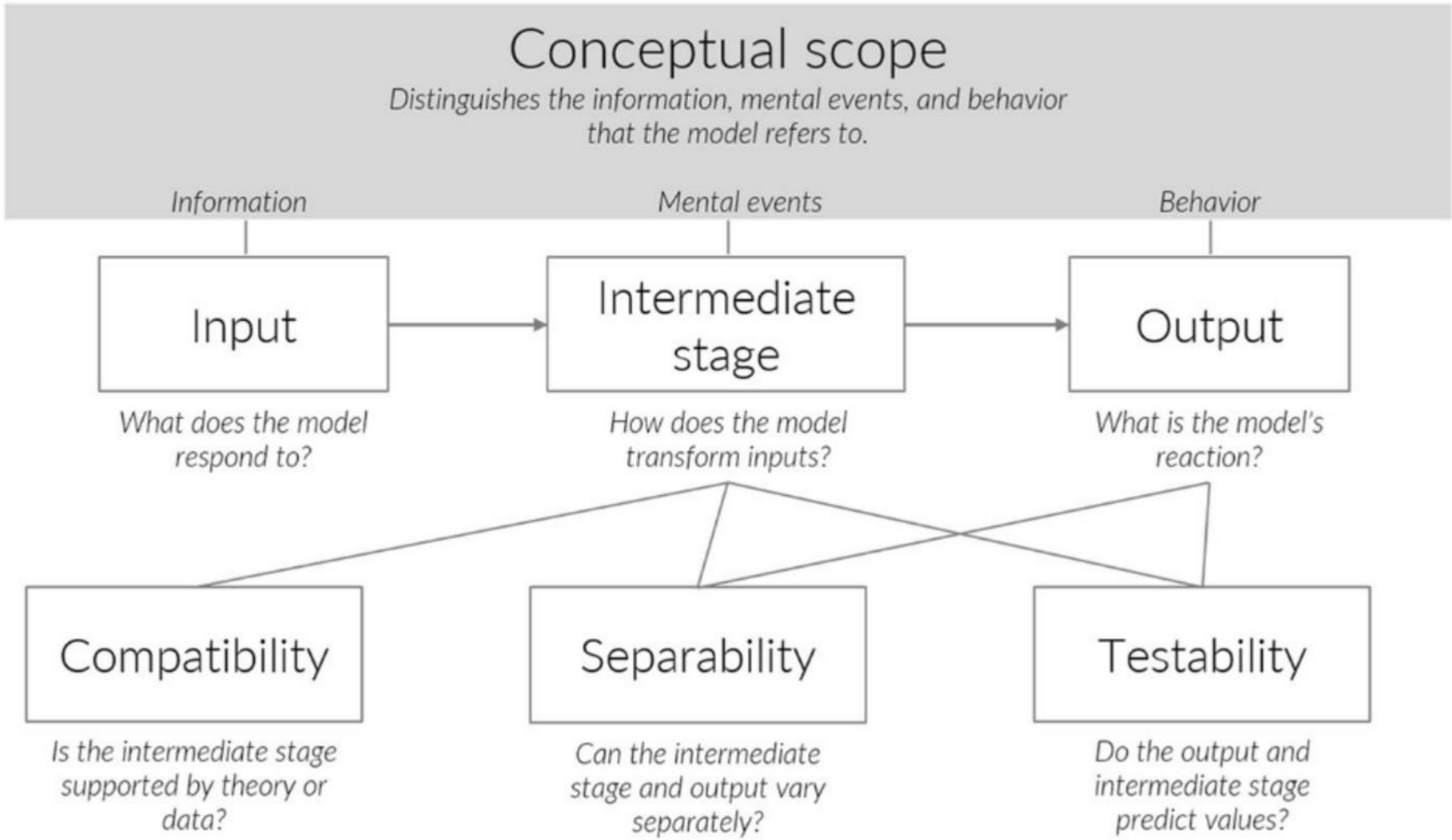

Figure 8. A conceptual scope for cognitive model development.[6]

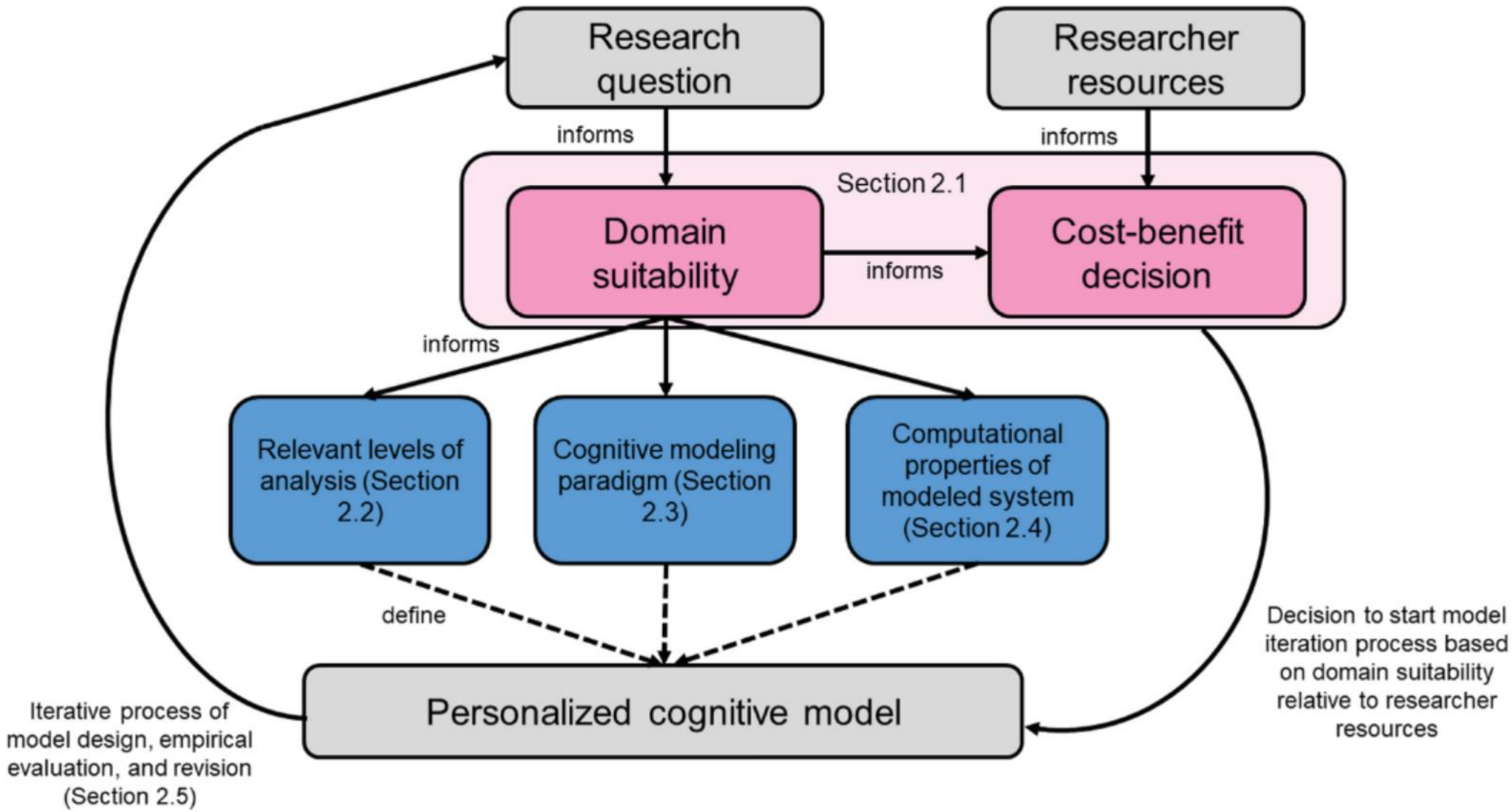

Figure 9. A conceptual scope with domain suitability and cost-benefit decisions.[63]

[63] (Schürmann & Beckerle, 2020, p. 2)

Rational Analysis

According to the theory of rational analysis, knowledge is accessible based on its probability of being utilized in a specific context.[64] The immediate assumption defined in economics is that humans have unlimited computational resources and can generate the most optimal decisions in any context, the so-called 'rational person'[65]; but it is more correct to say that decisions are made after determining them sufficient based on a characterization of the environment. This process, called satisficing, is in line with the important theory of Bounded Rationality[66], where human decision-making is constrained by limited knowledge and various psychological aspects – analogously known as the 'dancer' (bounded rationality) and the 'ballroom' (task environment).[67] Bounded rationality is a key theoretical framework to consider in the research of decision-making in SPM tasks, where the software project manager is the dancer and the software project environment is the ballroom.

Expected Utility Theory (EUT) explains how rational decisions can be computed, having roots in behavioral economics. In EUT, all possible outcomes from a particular decision are averaged and the resulting score is used to weigh the probabilities of utilities;[68] finding usefulness in application such as network models which aim to find the optimal path from start to finish.

Marr's Levels of Analysis

While rational analysis is comprehensive of the whole cognitive system, the continual work of cognitive science is aimed at bridging the levels of analysis between the rational and computational levels to model how decisions are made, not just because of psychological processing but of abstract computational processes as well.

According to Marr, cognitive processes can be modeled on three different levels: the *computational* level serves information-processing as a top-down method of solving problems using abstract statistical calculations (why we do things); the *algorithmic* level explains the cognitive processes concerned with representing and manipulating information (how we do things); and the *implementation* level explains how the internal algorithms can be represented physically.[69] While there is an established consensus of the usefulness of these different levels, it is often the case for cognitive scientists where establishing a connection between the levels is difficult.

---

[64] (Anderson, 1996)

[65] (Fu & Pirolli, 2007, p. 361)

[66] (Simon, 1957)

[67] (Fitts, 1954, p. 391)

[68] (Kutsch & Hall, 2005)

[69] (Marr, 1982)

3.1.2 Biases, Heuristics, & Cognition

Cognition can be fundamentally biased. Cognitive biases exist when there are incongruities between a correct answer and an individual's answer, as determined by normative rules.[70] For example, let's say complexity comes from the ability and need to anticipate future events: hindsight bias, which gives individuals the belief that future events are more predictable than they really are, occurs because not all future events are related to past events.[71] In other words, foresight may indeed be correlated to hindsight, but it does not signify a causal link.

While cognitive biases can be generalized to many different contexts and tasks, the intent for this study is to only focus on the biases that are prevalent in the SPM field and thus can be adapted using CM, being anchoring and adjustment, order effects, availability bias, and planning fallacy. Anchoring and adjustment happens when choices are made under uncertainty and involve numerical outputs, as individuals tend to anchor to a number and adjust their choice depending on the distance from the initial anchor.[72] Order effects describes how the order in which information is received affects the overall judgement of the information.[73] Availability bias explains how the easiest information to recall cognitively provides an overconfidence of that information being correct.[74] Planning fallacy is a subcategory of optimism bias, which describes the tendency to underestimate task completion times.[75]

Heuristics are another cognitive phenomenon that are like biases in that they can induce errors into judgements and help to make quick decisions; however, it is best to think of heuristics as 'mental shortcuts' for solving problems with limited mental resources, because ultimately, they allow for the robustness of cognitive capacity.[76] Understanding how to model both biases and heuristics is important for applying theoretical assumptions to conceptual models.

3.1.3 Cognitive Models

Constraints exist when trying to match complex quantitative data to a model, as the structure of the task to be modeled along with the rigidity of the architecture adds

[70] (Montibeller & Von Winterfeldt, 2015, p. 1)
[71] (Cunha & Moura, 2015, p. 2)
[72] (Aranda & Easterbrook, 2005, p. 346)
[73] (Griffiths et al., 2012, p. 268)
[74] (Flyvbjerg, 2021, p. 540)
[75] (Kahneman & Tversky, 1979, p. 315)
[76] (Gigerenzer & Brighton, 2009, p. 109)

to its complexity, and the fundamental goal of CM is to model and predict the interplay of multiple complex systems in a simplified way.[77] CM is a powerful analytical technique that can identify and reproduce cognitive phenomena such as heuristics and context effects;[76,78] but in reality, to what extent of our understanding of cognition can be reproduced in a computational way is still up for debate.

In one way of thinking, the purposefulness of CM is mainly hypothetical; that is, to test and modify hypotheses of both separate and interconnected objects based on cognitive theories. In this way, a qualitative approach of analysis is taken to understand how cognitive systems are at work and the relation they have to other objects inside the whole cognitive system. To effectively model parallel objects of cognition, different theoretical approaches can be taken.

Rational, Probabilistic, & Connectionist Models

Just as there are different theoretical branches of psychology, there are diverse opinions on how to correctly model cognition. **Probabilistic Models** aim to deduce psychological or neural processes from top-down problems, but **Connectionist Models** are first examining the problem space that cognitive processing can solve before applying them to external problems.[79] Probabilistic models therefore make it easy to study the effects of different assumptions implemented on different tasks, to study biases and heuristics, and explain how humans generalize differently in different situations;[80] while connectionist models are more efficient at studying a specific stream of knowledge in explicitly defined processes.

Rational Models of cognition cover processes of thought,[81] memory,[82] and generalization that assumes an optimal choice is being made. These models are a key to bridging Marr's levels of analysis (computational and algorithmic) with the insights provided by probabilistic models.

The steps in creating rational process models are: (1) define an algorithm that can be probabilistically inferred; (2) determine whether the algorithm's components properly define the understanding of cognitive processes; (3) determine how well the model fits to certain behaviors. Rational process models could be a powerful resource when examining decision-making processes in SPM, as they are a way of

---

[77] (Chernova et al., 2022, p. 428)
[78] (Lee et al., 2019, p. 141)
[79] (Griffiths et al., 2010, p. 359)
[80] (McClelland et al., 2010, p. 353)
[81] (Sanborn et al., 2010, p. 1144)
[82] (Anderson, 1990)

combining modeling approaches in the context of information-processing constraints. To combine modeling approaches, researchers are required to provide precise conclusions on vague theories, which requires the act of formally modeling these theories to improve their precision. One way to do this is through the development of cognitive architectures.

3.1.4 Cognitive Architectures

Cognitive architectures act as proposals to existing theoretical frameworks. Representing cognitive theories as modular structures makes them easier to study and test and helps add to the wide body of research in cognitive science. They are the structures that cognitive models are built on top of and share the same constraints;[42] examining human behavior in terms of the interaction between many cognitive processes and the underlying mechanisms that make it up. Cognitive architectures have three main systems: *memories*, *processing units*, and *language*.[83] Memory systems act as storage for knowledge, processing units are the mechanisms for storing, selecting, and accessing knowledge, and language systems manage the representation of stored knowledge.

There is a clear distinction between mathematical models that implement cognitive processes (such as neural networks) and cognitive models built on cognitive architectures, in that the latter are directed at forming cognitive interpretations of behavioral processes as a form of abstraction of the underlying mechanisms.[84] In over 40 years of research and development of cognitive architectures, many have been identified and each has a specific range of cognitive theories that they include.[85] Due to the copious number of cognitive architectures in existence and the wide set of cognitive phenomena they reproduce, only those that are most prevalent in research and applicable to the field of SPM research will be covered in detail throughout this study.

ACT-R

Anderson first articulated about the structure of cognition and its complexity to model, that, "The whole is no more than the sum of its parts, but it has a lot of parts."[64] The cognitive architecture with the most widely accepted theoretical base, ACT-R, was developed specifically to model and fine-tune the learning and processing of knowledge from disparate parts, both numerous and abstract.[64] Originating from the idea of modeling how humans write recursive algorithms, ACT-R is

---

[83] (Laird, 2008, p. 2)

[84] (Prezenski et al., 2017, p. 4)

[85] (Kotseruba & Tsotsos, 2020)

made up of production rules that control how the theoretical parts interact. The main advantage that ACT-R has over other architectures is the incorporation of both long and short-term memory processors. Specifically, the way that the system deals with the processes of memory, categorization, and problem-solving is unmatched, which is why many researchers have used it as a baseline architecture to build individual theories on top of.

Soar

Another cognitive architecture that allows for a broad scope in its constraints on cognition, Soar, permits a similar level of abstraction of cognitive theories as ACT-R.[83] The specific advantage that Soar has over some architectures is its capability to generate new rules on top of the procedural rules placed by the modeler, which is a core function of complex cognition.[86] The Soar architecture generates behavior as a series of movements through a problem space, with operators that define different goal states;[87] where decisions are made in the architecture by it selecting the next operators to deploy. Unlike ACT-R, knowledge representation in Soar is *symbolic* instead of *hybrid*; memory processing is graph-based instead of connectionist; and learning mechanisms process analytically instead of from top-down or bottom-up flows.[88]

3.2 Cognitive Modeling & Behavioral Science

Where CM is a bottom-up technique of understanding the underlying interplay of cognitive processes, human behavior includes a multitude of phenomena that can be studied in a top-down way due to the observable nature of human performance. The intersection between cognition and observable behaviors forms the basis for various interdisciplinary fields, most notably in HCI, where designing seamless and intuitive UIs requires a fundamental understanding of how cognitive mechanisms shape user behavior. Because one of the principal goals of CM is to predict and study the behaviors that result from the influence of cognitive processes, a single theory of cognition represented in model form can predict not just one behavior but all behaviors within a derived space.[89]

Other than HCI research adding to the current knowledge base of human behavior, fields like behavioral economics have been pivotal in the study of behavioral science

---

[86] (Liu, 2009, p. 578)

[87] (Newell & Simon, 1972)

[88] (Ritter et al., 2019)

[89] (Howes et al., 2009, p. 722)

by predicting human behavior using rational models;[90] although behavioral economists are quicker to point out cognitive over behavioral biases when studying behavior using these models. Principles of behavioral economics can help understand the effects of biases on SPM decision-making by examining the underlying processes and how they relate to errors in judgement and dual processing.[91] Researchers with this idea have identified 10 of the top behavioral biases in project management: *anchoring and adjustment*, *order effects*, *availability bias*, and *planning fallacy* are among the list of most relevant biases to consider when conceptualizing SPM processes. These insights are crucial components of cognitive architectures that aim to capture the complex relationship between cognition, biases, and decision-making behavior.

One important way to replicate cognitive theories into testable behavioral components is to separate the individual systems and observe their interaction. Alongside CM techniques, engineers of behavioral models also build based off solidified theoretical frameworks and must incorporate the cognitive processes underlying human behavior into architectures to properly study the resulting behavior.[92] Human performance is the focus of such models, especially in relation to environmental variables affecting the task.[93] Human performance models can then be thought of as quantitative simulations of behavior enacted within the scope of a specific task environment and have a wide application of research in fields such as military operations[94] and HCI.[95]

User models in HCI are typically constructed to replicate human behavior in the interaction and design of UIs. CM has been pivotal in the understanding of HCI behavior by providing models and architectures that incorporate the underlying processes that motivate behavior such as typing on a keyboard, moving a mouse, visual search and attention, and physically selecting elements on a screen.[19]

MHP

The benchmark method of incorporating cognition into behavioral theories is using the Model Human Processor[38] (MHP) framework. Again, seeing the mind as an information-processor, the MHP divides into three separate modules: *perceptual*, *cognitive*, and *motor*. Each system has a devoted processor, and different tasks require

---

[90] (Almomani et al., 2021, p. 1)

[91] (Flyvbjerg, 2021, p. 531)

[92] (John & Kieras, 1994)

[93] (Bagherzadehkhorasani & Tehranchi, 2022)

[94] (Kotseruba & Tsotsos, 2020, p. 37)

[95] (Ocak & Cagiltay, 2017)

humans to enact different levels of control over these systems. The perceptual system transfers sensory information from the physical world into internal representations. The cognitive system combines coded information from its working memory and previously stored information from its long-term memory to make decisions, while the motor system is responsible for carrying out the processed requests.

The way that individual processors in the MHP model interact is governed by nine basic principles (see MHP Principles of Operation for the full list). For example, the *Variable Perceptual Processor Rate Principle* (τ_p) states that the rate at which the Perceptual Processor cycles (τ) (responds to stimuli) is determined by the intensity of a stimulus (p). In addition, the *Principle of Rationality* states that the sum of *Goals, Task, Operators, Inputs, Knowledge,* and *Process Limits* equates to *Behavior.* This guiding principle is a vital display of how bounded rationality is implemented within cognitive architectures and behavioral models using MHP.

GOMS & KLM

The most significant model of human behavior observes cognitive functioning as a group of Goals, Operators, Methods, and Selection rules (GOMS). The GOMS model uses the MHP as its core methodological approach. A typical GOMS model aims to replicate one task, i.e., editing a document, and will include a set of goals, operators, methods, and selection rules that assist in the completion of the task. The GOMS model can be targeted at different analysis levels of user behavior (Functional, Argument, Unit-Task, and Keystroke).[96] The Keystroke-Level Model (KLM) is a widely used subset of a GOMS model that targets the motor and cognitive processing in making keystroke-level actions (such as moving a mouse or typing on a keyboard). This model is especially useful in fields such as HCI, where the interval of time – between physical movements and both cognitive and perceptual processing – is measured in humans by observing their interactions with digital tools.

EPIC

Another architecture where the specific goal of measuring human performance takes a similar approach. EPIC was developed for the specific purpose of including human perceptual and motor controls into cognitive theories, so it reasons that the motor system needs to be accounted for instead of purely the cognitive aspects of cognition.[17] For example, the visual system relies on inputs from users' retina and fovea and thus depends on the motor coordination of different parts to create a perceptual understanding of an environment. Furthermore, the key deliberation of EPIC is that human performance is controlled equally by cognitive processes and

[96] (Card et al., 1983, p. 259)

motor and perceptual mechanisms. Considering this, the various cognitive processors of EPIC are surrounded and therefore bounded by motor skills. Due to the interplay of processors and both motor and perceptual mechanisms of EPIC, measuring the metric of time is critically important when computationally simulating human performance.

3.3 Cognitive Modeling & Software Project Management

The field of SPM is especially suited for CM as the tasks that comprise the role of a software project manager are dependent on cognitive architectures for decision-making, problem-solving, and memory – complex internal processes that are difficult to objectively define yet produce patterns of observable behavior inseparable from theory. While phases of projects (such as feasibility and design) may be implemented in similar ways throughout software projects as they are in construction projects, the difference is in the complexity of IT products and tasks.[97] It is in the very lifecycle of software development (SDLC) that gives SPM its specific focus towards the delivery of software products.[98] Therefore, it is vital to first examine how traditional PM models function so it can then be applied to SPM.

3.3.1 Existing Models of PM

The methods, domains, and artifacts that makeup the lifecycle of projects is fundamentally defined in knowledge base of the Project Management Body of Knowledge[99] (PMBOK). The main 'model' of PM is the PMBOK, as it standardizes and provides credentials for industry-standard practices and tasks. In PM, the assembly of tasks within a project create a network of interconnected activities. Tasks are known to be the fundamental building blocks of projects.[100] Tasks are often connected for the purpose of fulfilling business goals, where sometimes the connections between activities are stochastic and other times deterministic.

Tavares took a wide-scoping view in their research, simply questioning how a project can be modeled using a composition of tasks.[101] The model stems from an important concept in PM carried over from operational research (OR): activities within a project collectively add up to achieving a goal (project delivery) when completed. In this method, any project can be modeled using a construction of different sets:

---

[97] (Andrade et al., 2015)
[98] (Nitin & Saini, 2022, p. 1)
[99] (Project Management Institute, 2017)
[100] (Heloisa & Tseng, 1996, p. 373)
[101] (Tavares, 2002, p. 2)

activities (tasks), *precedence conditions* (activities that need to be completed before another task begins), *attributes* (each activity's properties in relation to PM processes), and *criteria* (factors that influence project managers' decisions, such as total duration or perceived risk).[102] Researchers here referenced a basic network model of project activities that follows a stochastic modeling method (Figure 10). This style of model represents a direct graph model that has a clear starting and ending point, where connections between nodes and arcs represent different activities and precedence conditions necessary to complete a project.

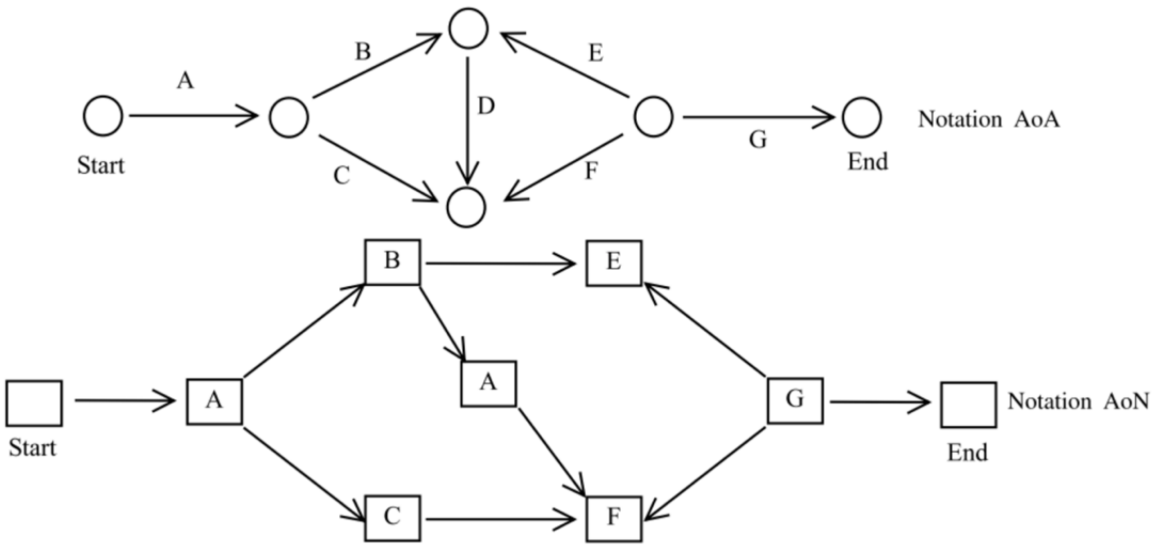

Figure 10. Two example network models of PM.[101] Activity-on-arc (top) positions activities on the connecting arcs while the nodes represent the completion of activities. Activity-on-node (bottom) positions activities on the nodes, while the connecting arcs represent precedence conditions for completing the activities.

Discrete Event models describe a discrete set of tasks that makeup the behavior of a system.[103] The tasks in this type of model are performed sequentially, where time increments only when events take place, and the completion of an entire simulated event passes through process blocks that define specified sets of human behavior.

System Dynamics models are used to simulate the architectural components of complex systems.[104] The events inside system dynamics models have cause-effect relationships with the surrounding events, which allows the simulation of feedback within the modeled system (see Figure 13). The ability to replicate the interplay be-

---

[102] (Battersby & Carruthers, 1967, p. 469)
[103] (Lakey, 2003)
[104] (Forrester, 1997)

tween events is an important technique to consider for modeling SPM, as the feedback between events allows a deeper analysis of the cognitive and behavioral dynamics between tasks in a project environment.

In an alternative method, Snider & Nissen use activity vectors to model how knowledge flows within projects.[105] Using software development projects as an example, they describe two different flows, horizontal and vertical, that give contrasting perspectives of the knowledge gained through the linear sequence of requirements (R), architectural design (A), software design (S), coding (C), and test (T). The horizontal flow of vectors (R-A-D-C-T) is highly explicit and follows formal project documentation, where knowledge flows horizontally as software is developed. The vertical process (r-a-d-c-t) corresponds to the knowledge gained through training and work experience. The latter flow is complimentary to the former due to its movement across flow times, between individual and group domains, and from tacit to explicit and back; whereas the only variation in the horizontal flow is the domain of time, as the software is developed.

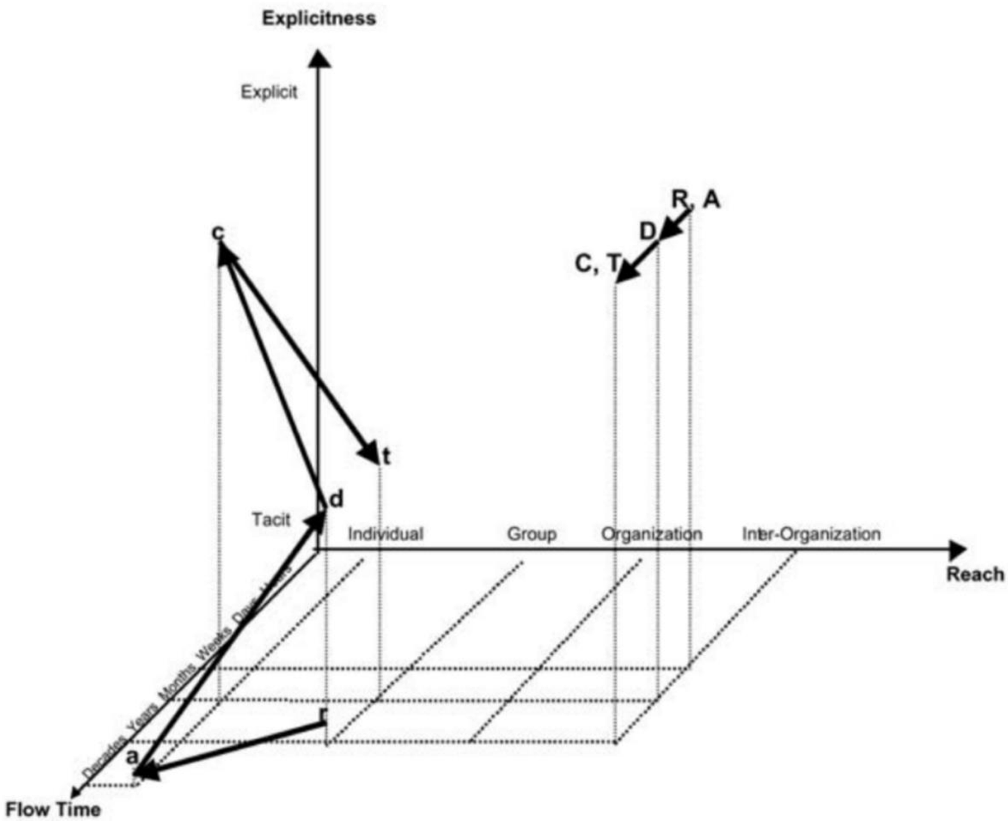

Figure 11. Model of PM knowledge flows[106]

[105] (Snider & Nissen, 2003)

[106] (Snider & Nissen, 2003, p. 9)

3.3.2 Existing Models of SPM

Breaking down the specific phases of a software development project into definable processes that can be replicated is the first step in modeling SPM. The phases of a software project can be simplified to *feasibility*, defining the project and capabilities needed to complete it; *requirements gathering*, the back-and-forth process between client and company that documents formal project requirements; *design*, planning of the project structure; and *implementation*, where the project is constructed and delivered.[107] The development of software products depends on incrementally adding features; thus, PM methods are recurrent and cyclical, as every increment requires the necessary planning and validation of project activities. The processes then follow a consistent template, because for certain activities, such as testing or frontend development, there needs to be certain proceeding events completed. This makes the different SPM process groups highly accessible to model and simulate while considering cognitive factors.

Researchers developed a system dynamics model of SPM incorporating both software development functions such as designing, coding, and testing, with PM activities such as planning, controlling, and monitoring.[108] The unique aspect of their approach is the feedback system that modifies how variables interact based on the complex dynamics between activities within the project environment. Figure 12 shows the early developments: a high-level model of SPM dynamics, which displays the basic cyclical nature of software projects. In this figure, each numbered item is a sequential step in the feedback loop of a project; showing how software projects are accomplished through resource utilization. If changes in steps 4 or 6 cause further adjustments in the level or distribution of project resources (step 6), the loop repeats from step 1. While this is recognized as a basic overview of the project cycle, the researchers further adapt this model to include dynamic feedback.

Further developments to their model can be seen in Figure 13. Here, project processes are grouped into sections according to the various activities of a software project manager (HR Management, Software Production, Planning, and Control), and the cyclical nature of projects is still on full display. What they adapted in this model, however, is the feedback mechanism of activities within the project. It is evident in this model that activities have variable effects on others. For example, the rate that software is developed can moderate both the rate at which errors are created, the perception of created tasks, project knowledge, and the actual and perceived productivity.

[107] (Nguyen, 2006, p. 66)
[108] (Abdel-Hamid & Madnick, 1989)

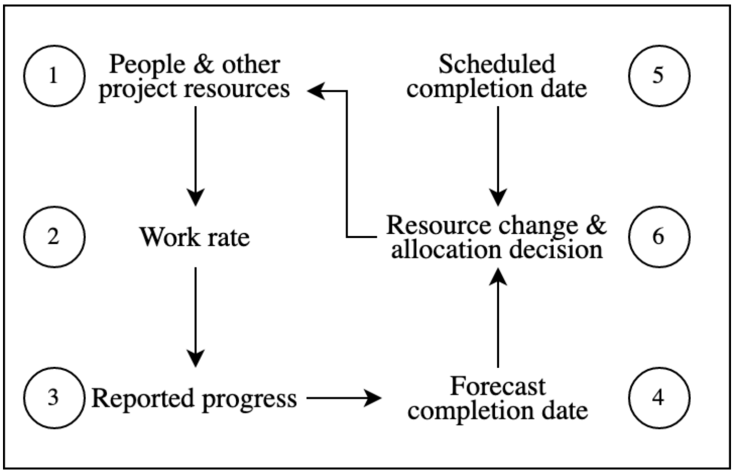

Figure 12. Basic model of SPM[109]

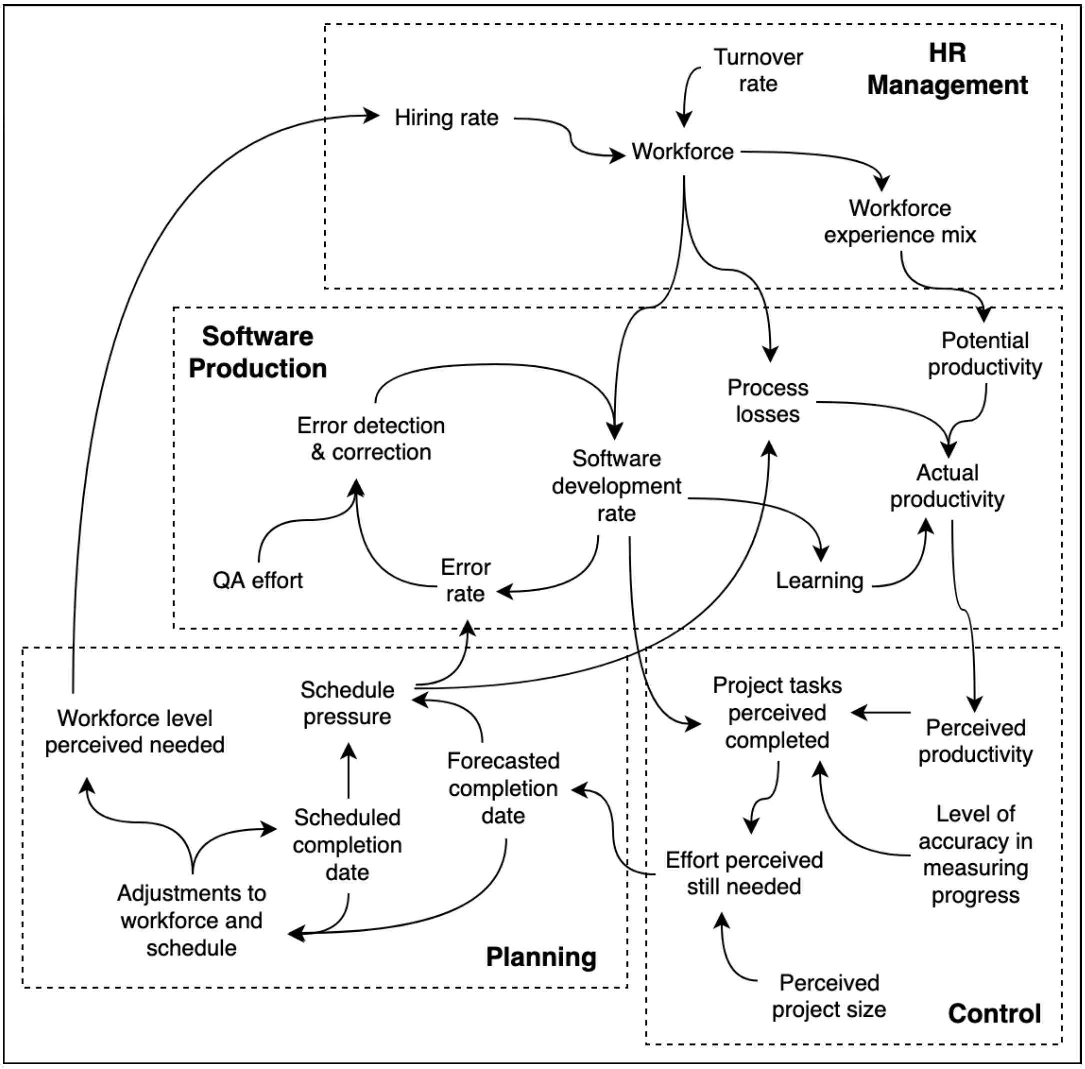

Figure 13. System dynamics model of SPM[110]

109 (reproduced from Abdel-Hamid & Madnick, 1989, p. 1427)

110 (reproduced from Abdel-Hamid & Madnick, 1989, p. 1430)

Other SPM models exist that take a differing approach in modeling the dynamics between project factors. Sukhodolsky proposed a deterministic model of SPM (Figure 14) that implements an important step in the control domain of PM.[111] In the first stage, product features are created from specifications, and in future iterations, the features are tested and/or corrected according to management decisions; the second stage is where measurements for analyzing project productivity and progress are collected; in third stage, measurements are analyzed; and the fourth stage is where management decisions for control are applied to modify the project plan. Each cycle of the control phase increments the cost of the project.

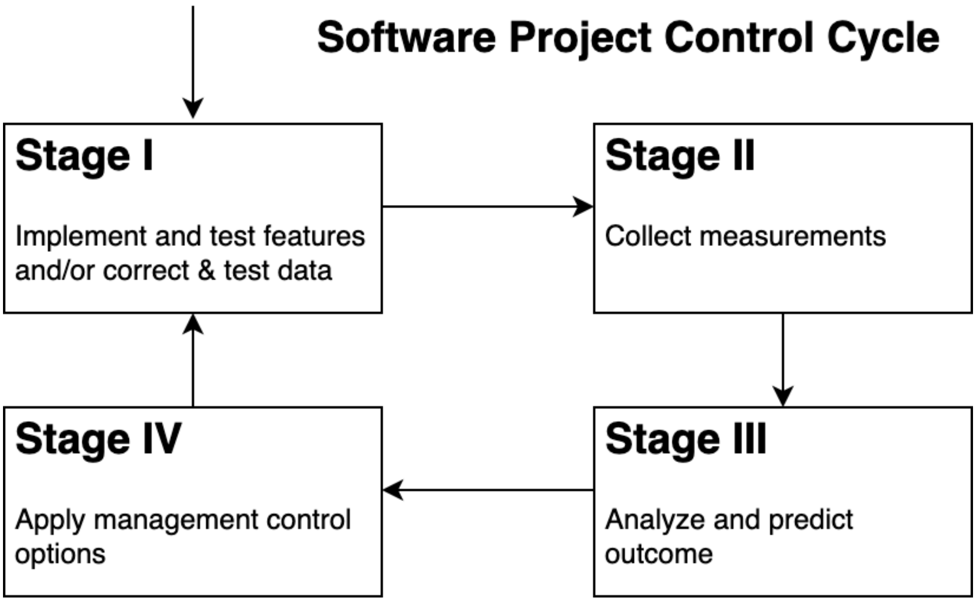

Figure 14. Sukhodolsky's model of software process control[112]

Researchers have developed a model of SPM called Software Project Management Net (SPMNet) that provides near-optimal solutions to software project problems by applying genetic algorithms to the solution-search space.[113] The 'Net' wording of the model refers to its foundations, borrowing key concepts (such as tokens) from petri nets and adapting beyond them. Petri nets are a modeling technique where tasks, users, and technical systems can be simulated, and its results analyzed.[129] The SPMNet model contains different sets: *places*, *transitions*, *constraints*, and *arcs* (see Appendix B for the complete formal definition). Figure 15 shows an example SPMNet.

[111] (Sukhodolsky, 2001)
[112] (reproduced from Sukhodolsky, 2001, p. 60)
[113] (Chang et al., 1998)

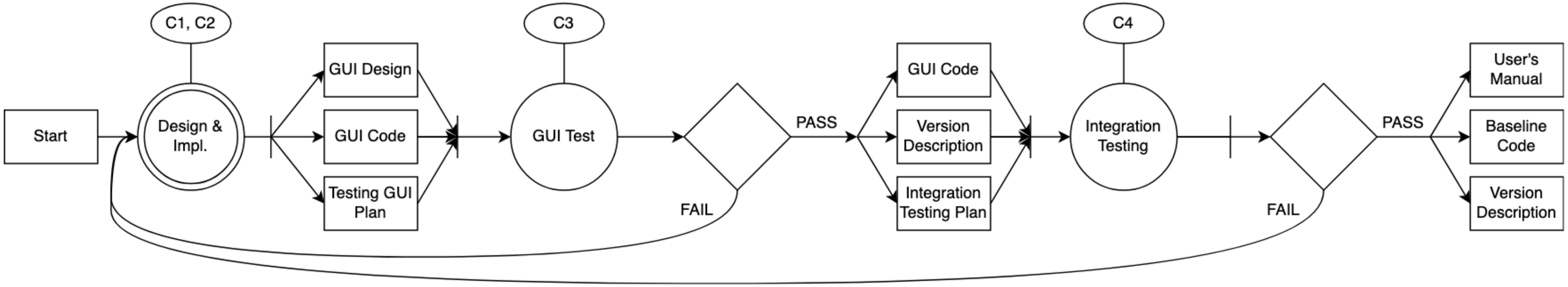

Figure 15. An example SPMNet model of SPM.[114] Rectangles represent product places; ovals represent constraints that managers can optimally place inside the system; circles represent atomic activity places; double circles represent abstract activities; and diamonds represent decision places.

[114] (adapted from Chang et al., 1998, p. 538)

3.3.3 Existing Models with Cognitive Components

Various models exist that either include cognitive components in the infrastructure of their approach, or they adapt basic modeling methods into cognitive systems. Some of these variations are provided as follows.

Fuzzy Cognitive Map (FCM) modeling combines fuzzy logic and neural network algorithms to solve complex problems;[115] therefore, FCMs are a graphical way of representing fuzzy rules that define a specified environment. Factors in FCMs are represented as nodes, with connecting arcs which have an attached positive or negative value describing their causal relationships (positive represents a direct relationship while negative equals an inverse one).[116]

There are a few instances of FCMs being used specifically used for modeling various aspects of SPM. First, Tlili et al. developed a weighted FCM model that specifically examines software project risk, and identified five major risk factors: *bad task scheduling, unskilled developers, technological aspects, budget limitations,* and *fuzzy objectives*.[4] This model was further developed (Figure 16) to combine reinforcement learning (RL) within the FCM. RL helps model decision-making by first evaluating a set of decisions by trial and error, then computing their posterior distributions against the outcomes.[117] Second, Chernova et al. adapted a broader approach, modeling project implementation of software projects using FCMs (Figure 17).[7] Instead of specific weights attached to each arc, the model shows a direct or inverse causal relationship between factors. For example, an increase of risks negatively impacts both project duration and resources, while an increase in resources, although it adds to the project duration, positively impacts the product quality.

Petri nets are one method of modeling human interactions that are cognitively inspired. One researcher integrated task and user models with Petri nets to simulate and study task performing behavior.[118] Figure 18 provides a visual display of the parallel operators of this model, while Figure 19 provides a closer look at the flow of tasks within the model. It is apparent, looking at these two examples, that tasks are the main building blocks used when incorporating Petri nets with cognitive user and task models.

---

[115] (Case & Stylios, 2016, p. 2)

[116] In the case of *weighted* FCMs, they have an attached value; but it is often permissible to use either a +/- symbol to represent a direct or inverse relationship between nodes.

[117] (Bagherzadehkhorasani & Tehranchi, 2022, p. 2)

[118] (Kontogiannis, 2005)

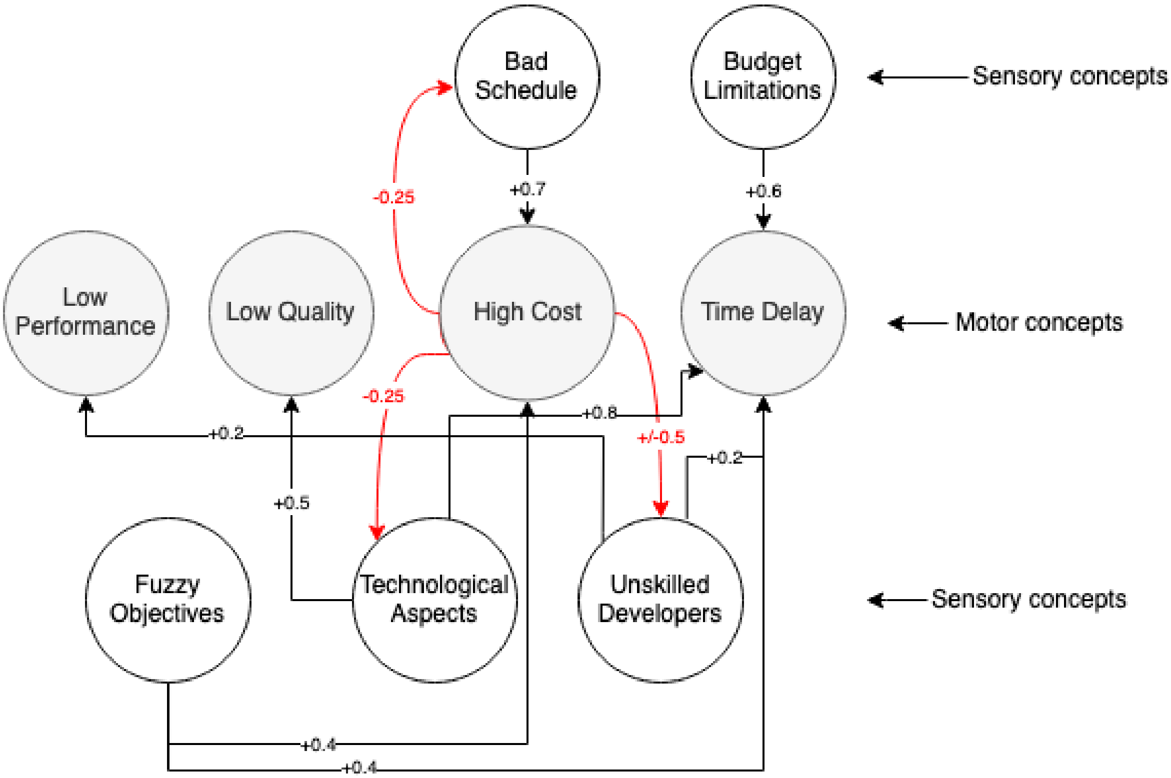

Figure 16. An FCM model of SPM risk factors with RL.[119] Sensory concepts represent the risk factors that either directly or inversely affect outputs of risk (motor concepts). The addition of RL shows that feedback from high project costs inversely affects risk factors such as technological aspects and bad schedule.

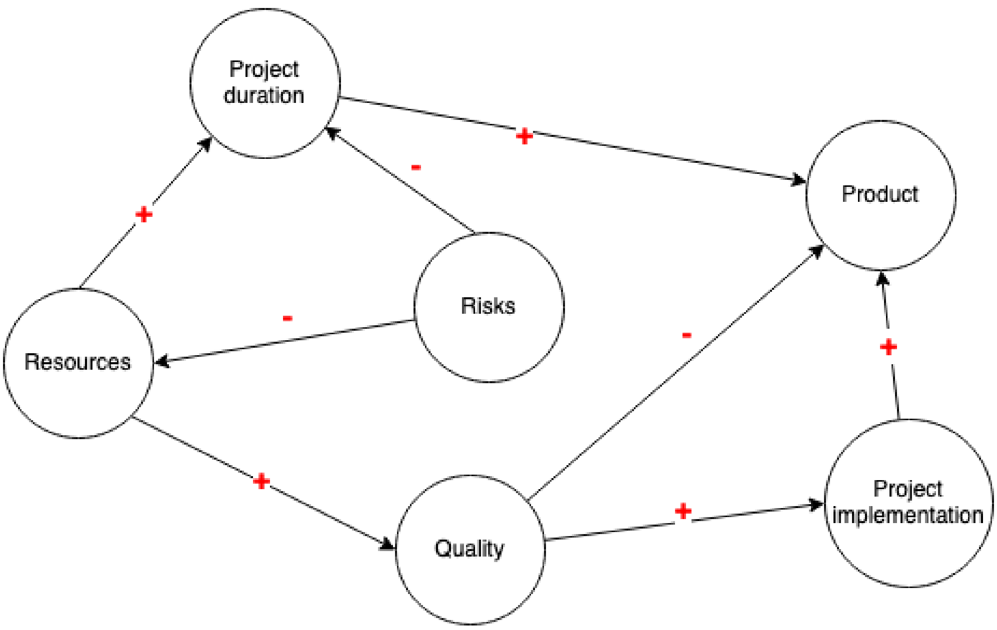

Figure 17. FCM model of SPM project implementation[120]

[119] (adapted from Tlili & Chikhi, 2021, p. 137)
[120] (reproduced from Chernova et al., 2022)

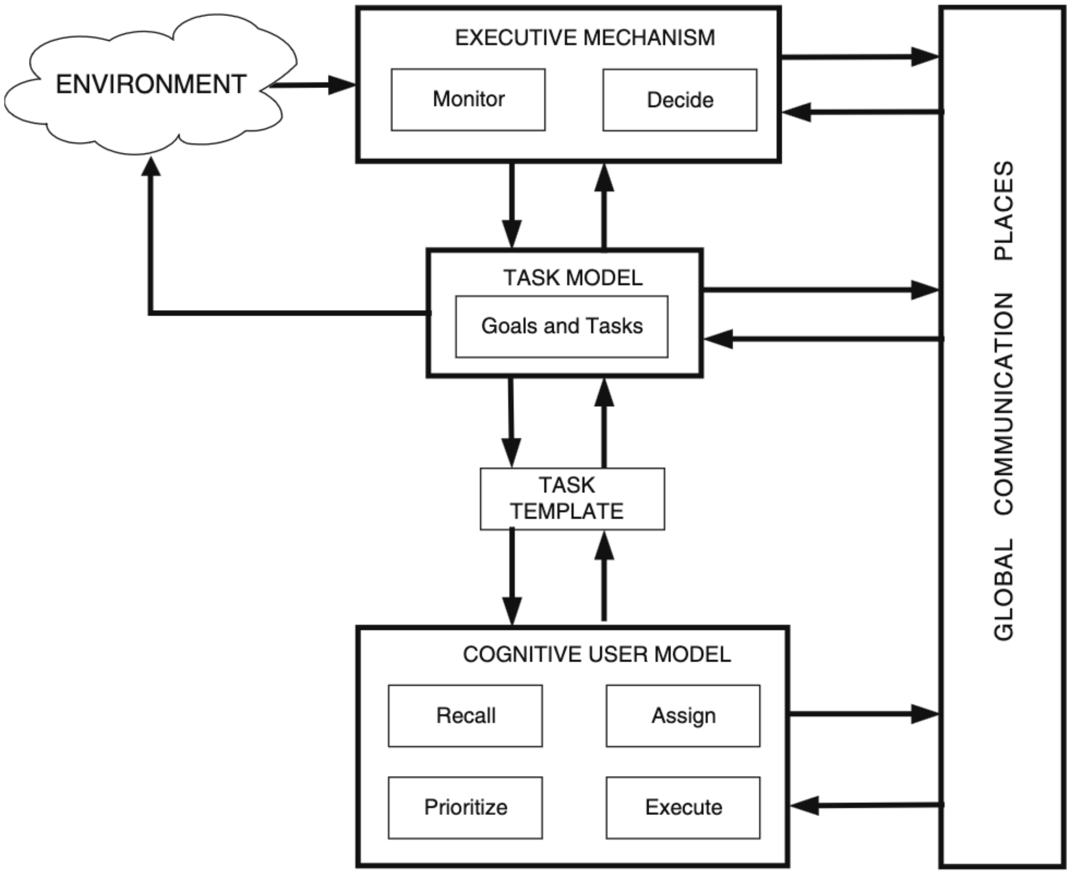

Figure 18. One example of an integration of user and cognitive models.[121] The Task model organizes tasks required to achieve goals; the cognitive user model controls how agents cognitively select and prioritize tasks; and the executive mechanism is responsible for monitoring external events and making decisions about task processing. All these mechanisms operate in parallel with each other.

Business Process Model (BPM) methods have been introduced to cognitive aspects as well. Figure 20 shows an example of a process model that uses cognitive agents to simulate workflows. In this model, each element inside a rectangle represents a cognitive agent; some event that controls and interacts with other agents according to specified rules. As a conceptual diagram, this model provides an overview of how each of the cognitive agents interact, and the nature of their interactions. A further look into how the cognitive agents in their model specifically operate is provided in Figure 21.

[121] (Kontogiannis, 2005, p. 243)

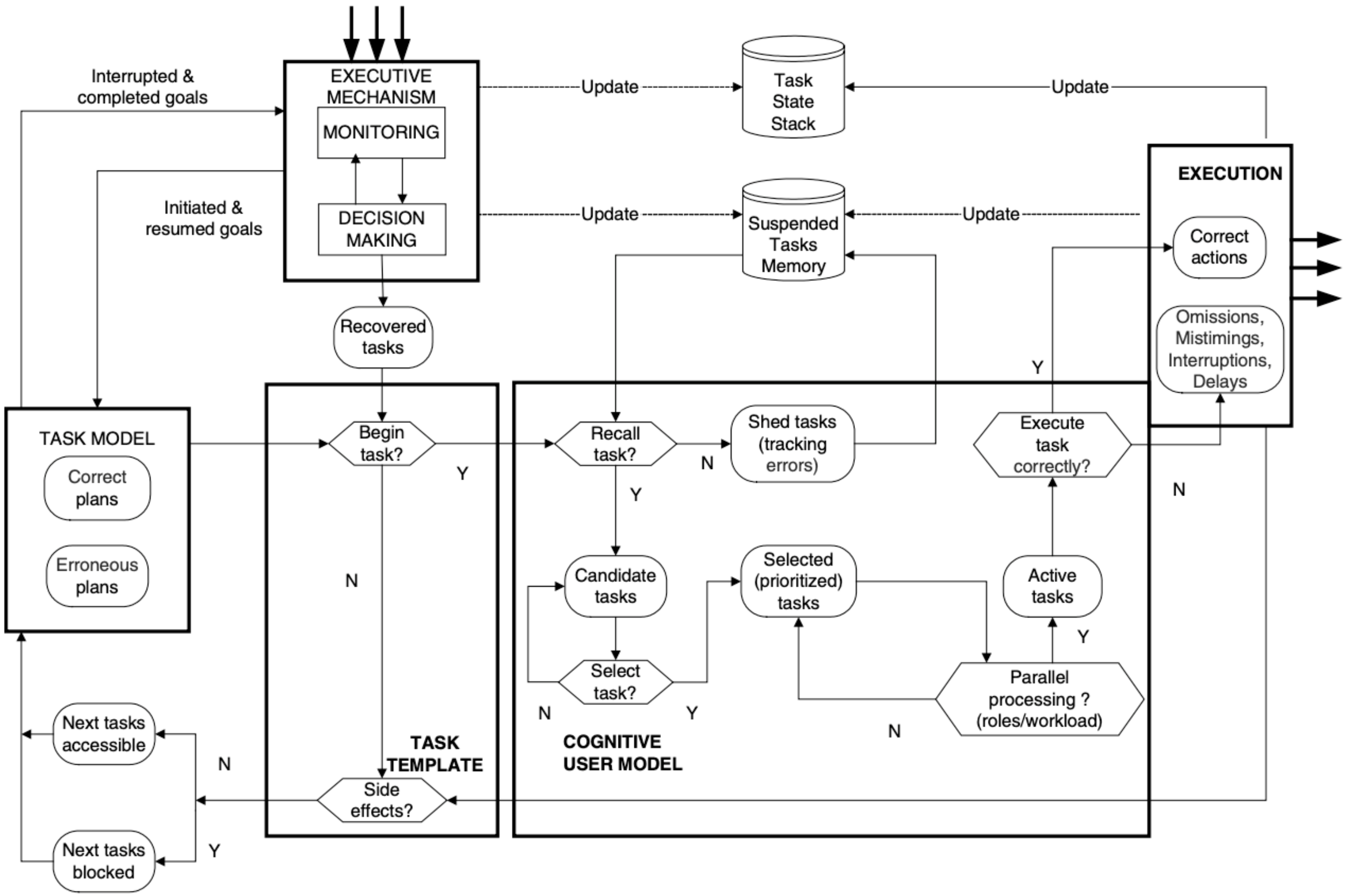

Figure 19. The flow of decisions in and out of a cognitive user model[122]

[122] (Kontogiannis, 2005, p. 245)

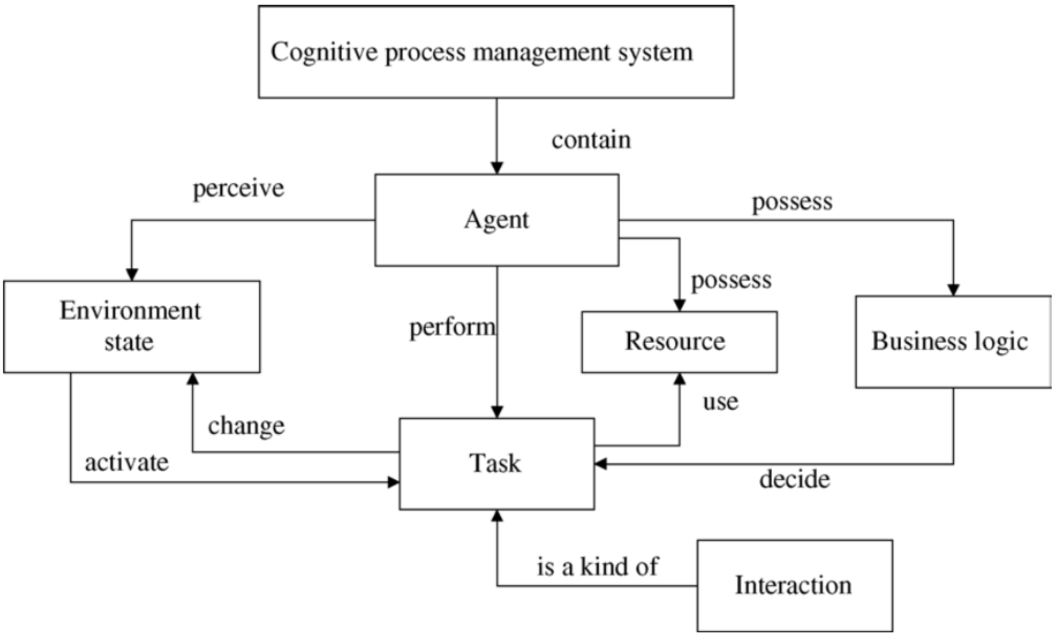

Figure 20. A cognitive process management model.[123]

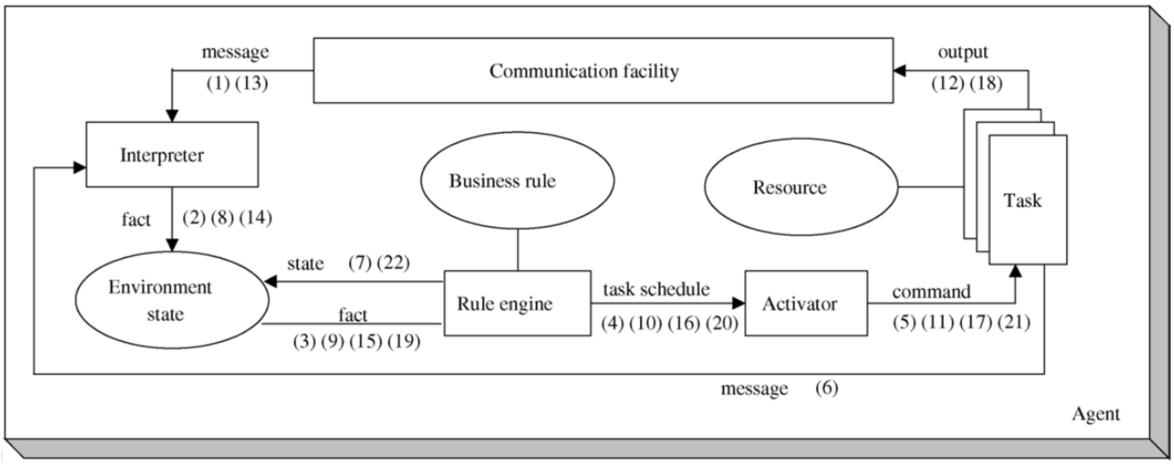

Figure 21. How a cognitive agent operates inside the cognitive process management model.[124] The process flow is in order from points 1-21: (1) A message is sent from an external agent; (2) The event is transformed according to entered rules; (3) A specific rule is fired; (4) A specified task is set in motion; (5) The task is activated; (6)The request is sent, but there is an error; (7) The error is recorded into a report for later; (8) The error triggers a new interpretation; (9) New rules are fired accordingly; (10) Another task is scheduled and (11) activated; (12)The new request is sent to the external agent; (13) The request arrives and the process is repeated inside another agent for steps 14-21.

---

123 (M. Wang & Wang, 2006, p. 186)

124 (M. Wang & Wang, 2006, p. 190)

4 Results

Included studies in the SLR and outcomes from document classification and clustering will be methodically detailed in the following section. Section 4.1 provides the outputs of the SLR in terms of literary volume and source variety. Following that section, the results of document classification and keyword extraction will be presented for further discussion. Finally, in section 4.2 the clustering results will be analyzed in their relation to the included literature of the SLR and the research questions attached to the overall study scope.

4.1 SLR Results

Precise constraints were used throughout each phase of searching, screening, and selecting literature. Because software with ML capabilities was used to help facilitate these processes, outcomes were efficiently and accurately drawn. Advanced tools were not relied on too heavily, however, as manual selecting and filtering managed to draw more results. For example, the search results in the proceeding section produced around 10,000 studies to screen; but the number of studies selected that were the result of citation screening and snowball sampling (see: Figure 25) were nearly half of the included studies (46% of 97 total). This highlights both the effectiveness of the search strings and databases used, and an area which can be improved upon in future iterations. One improvement that has supported this step of an SLR for other researchers is the use of ML tools for creating search strings.[125] Nevertheless, sufficient resources were utilized for the entire search process.

4.1.1 Search Results

The search strategy was adapted for each source and therefore needed specific refinement. The search strategy flow for every source is detailed below in Table 3, which requires a brief explanation of the process and rationale of each. The following sources were used to find studies:

1. **IU Library**. After an initial wide-scoping search resulted in 13,685 articles, a filter was added to delimit the source type of results (to remove 'magazines' and 'news' types). With 10,073 results remaining, a filter was used to restrict subjects: all subjects relating to *mental health*, *healthcare, learning, development,* or *construction/industry*. A further update to the search string to restrict subjects (*construction, industry,* and *offshoring*) produced 4,721 articles which was deemed enough to proceed with. Finally, more variations of

[125] (Jiminez et al., 2022)

search strings were used in the same register to cover the full scope of research. This produced another 642 results.

2. **Google Scholar**. Iterative updates were not necessary from this source, but the approach was adapted slightly: When a relevant article is identified from the search, there is a link called 'Related articles' that directs the researcher to a new list of results of citations that are semantically related. Only the articles that included multiple study topics (i.e., CM *and* BPM *and* SPM) were scanned for related articles. This process resulted in a total of 213 citations.
3. **Epistemonikos**. There was no further refinement needed from this source, as the amount of related SLRs is relatively low for the research scope. 14 total results were returned.

4.1.2 Screening Results

As there was an individual screener, the screening phase was a multi-stage process with the assistance of ML tools to aid the screening. Figure 22 shows the prediction scores of the first screening phase in Abstrackr. 1,001 citations were screened on title/abstract, and a total of five citations were predicted to be relevant to the study (of the remaining and unscreened citations).

Results of the keyword extraction process can be seen in Figure 24. Keywords from this phase were used to train a classification model using the k-nearest neighbors technique (kNN). Test scores of the model were sufficient (Figure 23) and helped to classify documents into clusters (

Figure 27) used for further analysis in section 4.2.

4.1.3 Selection Results

Figure 25 shows the results for the process of selecting studies using the PRISMA 2020 flow diagram. A total of 97 studies were included in the SLR.

4.1.4 Risk of Bias in Studies

Methods used to assess risk of bias in included studies were focused on confirmation and publication bias. Publication bias was mitigated by including all grey literature (reports, conferences, theses/dissertations) in the full study. Confirmation bias was mitigated using ML tools to supplement screening decisions.

Table 3. Search strategy and results

Date	Source	Search String	Results	Update Performed
04/04/2023	IU Library	cognitive modeling AND software project management	10,073	Subjects filter; removed all mental health, health related, learning, development, construction/industry
		cognitive modeling AND software project management NOT SU construction NOT SU industry NOT SU Offshoring	4,721	Added keywords
		(cognitive science OR behavioral science) AND (software project management OR IT project management)	24	Updated search string
		cognitive psychology AND (software project management OR IT project management) AND human computer interaction	4	
		process model AND (software project management OR IT project management) NOT construction	256	
		human cognition AND cognitive modeling	266	
		behavioral science AND human computer interaction	92	
04/04/2023	Cognitive Computation	cognitive models AND software project management	118	
04/04/2023	Cognition, Technology & Work	cognitive models AND software project management	190	
12/04/2023	Google Scholar	'cognitive' OR 'cognition' OR 'computational' AND 'project management' OR 'project manager' AND 'software' OR 'information technology' -construction -OR -industry -OR -medicine -OR -healthcare -OR -therapy -OR -therapeutical	112	
		human cognition AND cognitive modeling AND limitations	101	
12/04/2023	Epistimonikos	cognitive model software project management	14	

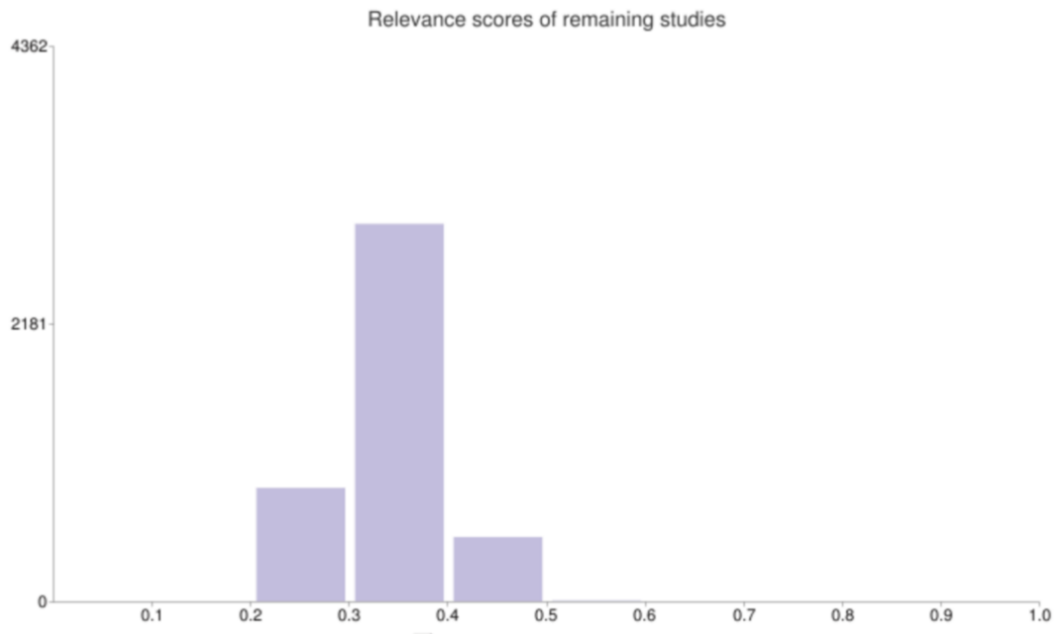

Figure 22. Prediction scores of remaining studies in Abstrackr, where the histogram's x-axis shows the prediction score, and the y-axis shows the count of studies. The graph was automatically generated by Abstrackr after 1,001 citations screened, with 4,362 remaining. The maximum prediction score of remaining citations equaled 0.55 with a predicted 5 relevant citations.

Test and Score Sun Apr 30 23, 11:01:38

Settings

Sampling type: No sampling, test on testing data
Target class: None, show average over classes

Scores

Model	AUC	CA	F1	Precision	Recall
kNN	0.9805682953630176	0.9572261827608555	0.9390795871139538	0.9253661697990926	0.9572261827608555

Figure 23. Scores from the kNN model show a highly accurate model. The Area Under Curve (AUC) score indicates a larger distinction or separability between classes of the model. Classification Accuracy (CA) metric (95%) indicates how accurately the model can predict new data instances. The F1 metric also measures accuracy of the model but also accounts for precision and recall. Precision and recall measure the number of correct predictions made from all possible correct predictions. These scores all being above 92% indicate a highly accurate model.

Word	TF-IDF
knowledg	0.023650469778701274
team	0.018991305029898355
system	0.01814038366852661
project	0.01638051892216778
inform	0.015970174983891007
learn	0.015757167216511263
manag	0.01564625718343219
model	0.015592001445519324
perform	0.01482061240957648
process	0.014803167562449248
user	0.014589714426263367
innov	0.014409975241805076
decis	0.014347487917979568
cognit	0.01418359718612832
technolog	0.013480474193817012
softwar	0.013137762502216365
work	0.012973701698567722
behavior	0.012545299000079868
research	0.012505307537088088
social	0.012459225925508636
develop	0.011915825502068446
data	0.011831679828415072
design	0.011675693177846194
human	0.011272339059761148

Figure 24. Keyword extraction results from Orange that were used to further refine the screening process. The 24 keywords with the highest TF-IDF score are shown (left). A word map was created to visualize the highest scoring keywords (right).

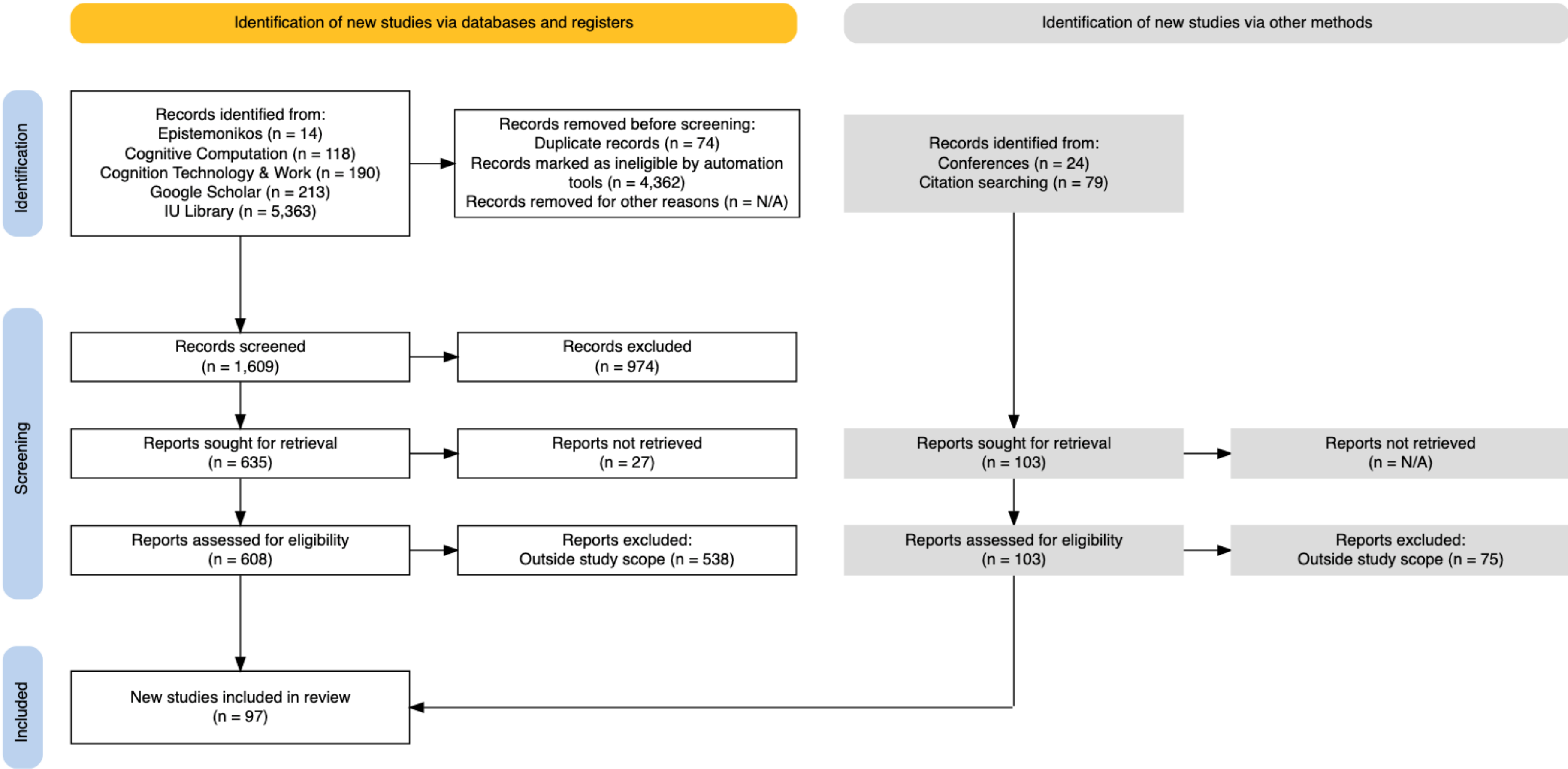

Figure 25. PRISMA 2020 flow diagram

4.2 Cluster Analysis

This section will display the connections across themes and research areas that were uncovered by the SLR through document classification and text mining. Significant associations will be interpreted for each respective clustering phase. Section 4.2.1 will explain how data was synthesized prior to the screening phase in Orange, using a training set of search results. Section 4.2.2 provides an analysis of how the included studies were grouped according to similar keywords in their title and abstract. Therefore, these analyses will provide the groundwork for a critical interpretation of the body of research within the highlighted domains.

4.2.1 Orange Clusters

After the initial screening phase, there were numerous studies remaining that were split into testing and training datasets. Within Orange, a kNN model was trained based on the keyword extraction results. Figure 27 shows the results of the clustering algorithm, and its outcomes are as follows.

The overall structure of the figure provides a document map that displays the semantic similarity between clusters. The clusters are spaced according to how they were arranged in a vector space in the visualization process, which is only a function of showing the clusters' distance from one another.

In each of the six clusters, there is an annotation of the top five keywords related to each study within their clusters. These keywords are significant in highlighting the characteristics of the SLR and the chosen research focus. First, the difference in commonalities between the sets of keywords is quite noticeable. For example, the set that contains *human* aspects (*cognit, human, user, interact, …*) is distant from the one with *social* aspects (*knowledge, social, manage, capital, learn*) and very distant from cluster 4 (*risk, assess, decision, analysis, …*); furthermore, the two clusters that have a *management* keyword are distanced from each other, with one having a more *team* and *decision* focus, while the other has a *social* and *knowledge* focus. Second, clusters 1 and 3 share some overlapping space although they are semantically different (*software, machine, algorithm* vs. *cognition, human, user*). This shows a somewhat unclear distinction between *human* and *machine* aspects amongst groups. Third, there are instances that were not clustered (the points without color on the figure), showing that although there was enough semantic similarity to be grouped in proximity to three clusters, the studies did not score highly for specific keywords; suggesting that, while the topics are very close to *cognition, software,* and *knowledge*, they include topics undiscovered by this clustering algorithm or not relatable to the SLR scope.

As a final note, it is noteworthy that these clusters are very distant in characteristics yet still included within the SLR scope. This ultimately provides a justification for further research connecting the domains of *cognition, software, AI, knowledge, and risk*.

4.2.2 Lingo3G Clusters

Clustering was also enacted on the studies selected in the final screening phase. The comparison between analyses of clusters is significant since the selected studies were manually extracted and separated from studies that did not fit the research scope. Accordingly, one could presume that the clusters should already share semantic similarities. In this line of thinking, the task for clustering in this phase is to focus more on the individual studies that were included in each cluster, along with how the main cluster labels were generated.

When looking at the overall makeup of each cluster, there are a few different analyses that could support. First, one could look at the number of included studies per cluster. Figure 26 shows this, which makes sense in the SLR perspective, as most studies are within the *Cognitive Modeling* and *Project Management* clusters. Second, one could examine the breakdown of individual studies in the clusters. Table 4 lists each study included in each of the clusters, with labels automatically created by EPPI-Reviewer (the surname of the first author and year of publication).[126] Table 5 provides a further breakdown of studies organized by which clusters they were grouped into, shown in descending order of the most clustered to least. The table also shows which research question the studies were coded for in data extraction, and the respective sub-topic labels which were automatically added during clustering. Finally, Figure 28 shows the distribution of selected studies by publication year, where one could observe a slight upwards trend in published studies from the years of 1994 to 2022.[127]

Cross-Tabulation of Clusters

Examining the correlation of cluster topics requires a precise look at the studies which share relations. Figure 29 provides a grid of the number of studies in common between each sub-topic of the clusters.

Cognitive Modeling & Project Management. The studies that share these clusters apply formal modeling techniques in a variety of ways. A common bridge between models and PM practices in the studies are with *Business Process Models* and *Fuzzy*

---

[126] While there was a small number of studies grouped into an 'Other' cluster, these were not listed in this table and treated as outliers to the cluster data.

[127] Although included in the SLR, the year 2023 was left out of the graphic as it would only include the first few months of the year and would skew the displayed results.

Cognitive Maps; suggesting that the use of FCMs to model SPM is one proven method of applying CM to SPM processes using BPMs. Additionally, the sub-topics of *Software Development* and *Software Project Management* connect CM and PM with two core topics: *cognitive complexity* and *cognitive maps*. This implies the applicability of cognitive-related concepts in BPMs.

Cognitive Modeling & Cognitive Biases. Although it can be assumed that articles found in both clusters will be related to cognition in some ways, there is a connection between studies found in *Probabilistic Models of Cognition* and *Cognitive Biases*. With this, it can be suggested that there are inherent biases specific to probabilistic models over other cognitive models. It also provides insight into how cognitive biases can be represented in CM approaches. While no specific cognitive bias sub-topics (*Planning Fallacy* or *Availability Bias*) are found in connection with CM, there is connection between *Reinforcement Learning, Decision-Making Processes*, and *Task Environment*.

Cognitive Modeling & Decision-Making. This group sees the highest conjunction of behavioral and cognitive science topics. The connection with the highest quantity of studies (4) is *Decision Model* and *Task Environment;* studies with these labels aim at modeling decision-making using a variety of cognitive-based strategies (NDM-based,[128] Petri nets,[129] etc.) and measure their performance based on the execution of specific tasks. The instance of CM and DM both being major cluster groups tells one that DM modeling is approachable using CM techniques, and that when discussing CM, DM is at the top of the list of related subjects being modeled.

Project Management & Decision-Making. The sub-topic of *Software Development Projects* within the DM cluster had the most connection with the *SPM* sub-topic within the PM cluster. The articles attached deal with simulation within SPM,[130,131] resources allocation and prediction,[132,133] and SPM decision models.[134,135,136] The *Indi-*

---

128 (Fan et al., 2010)
129 (Kontogiannis, 2005)
130 (Medeiros, 2015)
131 (Kouskouras & Georgiou, 2007)
132 (Ge & Xu, 2016)
133 (Masoud et al., 2018)
134 (Colomo-Palacios et al., 2013)
135 (Cunha et al., 2016)
136 (Cunha & Moura, 2015)

vidual Level sub-topic of the DM cluster included one study that connected with *Decision Model, Project Management Research,* and *Behavioral Decision Making:* a study that modeled individual perceptions of project managers.[137]

Project Management & Cognitive Biases. Articles that belong to the PM cluster only have either *SPM, SE,* or *Behavioral Science* labels, which correlates with the selection of research in this study. In the CB cluster, there is at least one article in each of the generated sub-topics; and it is noteworthy that two sub-topics were created that are cognitive biases in of themselves, *Planning Fallacy* and *Availability Bias.* This highlights the prominence of these two biases within SPM and could help to aim further biases research in this direction.

Cross-Tabulation of Clusters and RQs

Another cross-tabulation that would prove to be effective is the correlation between clusters and RQs. Figure 30 exhibits the accuracy of clustering and extraction (RQ1 is on cognition and represented well in the CM cluster; RQ3 is on project management and shows the most distribution in the PM cluster), and it also shows the opportunities to synthesize connections between research questions.

RQ1 is almost solely disbursed inside the CM cluster, which is logical given that the question is largely centered around CM. It is notable but unsurprising that the highest number of studies resides within the *Human Computer Interaction* label. **RQ2** shows a similar distribution, with one article belonging to the *behavioral science* label. **RQ3** is disbursed into almost every sub-label of the clusters. Because RQ3 examines the preconditions of cognitive science in SPM, this connection suggests that Decision-Making and Cognitive Biases cluster labels are representative of these preconditions. **RQ4** interestingly does not include many studies within the latter two clusters, although RQ4 is directly related to the preconditions mentioned before. Alternatively, one could examine how the relation of RQ4 to the thresholds of cognitive science in SPM modeling is only relevantly related to *modeling*; suggestive of the idea that these thresholds are dependent on modeling strategies.

[137] (Hackman, 2021)

Table 4. Included studies by cluster

Cognitive Modeling	Project Management	Decision-Making	Cognitive Biases
Aranda (2005)	Abdel-Hamid (1989)	Afacan-Seref (2018)	Chanceaux (2014)
Bagherzadehkhorasani (2022)	Afacan-Seref (2018)	Baia (2015)	Cunha (2015)
Baia (2015)	Anderson (1996)	Chanceaux (2014)	Cunha (2016)
Chater (2008)	Barros (2000)	Chen (2008)	Fechner (2015)
Chen (2010)	Bendoly (2007)	Cunha (2015)	Mohanani (2018)
Cunha (2016)	Borji (2012)	Cunha (2016)	Morita (2019)
Fan (2010)	Chernova (2022)	Flyvbjerg (2021)	Tlili (2021)
Fechner (2015)	Cunha (2015)	Griffiths (2012)	Wang (2010)
Fernandes (2022)	Cunha (2016)	Gruhn (2006)	Yang (2022)
Fleischmann (2014)	Emond (2003)	Hackman (2021)	
Flyvbjerg (2022)	Fagerholm (2022)	Hiatt (2022)	
Griffiths (2010)	Ferreira (2009)	Kieras (2015)	
Howes (2009)	Gray (2008)	Kutsch (2005)	
Jiménez (2021)	Gruhn (2006)	Mohanani (2018)	
Kennedy (2010)	Hackman (2021)	Morita (2019)	
Kennedy (2012)	Jarecki (2020)	Prezenski (2017)	
Kieras (1997)	John (1994)	Reitter (2010)	
Kouskouras (2007)	Jöhnk (2020)	Schürmann (2020)	
Kutsch (2005)	Kieras (2015)	Snow (2007)	
Kyllingsbæk (2006)	Kontogiannis (2005)	Stingl (2017)	
Laird (2010)	Kriegeskorte (2018)	Yang (2022)	
Lakey (2003)	Heloisa (1995)		
Lee (2019)	Ritter (2019)		
Liu (2009)	Rubinstein (2001)		
Mair (2009)	Snow (2007)		
Mair (2012)	Tlili (2021)		
Mohanani (2018)	Vinciarelli (2015)		
Montibeller (2015)	Williams (2003)		
Nobandegani (2019)			
Prezenski (2017)			
Reitter (2010)			
Riesterer (2020)			
Ritter (2019)			
Sanborn (2010)			
Schürmann (2020)			
Snider (2003)			
Stingl (2017)			
Sukhodolsky (2001)			
Sun (2006)			

Cognitive Modeling	Project Management	Decision-Making	Cognitive Biases
Tavares (1994)			
Tavares (2002)			
Tsesliv (2022)			
Valiente (2012)			
Venkatesh (2018)			
Vinciarelli (2015)			
Wang (2006)			
Wang (2010)			
Wu (2022)			
Yang (2022)			
Zugal (2011)			

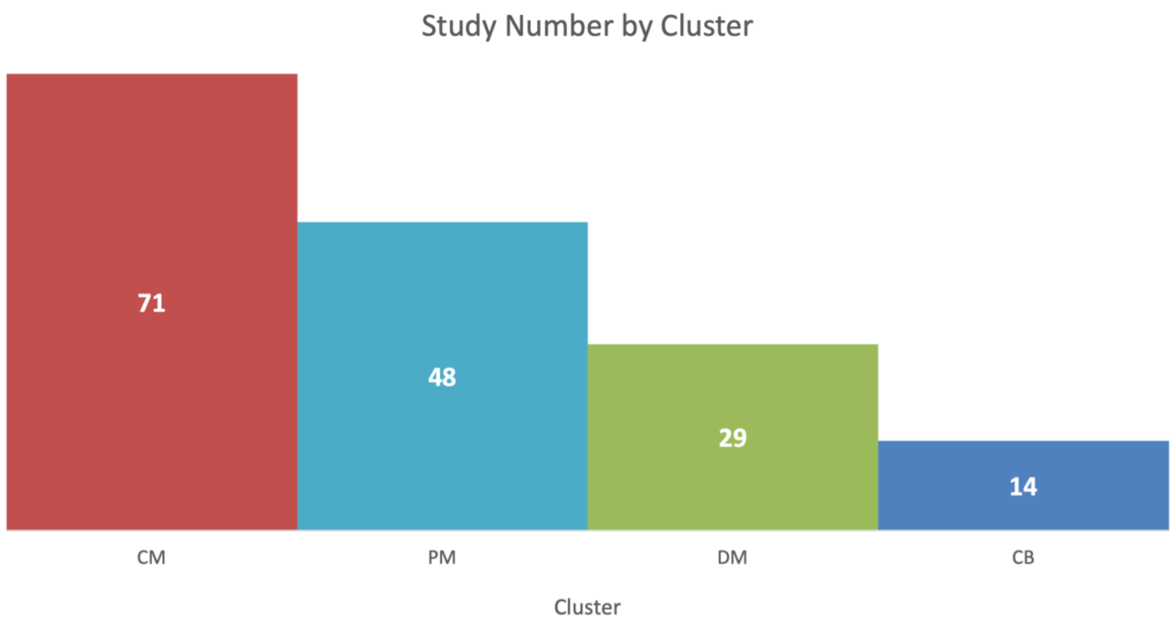

Figure 26. Study number by cluster

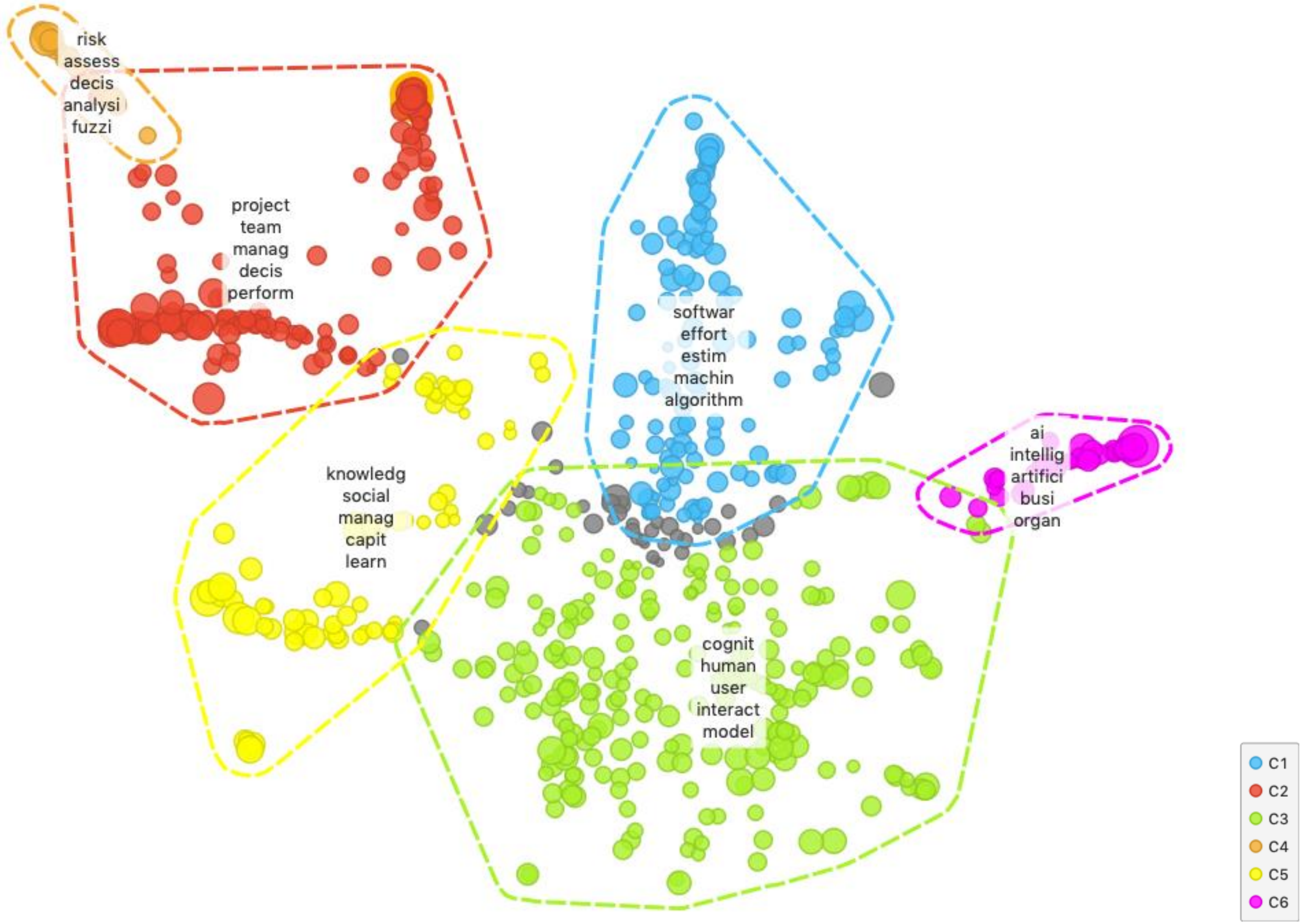

Figure 27. Clustered training dataset

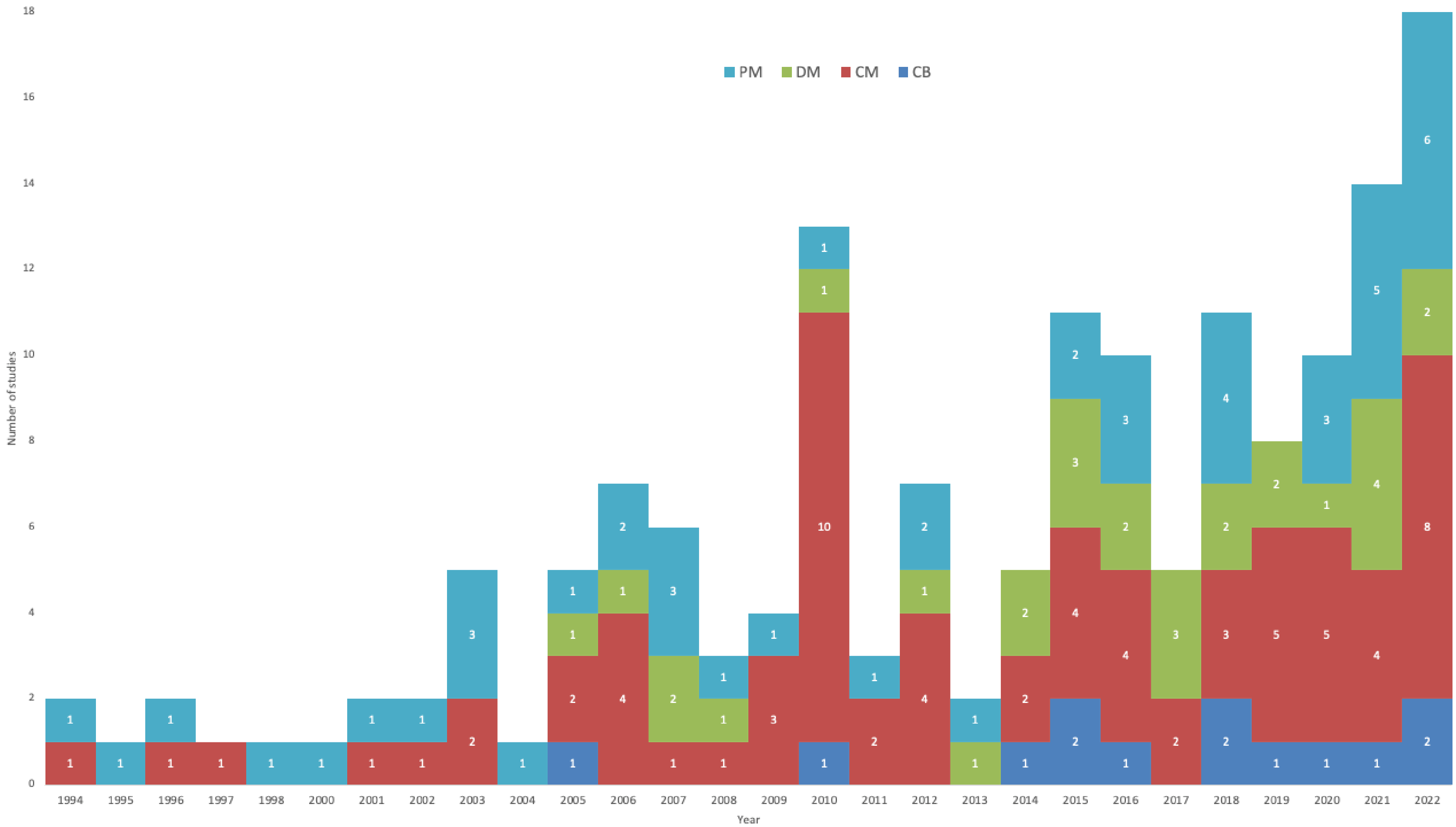

Figure 28. Study number by year and cluster

Table 5. Results of selected studies by cluster

Cluster(s)	Item	RQ(s)	Sub-topics
CM, PM, CB, DM	Cunha (2016)	3	Computational Models
CM, CB, DM	Mohanani (2018)	1	Information Technology
	Yang (2022)	2	Cognitive Processes Decision-making Processes Decision Model Cognitive Architectures Task Environment Reinforcement Learning
PM, CB, DM	Cunha (2015)	3	Software Development Business Process Models Software Project Management Business Process Models
CM, DM	Kutsch (2005)	1, 2	Software Development Projects Project Management Research Support Software Project Software Project Management Project Cost
	Baia (2015)	1	(Other topics)
	Prezenski (2017)	2	Cognitive Architectures Cognitive Science
	Reitter (2010)	1	(Other topics)
	Stingl (2017)	1	(Other topics)
	Schürmann (2020)	1	Behavioral Science Project Management Research
CM, CB	Fechner (2015)	1	Project Management Research Project Cost
	Wang (2010)	1	Cognitive Processes Software Project Management Software Engineering
PM, CB	Tlili (2021)	2	Project Cost Project Risk Information Technology
PM, DM	Afacan-Seref (2018)	3	Human Computer Interaction
	Gruhn (2006)	3	Cognitive Architectures Task Environment
	Hackman (2021)	2	Cognitive Processes Cognitive Science Probabilistic Models of Cognition
	Kieras (2015)	4	Information Systems

Cluster(s)	Item	RQ(s)	Sub-topics
	Snow (2007)	3	Cognitive Architectures
CM	Mair (2009)	1, 2, 4	Decision Model Complex Cognitive Task Environment
	Wang (2006)	2, 3, 4	Individual Level Cognitive Science
	Fernandes (2022)	4	(Other topics)
	Flyvbjerg (2022)	1, 2	Cognitive Architectures Computational Models Human Computer Interaction
	Wu (2022)	2, 4	Business Process Models
	Jiménez (2021)	1, 4	Computational Models Probabilistic Models of Cognition
	Kieras (1997)	1, 2	Computational Models Psychological Processes Complex Cognitive Information Systems
	Lakey (2003)	1, 2	Parameter Estimation
	Lee (2019)	1, 2	(Other topics)
	Ritter (2019)	1, 4	Psychological Processes Parameter Estimation Decision Model
	Tavares (2002)	1, 4	Project Cost
	Vinciarelli (2015)	2	Software Project Management Information Technology
	Aranda (2005)	1	Software Project Management
	Chater (2008)	1	Decision Model Behavioral Decision Making Information Systems Task Environment
	Fan (2010)	1	Cognitive Architectures Computational Models Human Computer Interaction
	Fleischmann (2014)	1	Software Project Management Software Engineering Resource Allocation
	García (2006)	1	(Other topics)

Cluster(s)	Item	RQ(s)	Sub-topics
	Gigerenzer (2008)	1	Cognitive Architectures Computational Models Reinforcement Learning
	Griffiths (2010)	1	Computational Models
	Howes (2009)	2	Software Project Management Project Cost Machine Learning
	Kennedy (2010)	1	Computational Models
	Kennedy (2012)	1	Software Development Projects Software Project Management
	Kouskouras (2007)	2	Cognitive Processes Research and Practice Theory of Project Software Project Management Decision-making Processes Planning Fallacy Software Project Management
	Kyllingsbæk (2006)	1	Computational Models Cognitive Science Human Computer Interaction Software Development Projects Support Software Project
	Laird (2010)	1	Parameter Estimation Software Development Projects Support Software Project
	Liu (2009)	1	(Other topics)
	Mair (2012)	1	Decision Model Software Development Projects Software Project Management Resource Allocation Machine Learning
	Montibeller (2015)	2	Software Project Management Software Engineering
	Nobandegani (2019)	2	Cognitive Architectures
	Riesterer (2020)	1	(Other topics)
	Sanborn (2010)	1	(Other topics)
	Snider (2003)	1	Decision Model Individual Level Project Management Research Behavioral Decision Making

Cluster(s)	Item	RQ(s)	Sub-topics
	Sukhodolsky (2001)	1	Behavioral Science Planning Fallacy Availability Bias
	Tavares (1994)	2	Cognitive Architectures
	Tsesliv (2022)	2	(Other topics)
	Valiente (2012)	1	Cognitive Architectures
	Venkatesh (2018)	1	Complex Cognitive Future Research Psychological Processes
	Zugal (2011)	1	Information Systems
DM	Chen (2008)	1	Cognitive Processes
	Flyvbjerg (2021)	2	Psychological Processes Problem Solving
	Griffiths (2012)	1	Computational Models Human Computer Interaction
	Hiatt (2022)	1	Computational Models Psychological Processes Problem Solving Business Process Models Software Project Management Software Engineering
PM	Abdel-Hamid (1989)	3	Project Risk
	Anderson (1996)	3	(Other topics)
	Anderson (2003)	3	Cognitive Architectures Computational Models Human Computer Interaction
	Barros (2000)	4	Software Project Management Resource Allocation
	Bendoly (2007)	3	Cognitive Architectures Computational Models Human Computer Interaction
	Borji (2012)	3	Software Project Management
	Chernova (2022)	3	(Other topics)
	Emond (2003)	3	Cognitive Science Business Process Models
	Fagerholm (2022)	3	Software Project Management

Cluster(s)	Item	RQ(s)	Sub-topics
	Ferreira (2009)	3	Cognitive Architectures Cognitive Science Task Environment
	Jarecki (2020)	3	Decision Model
	John (1994)	4	Computational Models
	Kontogiannis (2005)	4	Cognitive Architectures
	Kotseruba (2020)	3	Complex Cognitive Cognitive Architectures
	Kriegeskorte (2018)	3	Software Development Projects Project Management Research Support Software Project Cognitive Processes
	Heloisa (1995)	4	Behavioral Decision Making
	Jöhnk (2020)	4	(Other topics)
	Rubinstein (2001)	3	Software Project Management Project Risk Fuzzy Cognitive Maps Software Development Reinforcement Learning
	Williams (2003)	3	(Other topics)

		Cognitive Modeling														Decision-Making									Cognitive Bias						
		CA (cm)	CM	CS	HCI	PP (cm)	TE	PS	SD	BPM (cm)	PE	RL	PMC	FCM (cm)	SPM (cm)	DM	SDP	CC	PMR	BDM	IS	SS P	CA (dm)	PP (dm)	CP	SE (cb)	TP	SPM (cb)	DMP	PF	AB
Project Management	SPM (pm)	0	1	0	0	0	0	0	4	1	0	1	0	1	2	3	8	0	3	0	0	3	0	0	2	1	2	2	1	1	0
	PC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	2	0	0	3	0	0	0	0	0	0	0	0	0
	SE (pm)	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	1	1	0	0	0	0	0
	PR	0	1	0	0	0	0	0	1	0	0	1	0	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	IT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	RA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	0	0	0	0	1	0	0	0	0	0	0	0	0	0
	ML	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	IL	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
	FCM (pm)	0	1	0	0	0	0	0	1	0	0	1	0	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	BPM (pm)	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	IE	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	BS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	1
Cognitive Bias	CP	1	0	1	0	0	1	0	0	0	0	1	1	0	0	1	1	0	1	0	0	0	1	0							
	SE (cb)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0							
	TP	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	2	0	2	0	0	0	0	0							
	SPM (cb)	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	2	0	2	0	0	0	0	0							
	DMP	1	0	0	0	0	1	0	0	0	0	1	0	0	0	1	1	0	1	0	0	0	1	0							
	PF	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0							
	AB	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0							
Decision-Making	DM	1	1	0	0	1	4	0	1	0	1	1	0	0	1																
	SDP	0	0	0	0	0	0	0	1	0	0	0	0	0	1																
	CC	1	0	0	1	0	2	1	0	0	0	0	0	0	0																
	PMR	0	0	0	0	0	0	0	1	0	0	0	0	0	1																
	BDM	0	0	0	0	0	1	0	0	0	0	0	0	0	0																
	IS	0	0	0	1	0	2	1	0	0	0	0	0	0	0																
	SSP	0	0	0	0	0	0	0	0	0	0	0	0	0	0																
	CA (dm)	2	0	0	0	0	1	0	0	0	0	1	0	0	0																
	PP	0	0	0	0	1	0	0	0	0	1	0	0	0	0																

Figure 29. Cross-tabulation of clusters showing the relationship of studies between clusters. Synthesis of studies within each cell provides results for the connection between the topics. Cluster sub-group abbreviations can be found in Cluster Sub-Group Label Abbreviations.

		RQ1	RQ2	RQ3	RQ4
CM	Cognitive Architectures	3	3	0	3
	Computational Models	5	2	2	4
	Cognitive Science	3	1	1	2
	Human Computer Interaction	6	3	1	1
	Psychological Processes	2	0	1	4
	Task Environment	1	0	0	1
	Problem Solving	1	1	0	1
	Software Development	0	0	2	0
	Business Process Models	0	0	1	2
	Parameter Estimation	0	0	0	2
	Reinforcement Learning	0	0	1	0
	Probabilistic Models of Cognition	0	0	0	1
	Fuzzy Cognitive Maps	0	0	1	0
	Software Project Management	0	0	1	0
PM	Software Project Management	0	0	20	9
	Project Cost	0	0	3	1
	Software Engineering	0	0	5	1
	Project Risk	0	0	3	3
	Information Technology	0	0	3	3
	Resource Allocation	0	0	5	1
	Machine Learning	0	0	2	0
	Fuzzy Cognitive Maps	0	0	1	0
	Business Process Models	0	0	1	0
	Internal and External	0	0	1	0
	Behavioral Science	0	1	0	0
DM	Decision Model	0	0	2	1
	Software Development Projects	0	0	5	1
	Complex Cognitive	1	1	0	0
	Project Management Research	0	1	2	0
	Behavioural Decision Making	0	0	1	0
	Information Systems	1	1	0	0
	Support Software Project	0	0	1	0
	Cognitive Architectures	0	0	0	0
	Psychological Processes	0	0	0	0
CB	Cognitive Processes	1	0	2	0
	Software Engineering	0	0	1	0
	Theory of Project	0	0	1	0
	Software Project Management	0	0	1	0
	Decision-making Processes	0	0	1	0
	Planning Fallacy	0	0	1	0
	Availability Bias	0	0	0	0

Figure 30. Cross-tabulation of research questions and clusters. Columns show which research question was identified in the data extraction phase, while rows show the corresponding cluster assignments.

5 Discussion

This section will begin by exploring the scope of reproducibility of cognitive processes using CM techniques. In doing so, a few different theoretical approaches to understanding cognition will be discussed and their relevance to cognitive science research interpreted. This will lead to the discussion of behavioral science and the influence that CM research does, can, and should have on the field. Eventually, different models of PM processes will be presented, and their applicability to SPM will be discussed. Finally, the preconditions for applying such models to SPM, and for developing new conceptual models, will be considered, and the opportunities for the ideas of this study to strengthen the robustness of cognitive science will be outlined in the Synopsis.

5.1 Reproducibility of Cognition

Computational models of complex cognitive processes like decision-making are relevant for understanding the scope of how CM can reproduce cognition. In one study, researchers implemented decision models in ACT-R to further examine the underlying sub-processes of cognition in decision-making strategies;[138] in another, CM was used to study the cognitive interaction of individuals and their task-environments in dynamic decision-making.[39]

Decision-making has also been modeled on a neurocomputational level using weighted alternatives in sensory dynamics, which included biases in model decisions.[139,140] Another study with similar methods observed the effects of motivation on decision-making using rewards to observe behavioral variables in task environments.[141] Although such models with a bottom-up approach are at times a more precise fit to the underworking of human behavior, the current state of cognitive science research depends on the refinement of cognitive theories and building complete theoretical pictures from a top-down perspective.

More research has been conducted in modeling spatial path-planning, where it was argued that path-planning was advantageous to three cognitive domains: creativity, adaptivity, and decay.[142] Creativity and perceptual path-planning are related by humans being able to create new paths to goals with limited visual information, which

---

[138] (Fechner et al., 2015)

[139] (Afacan-Seref et al., 2018)

[140] (Kriegeskorte & Douglas, 2018)

[141] (Vassena et al., 2019)

[142] (Reitter & Lebiere, 2010)

has implications for creatively finding other paths in one's visual field. This assumption can be easily carried over into the SPM domain by assessing the creativity of project managers' decisions in software projects. Adaptivity explains the ability for humans to adapt to environmental stimuli by forming mental models in a form of a mental topological map; this could also be applied to SPM by developing conceptual models of different topological maps in visual project task environments. Decay combines memory recall to explain how behaviors may change as the mental models start to degrade, which also has interesting implications in potential modeling research.

Other cognitive processes have been modeled by researchers as well. One study looked at *syllogistic reasoning* and found that the state of CM focused in this area is in dire need of improvement.[143] Another study adapted ACT-R to mechanistically model motivation based on the *expected value of control theory* which is the dominant theory for fitting motivation inside a computational framework.[144] The exploration of modeling cognitive processes extends to the comparison between different modeling approaches, such as probabilistic and rational models.

5.1.1 Probabilistic vs. Rational Models

The separation between probabilistic and rational models suggests significant insights into the assorted ways in which complex cognitive phenomena are conceptualized and simulated. Researchers have argued that probabilistic models do more to reproduce human cognition because they take a top-down approach by attempting to explain behavior and reducing it to lower-level cognitive processes.[80] Probabilistic models such as Bayesian models of cognition have the scope of modeling multiple cognitive processes, from perception and categorization to inductive reasoning and argumentation,[145] perception, memory, learning, and decision-making;[146] but fail to model algorithmic or process-approaches accurately.[143] Nevertheless, observing humans as probabilistic machines does not account for human behavior when considering all aspects of social environments that form the bases of motivations for rational behavior.

Rational models of cognition provide a deeper understanding of how environmental factors together with psychological processes influence cognition. Conjoining

[143] (Riesterer et al., 2020)
[144] (Yang & Stocco, 2022)
[145] (Chater et al., 2010)
[146] (Lee, 2018)

these levels of analysis into computational models has been completed by some researchers to model category learning,[147,148] with the benefit of including both category *and* reinforcement learning, something that cognitive architectures ACT-R and Soar do not do.[64,149]

Another argument for rational models is supported by Monte Carlo simulation techniques. Monte Carlo simulation has been identified to support rational model development by its capacity to reduce probabilistic computations into a single operation of 'generating samples from a probability distribution.'[150] Furthermore, the features of Monte Carlo techniques are already part of fundamental psychological processes, being used previously in psychological process models for decision-making, as one example.[151]

ACT-R uses a rational analysis approach to model knowledge, asserting that the odds of knowledge being used in a certain context determines how it is made available. In taking this approach, Anderson developed ACT-R to model the domains of memory, categorization, and problem-solving, with the implication that complex cognition used in these domains is simply a 'reflection of one's environment mapped into the cognitive space'[64]. In developing cognitive architectures, like ACT-R and Soar, researchers combine theory and computation to provide comprehensive structures for modeling complex cognition.

5.1.2 Cognitive Architectures

Although many suggest that research attention into the further optimization of cognitive architectures will push the current start of the art for cognitive science,[152] Anderson's ACT-R architecture already models complex cognition at an industry standard level. In the development of ACT-R, they claimed that intelligence is simply the accrual of numerous small units of knowledge that makeup complex cognition; then these parts can be fine-tuned, studied, and the whole of cognition can be replicated using CM techniques.[64] The strength of ACT-R as an approachable architecture comes from it being able to model thought and memory at symbolic and sub-symbolic levels both, and there are several use cases for ACT-R to model various aspects of cognition.

---

[147] (Sanborn et al., 2010)
[148] (Wang & Laird, 2010)
[149] (Laird, 2008)
[150] (Sanborn et al., 2010, p. 1147)
[151] (Chater et al., 2010, p. 13)
[152] (Gray, 2008)

Researchers in one study created a computational cognitive model of users' online 'web-surfing' behavior based on the ACT-R architecture.[65] Their model quantifies web links in terms of relevance to users' goals when using the internet, and it provides a baseline for future research in modeling digital information foraging, which essentially describes the main tasks of a software project manager. ACT-R has also been integrated into 3D game engines to bridge cognitive agents within cognitive architectures.[153] Incorporating cognitive architectures into HCI research could be highly advantageous when considering the cognitive aspects of humans as the interactive agents and its implications for cognitive science research.

ACT-R has also been applied to model the dual processes theory of cognition, to show how the slow process (S-II) directly inhibits the fast process (S-I) in various tasks.[75,154] Other researchers have applied ACT-R to model decision-making and S-I cognition, and were able to replicate human performance on implicit learning tasks; but the interesting aspect of this study is that their developed model was inconsistent with how procedural memory works in humans yet performed better on recognizing intentional tricks in the training data.[155] This suggests two different conclusions: either what we think we understand theoretically about memory structures and their mechanisms is not entirely accurate, or, this knowledge is simply not properly implemented within the ACT-R cognitive architecture.

Other processes such as visual attention can be mathematically modeled using both ACT-R and generalized architectures.[156] Researchers in one study developed a computational cognitive model that replicated the visual attention of searching a web page for textual information;[157] and Kieras et al. implemented models of visual search using their EPIC architecture that further refined preliminary models.[158]

Cognitive architectures for the most part do incorporate motor mechanisms into their constraints of cognition; however, one researcher has formalized a framework that integrated a computational model of reaction time (QN-ACES) within four cognitive architectures (ACT-R, CAPS, EPIC, and Soar).[159] When modeling reaction time, the researcher aimed to answer the question of why there is a delay from the time a stimulus is presented to when a response is initiated. In this way, the creators

---

[153] (Morita et al., 2019)
[154] (Kennedy & Bugajska, 2010)
[155] (Kennedy & Patterson, 2012)
[156] (Kyllingsbæk, 2006)
[157] (Chanceaux et al., 2014)
[158] (Kieras et al., 2015)
[159] (Liu, 2009)

of EPIC and QN-ACES models effectively included subtle aspects of human performance into cognitive architectures from the perspective of HCI. Observing through a lens of behavioral science demonstrates how CM applied to HCI research is an important method in the development of both theories of cognition and HCI behavior.

5.2 Behavioral Science

To understand the impact that CM can have on SPM, it is necessary to examine the interactive behavior of humans and technology. Researchers argue this relevance along with the ability for CM to anchor cyberpsychology theories into cognitive architectures.[160] Behavioral science experiments measuring human performance focus on whether tasks are familiar or not, whether the rules are simple or complex, and if visual cues are present in addition to their quality of information. As one example, Rubenstein et al. created a model of executive cognitive functioning that was used to measure performance in task-switching activities.[161] Their experiments laid the groundwork for future computational models of executive control of cognitive processes after task-switching performance was found to be hindered by familiarity.

Behavioral decision-making research in project contexts also examines heterogeneous perceptions of project managers in its research.[162] In this study researchers highlight the presumption of *random utility theory* (RUT) being applied to a decision space when perceiving possible outcomes. Whether the result of individual perceptions has significant impact on task completion in projects requires further attention; but it is worthy to note the differences in choice between individuals, and the opportunity to include RUT in SPM modeling approaches to improve the scope of rational models.

Of course, there are other approaches to rational choice in behavioral decision-making. Researchers in one study argue that bounded rationality is cognitively constrained.[163] While this assumption, aptly called *cognitively bounded rationality*, adapts the original theory of behavior to reduce limitations due to information processing, it intentionally narrows the space of possible behaviors that can be studied. Researchers argue that by narrowing the behavioral space they are more finely tuning cognitive architectural research to focus on one specific theory or set of theories. In the development of cognitive architectures, research has argued for a multi-level

[160] (Emond & West, 2003)
[161] (Rubinstein et al., 2001)
[162] (Hackman, 2021)
[163] (Howes et al., 2009)

approach that comprises the interaction of both biological activity and the surfacing of behaviors from cognitive processes.[164]

Researchers in another study examined decision-making behavior of IT project managers when making risk mitigating decisions, through the lens of EUT.[165] The EUT model is robust enough to include a large range of possible outcomes, making it a worthy choice to model decisions in PM; however, what the researchers found in this study is a behavioral bias when project managers are to report risk: they will either deny, delay, or avoid uncertainty which leads to project risks not being properly managed. The effects of information processing on project management activities was also researched using decision-making behavior as the independent variable: researchers here suggested that access to greater amounts of situational information in project environments affects project managers' actions and perceptions of both others and their own behavior.[166] The study examines this through a lens of *informed rationality* which basically states that humans' rational understanding of a situation is information-dependent. When cognitive processing is dependent on environmental or situational knowledge and processing, there are common misunderstandings of the behavior that results from so-called 'mental shortcuts' such as biases and heuristics.

5.3 Cognitive Preconditions for SPM

Researchers argue for the use of cognitively inspired models in the understanding of human behavior, as modeling of errors and biases cannot be properly accounted for in pure machine learning algorithms due to their inherently complexity.[183] In an attempt to reduce the computational complexity of cognitive fallacies, researchers have modeled them with 'perspective-implication'[167] relationships, where the existence of one bias implies the existence of another based on their implied relationship. While this view is entirely mechanistic, there is some insight gained about the nature of relationships between cognitive biases that may be implemented in formal cognitive models – or even simple models that are inclusive of cognitive mechanisms. Such relationships are what make-up the formal preconditions that must be considered when modeling the complete cognitive path from stimuli to decision to behavior.

---

[164] (Jiménez et al., 2021)

[165] (Kutsch & Hall, 2005)

[166] (Bendoly & Swink, 2007)

[167] (Nobandegani et al., 2019)

5.3.1 Biases & Heuristics

Errors and shortcuts in cognition are arguably what separates humans and machines, and the development of cognitive systems needs recognition of these invariable miscues. There are varied reasons as to why humans need to take shorter cognitive routes to achieve behavioral goals. Gigerenzer tackles the misconceptions of heuristics by scoping it into the concept of *ecological rationality*, which matches cognitive processes to environmental structures and computationally analyzes its results.[168] This form of CM takes a Darwinian approach at why humans need to quickly make decisions and come to conclusions about objects in their environment; the main idea lies in the claim that heuristics are not in fact cognitive limitations, but adaptations, and the proper selection of a heuristic in any scenario can be computationally modeled and therefore studied. One study has looked at how different cognitive models themselves are biased in modeling decision-making;[93] although it was inconclusive, the study provided groundwork for the concept that multiple cognitive models can simulate behavioral biases in different contexts.

As an alternative take, Flyvbjerg strongly asserted that it's a misconception to label all biases as *cognitive*. In their study on biases in PM, they highlight the top ten behavioral biases affecting project managers' decisions.[169] The study claimed that biases in PM are overly political; that is, the most prevalent bias as claimed by the study was strategic misrepresentation, also known as *political bias* or *power bias*. This bias is represented by humans deliberately or systematically misrepresenting information as a strategy for their own perceived power. What this highlights is an important aspect of behavioral economics, that decision-making is not purely attached to rational or mechanistic cognitive processes, as RUT might proclaim;[137] it varies based on individuals' motivations and perceptions of the decision's outcome.

The other top biases in PM identified by Flyvbjerg were backed by other studies: Griffiths et al. found that anchoring and adjustment are common in software estimation and can severely alter the estimations made regardless of estimation technique;[72] Mohanani et al. found that anchoring and adjustment, availability, and confirmation bias were the most prevalent in current SE research;[9] Cunha reviewed the most common biases in SPM, finding that planning fallacy was related to an overoptimism to meet stakeholders' expectations combined with a lack of knowledge in business and/or technology;[136] and further qualitative research has highlighted the behavioral biases in PM, claiming that 60% of the time project managers' status reports were biased, with an overwhelming majority of status reports being optimistic

[168] (Gigerenzer, 2008)
[169] (Flyvbjerg, 2021)

as opposed to pessimistic.[170] These claims were again made in a review of the behavioral biases common to decision-making and risk analysis.[70] Researchers in this study also suggested ways of debiasing decision-making processes, for example by adjusting the weights of initial decisions to avoid anchoring bias, that can be applied to SPM decisions. When considering biases and heuristics as cognitive preconditions, one must also account for a software project managers' experience level as it influences the volume and variety of cognitive mechanisms enacted in decisions.

5.3.2 Domain Experience

In their SLR, Cunha et al. apply the naturalistic perspective of decision-making in SPM to the research scope.[135] Under the naturalistic point of view, instead of generating 'option sets' that span a space of possible behaviors, humans apply their domain experience to make judgements; and in this way they commit to a decision while anticipating a set of possible outcomes that could result from it. This highlights the importance of including project managers' experience level into their decision-making capabilities. Another study that used CM to assess the usability of touch screen UIs remarked on the necessity to include expertise into cognitive models, as skilled users are understood to perform tasks without an abundance of cognitive processing.[95] Similarly, another study showed how knowledge acts as a mitigating factor between project risk and developer performance in IS projects.[171] Regardless of to what extent knowledge flows throughout a project's life cycle – especially considering the perspective of naturalistic decision making – certain preconditions for models exist based on the dynamic behavior that is created by the decision space.

5.4 Modeling SPM

Revisiting the basic SPM model (Figure 13) shows how the researchers amended the model structure to include feedback loops within the project environment. While this model effectively encapsulates how feedback between activities in SPM changes the weight of project variables, one study has argued that the same model does not allow project managers to exact uncertainty in project activities, where Monte Carlo simulations were implemented as an additional step.[172]

---

[170] (Snow et al., 2007)

[171] (Venkatesh et al., 2018)

[172] (Barros et al., 2000)

Additional research applying system dynamics models to SPM has been conducted. In one study, aimed at assisting project managers in reducing uncertainty of projects, researchers developed a model to predict project complexity and more precisely control resource allocation.[173] The researcher in another study provided a software process simulation model and proposed to combine both system dynamics and discrete event models;[103] however, the model proposed does not adapt to SPM processes as it is mainly concerned with iterations of the SDLC and not the management aspects.

Sukhodolsky's model takes a step back from the others by seeing the project control process as a vital piece of SPM that can be modeled and optimized (Figure 14). While the advantage of this model is that it formulates the control process as a discrete optimization problem; it does not fully encapsulate the intricacies of SPM from a cognitive perspective. Rather, the solution to problems faced at the control stage are, based on this model, fully dependent on the decisions of the project manager in Stage IV of the process. While this is a realistic approach, it does not make for an applicable model that can accurately predict project parameters.

Researchers proposed an integrated framework that mixes software process simulation methods with project knowledge to improve SPM processes.[174] The benefit of this model is that it was meant for practicing project managers to use and understand how to improve their processes, and because different software development methodologies can be implemented in the simulation model. However, it acts more as a process model than a cognitive model of SPM. This may be due to the nature of simulation methods because they often require a mechanistic flow from input to output and complex algorithms must be implemented to account for dynamics.

The idea of adapting processes like planning and execution with inputs that are dynamic and adjusted in real time is valuable to SPM. In this light, Ge and Xu proposed a model of software project scheduling.[132] The proposed model incorporates a team productivity model that supports the project schedule generation process; a process that dynamically schedules and reschedules project tasks based on genetic algorithm optimizations.

5.4.1 Applying Existing PM Models to SPM

Activities within projects are inherently random. Tavares argued that most models of PM take a deterministic point of view to describe project resources and expenses, and therefore proposed a stochastic model that assumes a random nature of project

---

[173] (Chen et al., 2008)

[174] (Baia, 2015)

tasks and risks (Figure 10).[175,176] The stochastic nature of a project network model, like the one highlighted by Tavares, describes how tasks with causal relationships have either direct or inverse effects on surrounding tasks. Furthermore, uncertainty stems from the fact that project activities do not have binary outputs; task completion itself is variably defined – the decision to label a task as completed is entirely subjective – but the goal of project delivery is clearly modeled. The challenge, then, of applying PM models to SPM is that there is arguably more uncertainty involved in software development activities, with varying team dynamics, requirements modifications, and unpredictable technical problems.

A further issue in the PM field is intensified by analysts' desire to build the most complex solutions to PM problems, giving in to the idea that elegance and impressiveness of algorithms equates to being able to solve real-world problems.[177] Initial research into the computational methods of team formation in PM has been completed, which calls for a different type of decision-making to be modeled because certain personnel decisions are not immediately or visibly met with a metric of success or failure.[178] Models of team formation can be effectively applied to SPM models as team formation does not vary substantially between software projects and projects in other fields.

Another challenge of applying PM models to SPM is taking consideration of the flow of knowledge throughout the project life cycle. Snider & Nissen argue that the PMBOK and similar methodologies do not account for the dynamic flow of knowledge within projects; addressing this aspect and applying it to model a software project case study (Figure 11).[105] Their model adds the dimensions of *reach, time,* and *explicitness* to PM models. These are vital dimensions to consider while transforming project flows into vectors; a technique that adds numerous research directions to CM in SPM. Such models can be highly malleable in the inclusion of cognitive processes when modeling cognitive mechanisms as well as user behavior.

5.4.2 User Models

Studying humans performing computer-interactive tasks creates an established baseline for CM research. Researchers proposed a framework for integrating task models with cognitive models using Petri net simulations to model the task environment as a network (SPMNet; Figure 15).[129] An important consideration of this model is the manually placed constraints by software project managers into the

---

[175] (Tavares, 1994)
[176] (Tavares, 2002)
[177] (Williams, 2003, p. 22)
[178] (Tsesliv, 2022)

model. Additionally, it is the only model covered in this research scope with a use case for project managers – other models have a learning curve meant for expert cognitive scientists, while SPMNet is meant to be used by managers themselves to optimize their workflow by implementing genetic algorithms to compute optimal resources.

Furthermore, researchers have used computational models of shared workspaces to study the execution time of tasks of individuals within groups with human information-processing models such as GOMS and KLM;[174,179] while others have developed the method of 'action modeling'[41] to predict human behavior in circumstances where a decision needs to be made with alternative actions – using Soar as the main architecture.

Aligning user and cognitive models into a task network has the advantage of using psychological metrics when evaluating task performance. This consideration highlights the implications of incorporating various cognitive factors into SPM models, a notion that finds resonance in process models.

5.4.3 Process Models

The main difference between process models and cognitive models is that typical business processes are inherently deterministic, where one task needs to finish before the next begins. This workflow does not accurately reflect the rational approach that humans take when making decisions and completing tasks in software projects. For example, UML diagrams have been used to model SE and ITSM processes by including ontology-based rules.[2] While this is a convenient way to conceptualize SD processes, as rule-based ontologies closely align with SD practices, there is a lack of cognitive dynamics that would affect tasks within this model when applied to models of SPM.

In one study, researchers use process modeling to evaluate, predict, and improve software processes, using Abdel-Hamid and Madnick's software project model (Figure 13) to simulate the project environment.[180] They used monte carlo simulation techniques to improve upon the original model; but the main argument against the study is the lack of cognitive implications in project parameters. The researchers claim that creators of the original model did not incorporate uncertainty in its feedback system, hence the decision to include monte carlo simulations due to their ability to reduce abstract probabilistic computations to rational ones.[181]

---

179 (Ferreira et al., 2009)

180 (Barros et al., 2000)

181 (Sanborn et al., 2010, p. 1144)

Other research using simulated processes in the SPM domain has examined more closely the *execution* phase of software development and management, but the lack of scope and mechanistic view of project processes does not account for cognitive feedback within project or task variables.[131]

Similar critiques can be made about WFM techniques for SPM. Researchers have applied WFM technology to control business and PM processes.[100,] Their approach was based on the ad hoc nature within the overarching network of tasks, arguing that business processes and PM are related by *tasks,* and these can be monitored and controlled using WFM tools. This idea makes the applicability of WFM models to model software projects high; however, research has failed to bridge this gap, and without considering cognitive preconditions of said tasks, the models do not accurately reflect the complex dynamics of SPM.

5.5 Applying Cognitive Preconditions to Models

One study implemented cognitive complexity into BPM, but the lack of academic attention in the software domain and the inherent complexity of software processes gave researchers inconclusive results.[5] A separate study was undertaken to mesh the gap between cognitive psychology and BPMs, where researchers displayed how chunking, computational offloading, and external memory provide a lower mental effort in the understanding of process models.[44] Although human *understanding* of models is only the tip of the iceberg in this SLR – the importance of the models being able to accurately predict human behavior would be a sharper focus – the latter study raised an important aspect of the current state of modeling research. That is, incorporating human memory models into user, task, process, and behavioral models is essential to understanding and predicting human error in behavior from a cognitive perspective. One researcher suggested this, while also lobbying for the development of cognitive user models because of their estimation power of the dynamic effects of human performance.[129]

Further attempts have been made to introduce cognitive aspects into process models,[182,183] but more research remains on how to manage errors within such a system. For example, Figure 20 shows a proposed cognitive process management model. Although still a conceptual model, the disadvantage of this proposed system lies in the fact that it is a 'black box'; if an error occurs anywhere within the process, there

---

[182] (Kriegeskorte & Douglas, 2018)
[183] (Gonçalves et al., 2023)

is a heavy load of cognitive and manual effort that a manager must take to resolve or even understand the nature of the error using this system.

Fuzzy Cognitive Maps

It is possible to use FCMs to model dynamic behavior of software project managers while considering cognition. The comparison of FCMs to cognitive models is in the fuzzy logic system of FCMs which is considerate of certain cognitive dynamics.

For example, the model that Tlili et al. proposed (Figure 16) only examines the singular aspects of *risk* in SPM. If similar attention was expended in each of the processes in SPM, and those were collectively analyzed and compared, it would provide a more comprehensive view of the interactions inside projects. Comparatively, the model proposed by Chernova et al. (Figure 17) provides a comprehensive scope that includes the whole software project life cycle. The disadvantage is that it does not focus in-depth on the relationships between the processes as the former model, although it does provide insights into the conditional aspects of these relationships. Combining these models – by adapting every node of Figure 17 to have the same depth as Figure 16 – researchers could effectively visualize all aspects of SPM while further accounting for cognitive preconditions.

Others have recognized FCMs to be a useful method for representing human cognition. In their review of cognitively inspired methods of modeling decision-making processes, researchers identified developments of FCMs in multi-attribute decision-making (MADM) research.[184] They argue that fuzzy sets can properly manage the complexity and uncertainty involved in MADM and human thinking. Adapting computationally complex modeling methods like FCMs is not without its drawbacks, however, because increasing the modeling complexity transmutes symbolic graphical models into computational models that require more precise metrics for assessing their accuracy. When critiquing certain methods for being black boxes, FCMs and neural networks are along the same lines, where analysts must rely on error metrics to establish measures of accuracy and efficiency of these models.

5.6 Model Accuracy

Choosing one model over another is an important subject of focus in current research.[86,95,143] Generally, the ability of a model to describe mental processes accurately is measured using the goodness of fit (GOF) metric, which describes various

---

[184] (Wu & Xu, 2022)

errors in the model's ability to explain variance.[185] GOF is important for model selection because it has the capability to explain cognitive processes as they occurred in the data, accounts for the most amount of variance, and is easy to calculate,[186] but there is an argument against relying on this metric alone to account for model accuracy: researchers in one study proposed a method of selection called *minimum description length* – where generalizability takes priority above GOF – and concludes that for a model to generalize properly to new data it needs to be simplistic enough that complex data does not lead to overfitting.[187] This suggests that generalizability can be used to supplement GOF as an additional measure of accuracy. Although generalizability can be defined as a computable function that measures expected error of model predictions,[188] using it as an accuracy metric adds to the idea that applying computational models is outside the realm of skills that a normal, non-expert researcher, analyst, or project manager should wield.

To further improve model accuracy measures, researchers have ventured into 'geometrically modeling complexity'[189], which could prove to be highly useful when modeling complex and uncertain environments like software projects. Complexity is a tradeoff analysts must consider when using GOF as a measure of accuracy, much like the two edges of Occam's razor: cognitive models need enough complexity to encapsulate underlying processes yet must be simplistic enough to be generalizable and avoid overfitting.

As a final note to the discussion, the choice of model, modeling concept, framework, or theory bears no weight if the initial research question asked in not suited for a CM task. Shürmann and Beckerle proposed a conceptual framework for this purpose.[19] Using their framework, researchers could formally answer the question 'is my research question fit for modeling?', and subsequently develop and refine their proposed model until they reach a stable starting point. Furthermore, Lee et al. argued that increasing the robustness of computational modeling in cognitive science advances the reliability of empirical psychological findings.[78] They proposed that researchers should provide detailed reports of their modeling decisions, pre- and

---

[185] (Shepard, 1991, p. 53)

[186] GOF is calculated using common metrics such as root mean squared error (RMSE), the square root of the average of all deviations between real and predicted data instances, and maximum likelihood, the probability of observing the model's data instances maximized in respect to its parameters.

[187] (Pitt et al., 2002, p. 473)

[188] (Linhart & Zucchini, 1986)

[189] (Hiatt et al., 2022)

post-study. What they are arguing for would add confidence to the findings generated in cognitive and behavioral science and would increment theoretical knowledge. It becomes evident that the success of CM centers not only on the choice of appropriate theoretical frameworks but also on the alignment of research questions, consequently reinforcing the relationship between theoretical precision and experimental robustness.

5.7 Synopsis

The SLR and its outcomes highlighted advantageous approaches to modeling SPM that provide a baseline understanding of the existing and potential relationship between the fields of cognitive/behavioral science and SPM, and how CM can be a worthwhile technique for expanding the knowledge in both cognitive theories and PM. The findings herein pinpoint a clear direction for future studies that further measure the shared characteristics between these bodies of research. In completion of the study, the main objectives can be answered as follows.

Scope and reproducibility of cognition using CM
With cognitive architectures such as ACT-R and Soar, cognitive processes of decision-making, knowledge acquisition, memory, attention, planning, reasoning, and intelligence can be deduced into the computational level using various theoretical approaches. Therefore, cognitive processes are reproducible using CM techniques; but the level of reproducibility and the scope of what cognitive processes can be applied is determined by the individual researcher's intentions and decision rules.

Replicability of computer-interactive human behavior
CM techniques can replicate behavior resulting from the interaction of humans with technology by eliciting cognitive steps in the interaction between motor and sensory experiences. Models such as GOMS and KLM use the MHP framework to replicate human behavior to a level that allows precise measure at different levels of analysis.

Preconditions of applying CM to SPM
Cognitive preconditions exist that affect the reproducibility of many SPM behaviors. Cognitive and behavioral biases are one of the preconditions that require consideration when developing cognitive models, as they have been identified in the literature to severely affect PM activities. Another precondition for modeling is the project manager's domain experience, as it correlates to the level of cognitive processing needed to make decisions.

Pre-existing models of PM

Of the existing models of PM, there are models that focus specifically on one aspect of PM, whether it is planning or risk management, and there are models that focus on the dynamics of interaction between PM activities in the complete project life cycle.

Thresholds for modeling SPM

The thresholds that exist when using CM to model SPM are entirely dependent on the cognitive theories used by the modeling researcher, along with any limitations of those theories, or if there are thresholds specifically placed by the researcher to narrow their study scope or to otherwise invoke some level of control over their research. Behavioral thresholds are also dependent on the individual researcher's decisions; but as SPM is constrained to a specific set of behaviors, the thresholds are less theoretical and more empirical by nature.

Accurate vs. inaccurate models of SPM

The accuracy of how well CM replicates certain cognitive mechanisms is dependent on the metrics used to measure both the model's fit to human behavior and the intentions of the researcher. The GOF metric is the main method used to assess accuracy of computational models, which uses common error measures such as RMSE and maximum likelihood to observe how accurately models fit predictions to observed behavioral data.

6 Conclusion

The SLR and its relevant analyses has highlighted the existing research in the fields of CM, SPM, HCI, and cognitive science. In doing so, it has also brought to the discussion table the gaps in research between these research domains. While there are numerous complexities involved in modeling cognitive processes and applying theoretical approaches to SPM, this study has shown the emergence of literature that has begun to synthesize these topics into a homogeneous group and reduce research scope complexity.

The most significant discrepancy observed in this study is in the semantic definition of *cognitive modeling*. Cognition can be modeled (conceptually) using symbolic methods, and it can also be modeled (mathematically) using algorithmic components; however, the observation after scanning the literature is that the difference in mathematical, computational, and conceptual or symbolic, is convoluted and at times these levels are used interchangeably. This misuse of terminology is not present in every study – there are several that clearly define and scope either computational or symbolic CM into their research – but the overall observation is that these terms are often used together to define *cognitive modeling* as both mathematical and conceptually symbolic.

Another observation is the lack of distinguishment, at times, between *process* and *cognitive* models. Process models are innately separate from cognitive models due in large part to the synchronous initiation and completion of project activities, and these two cannot be combined in the same definition. Although certain process models, as highlighted through this SLR, can contain cognitive elements (emphasized in the next section on future research directions), cognitive mechanisms do not have the same synchronous flow as tasks in industrial or mechanical environments do. Furthermore, software project activities are in themselves not restricted to sequential processes, and although researchers have used process models to describe SE projects, they cannot completely account for cognition as interpreted.

Finally, there exists a wide body of research and practice in different PM methodologies – waterfall, agile, lean, etc. – that are underrepresented in the literature encompassing this SLR. Agile development, one methodology that is dominating the mainstream project practices in software development, is not comprehensively modeled using CM techniques. While this can be seen as a research limitation, that agile practices not being included in the research scope is the cause of the research gaps, there is a more concise argument that even with the focus of different PM methodologies a similar amount of literature would be discovered.

6.1 Critical Reflection

Some automatic limitations occur when adjusting the scope to fit within this study's resource constraints. First, when limiting SPM behavior to be only the computer-interactive behavior (such as monitoring project progress from a visual dashboard or interacting with colleagues online), there is a complete social and environmental component that is missing. Some researchers have argued that this component is more essential than the cognitive processes themselves.[Error! Bookmark not defined.] The interaction between colleagues on virtual platforms provides sufficient social activity to observe the effects on human performance in the domain of SPM; however, this requires focused attention and research. Second, there is room for the inclusion of both internal and external aspects of managing software projects: at the individual level, there is vast potential to research the motives, feelings, and personality traits of managers, while on the interpersonal level, there is a social dynamic between individuals that can be theorized to moderate cognitive mechanisms. Although the research scope requires psychological theories that are inclusive of all these elements, the goal of focusing research on cognitive science is precisely to narrow the *cognitive* aspects of phenomena.

Other limitations of the study can be criticized from a technical standpoint. Specifically, the choice of algorithm used when clustering documents for analysis after SLR inclusion is such a limitation. The Lingo3G algorithm was utilized because it comes pre-packaged with EPPI-Reviewer software, but its parameters were not manipulated or otherwise optimized. The intention was to provide a benchmark analysis to directly observe the effects of document classification; thus, the results of this study should be interpreted from this perspective. Future studies have the opportunity of expanding on this work by more precisely fine-tuning the clustering model's hyperparameters, and by testing different clustering algorithms and analyzing the results.

6.2 Recommendations for Future Research

One study called for more research in computational models of interaction, under the lens that individuals' social lives are often determined by complex and unconscious cognitive processing, and the current analysis techniques of human behavior do not account for the meaning that individuals construct from social interactions.[190] The social aspects of PM add another level to the environmental aspects of cognition. Including this in future research on CM could provide a supplementation to

[190] (Vinciarelli et al., 2015)

cognitive theory. When researching models of attention, researchers identified need for studies that more closely examine the variation in time between tasks with different levels of demand in interactive and complex environments.[191] Perception is an under-studied concept in SE research although many processes include key perceptual mechanisms involving the visual field, such as text comprehension (i.e., reading code), and summarizing large amounts of information using visuals.[192]

In PM research, there is limited literature addressing the combination of rational and intuitive management processes.[193] One study introduced the idea of using a metric of entropy to evaluate and structure the effects of cognition on PM.[7] Another study highlights the lack of integrated approaches that are used in SPM-related simulations, arguing that software project knowledge, software development processes, and project-related awareness and learning are not optimally merged when leveraged to study PM decision-making.[174] In the software context, there is an identified gap in research that links PM and cognitive biases.[136]

Formal literature on how to build cognitive process models, and systematic reviews highlighting the characteristics of what defines a 'process model', would be highly advantageous to the scientific community. Additionally, cognitive user models as simulation tools have been mentioned as a future step in modeling research.[129] Connecting process models to probabilistic models is a potential way of modeling the mind.[60] Additionally, modeling psychological phenomena from a probabilistic inference perspective is a prospective field of research.[73]

6.3 Outlook

While the study of cognition is dynamically complex, supporting research of the different theoretical approaches is necessary to progress the field of cognitive science. By adapting CM techniques and computational models into formal PM and SPM models, the gap between research domains would be fulfilled in methodical and measurable ways. Finally, as human-computer interactive behavior can be used as a theoretical lens to observe these domains, supplementary research that combines cognitive elements within behavioral models is required to enhance the incorporation of all factors into academically sound theories.

[191] (Borji & Itti, 2012)

[192] (Fagerholm et al., 2022)

[193] (Fossum et al., 2020)

References

Abdel-Hamid, T. K., & Madnick, S. E. (1989). Lessons learned from modeling the dynamics of software development. *Communications of the ACM*, *32*(12), 1426–1438. https://doi.org/10.1145/76380.76383

Afacan-Seref, K., Steinemann, N. A., Blangero, A., & Kelly, S. P. (2018). Dynamic Interplay of Value and Sensory Information in High-Speed Decision Making. *Current Biology*, *28*(5), 795–802. https://doi.org/10.1016/j.cub.2018.01.071

Almomani, A., Monreal, C., Sieira, J., Graña, J., & Sánchez, E. (2021). Rational, emotional, and attentional models for recommender systems. *Expert Systems*, *38*(4).

Anderson, J. R. (1990). *The Adaptive Character of Thought*. Psychology Press. https://doi.org/10.4324/9780203771730

Anderson, J. R. (1996). ACT: A simple theory of complex cognition. *American Psychologist*, *51*(4), 355.

Andrade, P. R. M., Albuquerque, A. B., Frota, O. F., & da Silva, F. J. F. (2015). *PM5: One approach to the management of IT projects applied in the Brazilian public sector*. SERP'15 - The 13th International Conference on Software Engineering Research and Practice.

Anu, V., Hu, W., Carver, J. C., Walia, G. S., & Bradshaw, G. (2018). Development of a human error taxonomy for software requirements: A systematic literature review. *Information and Software Technology*, *103*, 112–124. https://doi.org/10.1016/j.infsof.2018.06.011

Aranda, J., & Easterbrook, S. (2005). *Anchoring and Adjustment in Software Estimation*. 346–355.

Bagherzadehkhorasani, A., & Tehranchi, F. (2022). *Comparing Cognitive, Cognitive Instance-Based, and Reinforcement Learning Models in an Interactive Task*. Proceedings of the 20th International Conference on Cognitive Modelling (ICCM 2022).

Baia, D. (2015). An Integrated Multi-Agent-Based Simulation Approach to Support Software Project Management. *2015 IEEE/ACM 37th IEEE International Conference on Software Engineering*, 911–914. https://doi.org/10.1109/ICSE.2015.290

Barros, M., Werner, C., & Travassos, G. (2000). *Using Process Modeling and Dynamic Simulation to Support Software Process Quality Management*. XIV Brazilian Symposium on Software Engineering-Software Quality Workshop.

Battersby, A., & Carruthers, J. A. (1967). Critical Path Methods. *Journal of the Operational Research Society*, *18*(4), 469–470.

Beimel, D., & Kedmi-Shahar, E. (2019). Improving the identification of functional system requirements when novice analysts create use case diagrams: The benefits of applying conceptual mental models. *Requirements Engineering*, *24*, 483–502.

Bendoly, E., & Swink, M. (2007). Moderating effects of information access on project management behavior, performance and perceptions. *Journal of Operations Management*, *25*(3), 604–622. https://doi.org/10.1016/j.jom.2006.02.009

Borji, A., & Itti, I. (2012). State-of-the-art in visual attention modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(1), 185–207.

Brereton, P., Kitchenham, B. A., Budgen, D., Turner, M., & Khalil, M. (2007). Lessons from applying the systematic literature review process within the software engineering domain. *Journal of Systems and Software*, *80*(4), 571–583. https://doi.org/10.1016/j.jss.2006.07.009

Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human-computer interaction*. L. Erlbaum Associates.

Case, D. M., & Stylios, C. D. (2016). *Fuzzy Cognitive Map to model project management problems*. 1–6. https://doi.org/10.1109/NAFIPS.2016.7851612

Chanceaux, M., Baccino, T., Guérin-Dugué, A., & Lemaire, B. (2014). A Computational Cognitive Model of Information Search in Textual Materials. *Cognitive Computation*. https://doi.org/10.1007/s12559-012-9200-1

Chang, C. K., Chao, C., Nguyen, T. T., & Christensen, M. (1998). *Software Project Management Net: A new methodology on software management*. 534–539. https://doi.org/10.1109/CMPSAC.1998.716715

Chater, N., Oaksford, M., Hahn, U., & Heit, E. (2010). Bayesian models of cognition. *Wiley Interdisciplinary Reviews: Cognitive Science*, *1*(6), 811–823.

Chen, C. L., Wang, C. S., Lee, Y. H., & Lee, Y. M. (2008). *A Process-Oriented System Dynamics Model for Software Development Project Prediction*. *2*, 126–131. https://doi.org/10.1109/NCM.2008.174

Chernova, L., Zhuravel, A., Chernov, S., Kunanets, N., & Artemenko, O. (2022). *Application of the Cognitive Approach for IT Project Management and Implementation*. 426–429. https://doi.org/10.1109/CSIT56902.2022.10000512

Colomo-Palacios, R., Casado-Lumbreras, C., Soto-Acosta, P., & García-Crespo, A. (2013). Decisions in software development projects management. An exploratory study. *Behaviour & Information Technology*, *32*(11), 1077–1085.

Cunha, J. A. O., & Moura, H. P. (2015). *Towards a substantive theory of project decisions in software development project-based organizations: A cross-case analysis of IT organizations from Brazil and Portugal*. 1–6. https://doi.org/10.1109/CISTI.2015.7170515

Cunha, J. A. O., Moura, H. P., & Vasconcellos, F. J. (2016). Decision-making in Software Project Management: A Systematic Literature Review. *Procedia Computer Science*, *100*, 947–954. https://doi.org/10.1016/j.procs.2016.09.255

Emond, B., & West, R. L. (2003). Cyberpsychology: A Human-Interaction Perspective Based on Cognitive Modeling. *CyberPsychology & Behavior*, *6*(5), 527–536. https://doi.org/10.1089/109493103769710550

Fagerholm, F., Felderer, M., Fucci, D., Unterkalmsteiner, M., Marculescu, B., Martini, M., & Khattak, J. (2022). Cognition in Software Engineering: A Taxonomy and Survey of a Half-Century of Research. *ACM Computing Surveys*, *54*(11s), 1–36.

Fan, X., McNeese, M., & Yen, J. (2010). NDM-Based Cognitive Agents for Supporting Decision-Making Teams. *Human-Computer Interaction*, *25*(3), 195–234.

Fechner, H. B., Schooler, L. J., & Pachur, T. (2015). *Quantifying Simplicity: How to Measure Sub-Processes and Bottlenecks of Decision Strategies Using a Cognitive Architecture*. 82–83.

Fernandes, R. G., & Vils, L. (2022). *Towards of Cognition in Project Management*. XLVI Encontro da ANPAD - EnANPAD 2022.

Ferreira, A., Antunes, P., & Pino, J. A. (2009). Evaluating shared workspace performance using human information processing models. *Information Research*, *14*(1).

Fitts, P. M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, *47*(6), 381–391. https://doi.org/doi:10.1037/h0055392

Fleischmann, M., Amirpur, A., Benlian, A., & Hess, T. (2014). COGNITIVE BIASES IN INFORMATION SYSTEMS RESEARCH: A SCIENTOMETRIC ANALYSIS. *Tel Aviv*.

Flyvbjerg, B. (2021). Top Ten Behavioral Biases in Project Management: An Overview. *Project Management Journal*, *52*(6), 531–546. https://doi.org/10.1177/87569728211049046

Forrester, J. W. (1997). Industrial dynamics. *Journal of the Operational Research Society*, *48*(10), 1037–1041.

Fossum, K. R., Aarseth, W., & Andersen, B. (2020). Exploring scenario development – a case study of two collaborative research projects. *International Journal of Managing Projects in Business*, *13*(2), 340–366.

Fu, W. T., & Pirolli, P. (2007). SNIF-ACT: A cognitive model of user navigation on the World Wide Web. *Human–Computer Interaction*, *22*(4), 355–412.

Ge, Y., & Xu, B. (2016). Dynamic Staffing and Rescheduling in Software Project Management: A Hybrid Approach. *PLOS ONE*, *11*(6), e0157104. https://doi.org/10.1371/journal.pone.0157104

Gigerenzer, G. (2008). Why heuristics work. *Perspectives on Psychological Science*, *3*(1), 20–29.

Gigerenzer, G., & Brighton, H. (2009). Homo Heuristicus: Why Biased Minds Make Better Inferences. *Topics in Cognitive Science*, *1*(1), 107–143. https://doi.org/10.1111/j.1756-8765.2008.01006.x

Gonçalves, J. C., Baião, F. A., Santoro, F. M., & Guizzardi, G. (2023). A cognitive BPM theory for knowledge-intensive processes. *Business Process Management Journal*, *29*(2), 465–488. https://doi.org/10.1108/BPMJ-11-2021-0746

Gray, W. D. (2008). Cognitive Architectures: Choreographing the Dance of Mental Operations With the Task Environment. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, *50*(3), 497–505. https://doi.org/10.1518/001872008X312224

Griffiths, T. L., Chater, N., Kemp, C., Perfors, A., & Tenenbaum, J. B. (2010). Probabilistic models of cognition: Exploring representations and inductive biases. *Trends in Cognitive Sciences*, *14*(8), 357–364.

Griffiths, T. L., Vul, E., & Sanborn, A. N. (2012). Bridging Levels of Analysis for Probabilistic Models of Cognition. *Current Directions in Psychological Science*, *21*(4), 263–268.

Gruhn, V., & Laue, R. (2006). *Adopting the cognitive complexity measure for business process models*. *1*, 236–241.

Hackman, D. C. (2021). Quantifying heterogeneous individual perceptions in project management research. *International Journal of Managing Projects in Business*.

Heloisa, M. S., & Tseng, M. M. (1996). Workflow technology-based monitoring and control for business process and project management. *International Journal of Project Management*, *14*(6), 373–378. https://doi.org/10.1016/0263-7863(96)00026-9

Hiatt, L. M., Brooks, C., & Trafton, J. G. (2022). Validating and Refining Cognitive Process Models Using Probabilistic Graphical Models. *Topics in Cognitive Science*, *14*(4), 873–888. https://doi.org/10.1111/tops.12616

Howes, A., Lewis, R. L., & Vera, A. (2009). Rational adaptation under task and processing constraints: Implications for testing theories of cognition and action. *Psychological Review*, *116*(4), 717–751. https://doi.org/10.1037/a0017187

Jarecki, J. B., Tan, J. H., & Jenny, M. A. (2020). A framework for building cognitive process models. *Psychonomic Bulletin & Review*, *27*(6), 1218–1229. https://doi.org/10.3758/s13423-020-01747-2

Jiménez, J. P., Martin, L., Dounce, I. A., Ávila-Contreras, C., & Ramos, F. (2021). Methodological aspects for cognitive architectures construction: A study and proposal. *Artificial Intelligence Review*, *54*(3), 2133–2192. https://doi.org/10.1007/s10462-020-09901-x

Jiminez, R. C., Lee, T., Rosillo, N., Cordova, R., Cree, I., Gonzalez, A., & Indave Ruiz, B. I. (2022). Machine learning computational tools to assist the performance of systematic reviews: A mapping review. *BMC Medical Research Methodology*, *22*(1), 1–14.

John, B. E., & Kieras, D. E. (1994). *The GOMS family of analysis techniques: Tools for design and evaluation*.

Kahneman, D. (2017). *Thinking, fast and slow.*

Kahneman, D., & Tversky, A. (1979). Prospect Theory: An Analysis of Decision under Risk. *Econometrica*, *47*(2), 263–291. https://doi.org/10.2307/1914185

Kennedy, W. G., & Bugajska, M. (2010). *Integrating Fast and Slow Cognitive Processes*. International Conference on Cognitive Modeling. http://doi.apa.org/get-pe-doi.cfm?doi=10.1037/e660352010-001

Kennedy, W. G., & Patterson, R. E. (2012). *Modeling Intuitive Decision Making in ACT-R*. *12*, 15.

Kieras, D. E., Hornof, A., & Zhang, Y. (2015). *Visual Search of Displays of Many Objects: Modeling Detailed Eye Movement Effects with Improved EPIC*. *55*.

Kieras, D. E., & Meyer, D. E. (1997). An Overview of the EPIC Architecture for Cognition and Performance with Application to Human-Computer Interaction. *Human–Computer Interaction*, *12*(4), 391–438.

Knauff, M., & Wolf, A. G. (2010). Complex cognition: The science of human reasoning, problem-solving, and decision-making. *Cognitive Processing*, *11*(2), 99–102. https://doi.org/10.1007/s10339-010-0362-z

Koch, J., Drazic, I., & Schermuly, C. C. (2023). The affective, behavioural and cognitive outcomes of agile project management: A preliminary meta-analysis. *Journal of Occupational and Organizational Psychology*, joop.12429. https://doi.org/10.1111/joop.12429

Kontogiannis, T. (2005). Integration of task networks and cognitive user models using coloured Petri nets and its application to job design for safety and productivity. *Cognition, Technology & Work*. https://doi.org/10.1007/s10111-005-0010-z

Kotseruba, I., & Tsotsos, J. K. (2020). 40 years of cognitive architectures: Core cognitive abilities and practical applications. *Artificial Intelligence Review*, *53*(1), 17–94. https://doi.org/10.1007/s10462-018-9646-y

Kouskouras, K. G., & Georgiou, A. C. (2007). A discrete event simulation model in the case of managing a software project. *European Journal of Operational Research*, *181*(1), 374–389. https://doi.org/10.1016/j.ejor.2006.05.031

Kriegeskorte, N., & Douglas, P. K. (2018). Cognitive computational neuroscience. *Nature Neuroscience*, *21*(9), 1148–1160. https://doi.org/10.1038/s41593-018-0210-5

Kutsch, E., & Hall, M. (2005). Intervening conditions on the management of project risk: Dealing with uncertainty in information technology projects. *International Journal of Project Management*, *23*(8), 591–599. https://doi.org/10.1016/j.ijproman.2005.06.009

Kyllingsbæk, S. (2006). Modeling visual attention. *Behavior Research Methods*, *38*(1), 123–133. https://doi.org/10.3758/BF03192757

Laird, J. E. (2008). Extending the Soar cognitive architecture. *Proceeding of the 2008 Conference on Artificial General Intelligence*.

Laird, J. E., Xu, J. Z., & Wintermute, S. (2010). *Using Diverse Cognitive Mechanisms for Action Modeling*. Proceedings of the tenth international conference on cognitive modeling.

Lakey, P. B. (2003, May). *A Hybrid Software Process Simulation Model for Project Management*. International Workshop on Software Process Simulation and Modeling (ProSim'03).

Lee, M. D. (2018). Bayesian Methods in Cognitive Modeling. In J. T. Wixted (Ed.), *Stevens' Handbook of Experimental Psychology and Cognitive Neuroscience* (pp. 1–48). John Wiley & Sons, Inc. https://onlinelibrary.wiley.com/doi/10.1002/9781119170174.epcn502

Lee, M. D., Criss, A. H., Devezer, B., Donkin, C., Etz, A., Leite, F. P., Matzke, D., Rouder, J. N., Trueblood, J. S., White, C. N., & Vandekerckhove, J. (2019). Robust Modeling in Cognitive Science. *Computational Brain & Behavior*, *2*(3–4), 141–153. https://doi.org/10.1007/s42113-019-00029-y

Linhart, H., & Zucchini, W. (1986). *Model selection*. Wiley.

Liu, Y. (2009). QN-ACES: Integrating queueing network and ACT-R, CAPS, EPIC, and Soar architectures for multitask cognitive modeling. *International Journal of Human-Computer Interaction*, *25*(6), 554–581.

Logan, G. D. (1988). Toward an instance theory of automatization. *Psychol Rev.*, *95*(4), 492–527.

Mair, C., Martincova, M., & Shepperd, M. (2012). *An Empirical Study of Software Project Managers Using a Case-Based Reasoner*. 1030–1039. https://doi.org/10.1109/HICSS.2012.96

Mair, C., Martincova, M., & Shepperd, M. (2009). *A Literature Review of Expert Problem Solving using Analogy*. 13th International Conference on Evaluation and Assessment in Software Engineering (EASE). https://doi.org/10.14236/ewic/EASE2009.13

Marr, D. (1982). *Vision: A computational investigation into the human representation and processing of visual information.* W.H. Freeman.

Masoud, M., Abu-Elhaija, W., Jaradat, Y., Jannoud, I., & Dabbour, L. (2018). Software Project Management: Resources Prediction and Estimation Utilizing Unsupervised Machine Learning Algorithm. In S. Şahin (Ed.), *8th International Conference on Engineering, Project, and Product Management (EPPM 2017)* (pp. 151–159). Springer International Publishing. https://doi.org/10.1007/978-3-319-74123-9_16

McClelland, J. L., Botvinick, Noelle, D. C., Plaut D. C., Rogers, T. T., Seidenberg, M. S., & Smith, L. B. (2010). Letting structure emerge: Connectionist and dynamical systems approaches to cognition. *Trends in Cognitive Sciences*, *14*(8), 348–356.

Medeiros, B. D. (2015). *An Integrated Multi-Agent-Based Simulation Approach to Support Software Project Management*. *2*, 911–914. https://doi.org/10.1109/ICSE.2015.290

Mohanani, R., Salman, I., Turhan, B., Rodríguez, P., & Ralph, P. (2018). Cognitive biases in software engineering: A systematic mapping study. *IEEE Transactions on Software Engineering*, *46*(12), 1318–1339.

Montibeller, G., & Von Winterfeldt, D. (2015). Cognitive and Motivational Biases in Decision and Risk Analysis. *Risk Analysis: An International Journal*, *35*(7), 1230–1251.

Morita, J., Nagashima, K., & Takeuchi, Y. (2019). *Method of Development of Interactive Agents Grounding the Cognitive Model to the Virtual World*. Proceedings of the 17th International Conference on Cognitive Modelling (ICCM 2019).

Newell, A. (1994). *Unified theories of cognition*. Harvard University Press.

Newell, A., & Simon, H. A. (1972). *Human problem solving* (Vol. 104). Prentice-hall.

Nguyen, T. N. (2006). A decision model for managing software development projects. *Information & Management*, *43*(1), 63–75. https://doi.org/10.1016/j.im.2005.01.006

Nitin, R. C., & Saini, J. R. (2022). *Robotic Process Automation for Software Project Management*. 1–5. https://doi.org/10.1109/I2CT54291.2022.9823972

Nobandegani, A. S., Campoli, W., & Shultz, T. R. (2019). *Bringing Order to the Cognitive Fallacy Zoo*. Proceedings of the 17th International Conference on Cognitive Modelling (ICCM 2019).

Ocak, N., & Cagiltay, K. (2017). Comparison of Cognitive Modeling and User Performance Analysis for Touch Screen Mobile Interface Design. *International Journal of Human–Computer Interaction*, *33*(8), 633–641. https://doi.org/10.1080/10447318.2016.1274160

Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., Mulrow, C. D., Shamseer, L., Tetzlaff, J. M., Akl, E. A., Brennan, S. E., Chou, R., Glanville, J., Grimshaw, J. M., Hróbjartsson, A., Lalu, M. M., Li, T., Loder, E. W., Mayo-Wilson, E., McDonald, S., … Moher, D. (2021). The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *BMJ*, n71. https://doi.org/10.1136/bmj.n71

Patterson, R. E., Pierce, B. J., Boydstun, A. S., Ramsey, L. M., Shannan, J., Tripp, L., & Bell, H. (2013). Training intuitive decision making in a simulated real-world environment. *Human Factors*, *55*(2), 333–345.

Pitt, M. A., Myung, I. J., & Zhang, S. (2002). Toward a method of selecting among computational models of cognition. *Psychological Review*, *109*(3), 472.

Prezenski, S., Brechmann, A., Wolff, S., & Russwinkel, N. (2017). A Cognitive Modeling Approach to Strategy Formation in Dynamic Decision Making. *Front. Psychol.*, *8:1335*. https://doi.org/10.3389/fpsyg.2017.01335

Project Management Institute. (2017). *A guide to the Project Management Body of Knowledge (PMBOK guide) (6th ed.)*. Project Management Institute.

Reitter, D., & Lebiere, C. (2010). A cognitive model of spatial path-planning. *Computational and Mathematical Organization Theory*, *16*, 220–245.

Riesterer, N., Brand, D., & Ragni, M. (2020). Predictive Modeling of Individual Human Cognition: Upper Bounds and a New Perspective on Performance. *Topics in Cognitive Science*, *12*(3), 960–974. https://doi.org/10.1111/tops.12501

Ritter, F. E., Tehranchi, F., & Oury, J. D. (2019). ACT-R: A cognitive architecture for modeling cognition. *Wiley Interdisciplinary Reviews: Cognitive Science*, *10*(3), e1488.

Rubinstein, J. S., Meyer, D. E., & Evans, J. E. (2001). Executive control of cognitive processes in task switching. *Journal of Experimental Psychology: Human Perception and Performance*, *27*(4), 763.

Sanborn, A. N., Griffiths, T. L., & Navarro, D. J. (2010). Rational approximations to rational models: Alternative algorithms for category learning. *Psychological Review*, *117*(4), 1144–1167. https://doi.org/10.1037/a0020511

Schürmann, T., & Beckerle, P. (2020). Personalizing Human-Agent Interaction Through Cognitive Models. *Frontiers in Psychology*, *11*, 561510. https://doi.org/10.3389/fpsyg.2020.561510

Shemilt, I., Khan, N., Park, S., & Thomas, J. (2016). Use of cost-effectiveness analysis to compare the efficiency of study identification methods in systematic reviews. *Systematic Reviews*, *5*(1), 140. https://doi.org/10.1186/s13643-016-0315-4

Shepard, R. N. (1991). Integrality versus separability of stimulus dimensions: From an early convergence of evidence to a proposed theoretical basis. In J. R. Pomerantz & G. L. Lockhead (Eds.), *The perception of structure: Essays in honor of Wendell R. Garner* (pp. 53–71). American Psychological Association.

Simon, H. A. (1957). *Models of Man*. John Wiley.

Simon, H. A. (2019). *The Sciences of the Artificial, reissue of the third edition with a new introduction by John Laird*. MIT press.

Snider, K. F., & Nissen, M. E. (2003). Beyond the body of knowledge: A knowledge-flow approach to project management theory and practice. *Project Management Journal*, *34*(2), 4–12.

Snow, A. P., Keil, M., & Wallace, L. (2007). The effects of optimistic and pessimistic biasing on software project status reporting. *Information & Management*, *44*(2), 130–141. https://doi.org/10.1016/j.im.2006.10.009

Squire, R. L. (2004). Memory systems of the brain: A brief history and current perspective. *Neurobiology of Learning and Memory*, *82*(3), 171–177.

Sukhodolsky, J. (2001). Optimizing software process control. *ACM SIGSOFT Software Engineering Notes*, *26*(2), 59–63. https://doi.org/10.1145/505776.505791

Sun, R. (2006). Prolegomena to integrating cognitive modeling and social simulation. In *Cognition and multi-agent interaction: From cognitive modeling to social simulation* (pp. 3–26). Cambridge University Press Cambridge.

Sweller, J. (1988). Cognitive load during problem solving: Effects on learning. *Cognitive Science*, *12*(2), 257–285.

Tavares, L. V. (1994). A stochastic model to control project duration and expenditure. *European Journal of Operational Research*, *78*(2), 262–266. https://doi.org/10.1016/0377-2217(94)90388-3

Tavares, L. V. (2002). A review of the contribution of Operational Research to Project Management. *European Journal of Operational Research*, *136*(1), 1–18. https://doi.org/10.1016/S0377-2217(01)00097-2

Tlili, A., & Chikhi, S. (2021). Risks Analyzing and management in Software Project Management Using Fuzzy Cognitive Maps with Reinforcement Learning. *Informatica*, *45*(1). https://doi.org/10.31449/inf.v45i1.3104

Tsesliv, O. (2022). *Economical and Mathematical Methods of Project Management*. 1–4.

Vassena, E., Deraeve, J., & Alexander, W. H. (2019). Task-specific prioritization of reward and effort information: Novel insights from behavior and computational modeling. *Cognitive, Affective, & Behavioral Neuroscience*, *19*(3), 619–636. https://doi.org/10.3758/s13415-018-00685-w

Venkatesh, V., Rai, A., & Maruping, L. M. (2018). Information Systems Projects and Individual Developer Outcomes: Role of Project Managers and Process Control. *Information Systems Research*, *29*(1), 127–148.

Vinciarelli, A., Riccardi, G., Esposito, A., André, E., Chetouani, M., Cohn, J. F., Cristani, M., Fuhrmann, F., Gilmartin, E., Hammal, Z., Heylen, D., Kaiser, R., Koutsombogera, M., Potamianos, P., Renals, S., & Bonin, F. (2015). Open Challenges in Modelling, Analysis and Synthesis of Human Behaviour in Human–Human and Human–Machine Interactions. *Cognitive Computation*. https://doi.org/10.1007/s12559-015-9326-z

Wang, M., & Wang, H. (2006). From process logic to business logic—A cognitive approach to business process management. *Information & Management*, *43*(2), 179–193.

Wang, Y., & Laird, J. E. (2010). *A Computational Model of Functional Category Learning in a Cognitive Architecture*. 271.

Williams, T. (2003). The contribution of mathematical modelling to the practice of project management. *IMA Journal of Management Mathematics*, *14*(1), 3–30. https://doi.org/10.1093/imaman/14.1.3

Wohlin, C. (2014). Guidelines for snowballing in systematic literature studies and a replication in software engineering. *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering*, 1–10.

Wu, H., & Xu, Z. (2022). Cognitively Inspired Multi-attribute Decision-making Methods Under Uncertainty: A State-of-the-art Survey. *Cognitive Computation*. https://doi.org/10.1007/s12559-021-09916-8

Yang, Y. C., & Stocco, A. (2022). *A Model of Motivation and Effort Allocation in the ACT-R Cognitive Architecture*. International Conference on Cognitive Modeling.

Zugal, S. (2013). *Applying cognitive psychology for improving the creation, understanding and maintenance of business process models*. Citeseer.

Zugal, S., Pinggera, J., & Weber, B. (2011). Assessing Process Models with Cognitive Psychology. *Enterprise Modelling and Information Systems Architectures (EMISA 2011)*.

Appendix A. MHP Principles of Operation

Principle	Description	Equation
Variable Perceptual Processor Rate	The cycle time of the Perceptual Processor diverges with stimulus intensity.	τ_p , where $\tau =$ cycle time and $p =$ perceptual processor
Variable Cognitive Processor Rate	Processor cycle time decreases when more effort is needed from task demands or information load.	τ_c , where $\tau =$ cycle time and $c =$ cognitive processor
Fitt's Law	The time it takes to move a hand (T_{pos}) to a target (of size S) which lies a certain distance away (D).	$T_{pos} = I_M \log_2 \frac{D}{S} + 0.5$, where $I_M = 100[70\sim120]$msec/bit
Power Law of Practice	The time it takes to perform a task on the nth iteration (T_n), which is determined by a power law.	$T_n = T_1 n^{-\alpha}$, where $\alpha = 0.4[0.2\sim0.6]$
Uncertainty	Decision time (T) increases with uncertainty about judgments or decisions to be made.	$T = I_C H$, where $H =$ information-theoretic entropy of the decision, and $I_C = 150[0\sim157]$msec/bit
Rationality	Humans act rationally to attain their goals, depending on the task, and action is bounded by limitations in knowledge of the task and information processing ability.	Goals + Task + Operators + Inputs + Knowledge + Process limits → Behavior
Problem Space	The problem space that defines the bounds of how humans rationally solve problems is determined by: 1. A set of states of knowledge, 2. Operators that change one state into another, 3. Constraints that control how operators are applied,	

Principle	Description	Equation
	4. Knowledge that guides the decision of which operator to apply next.	
Recognize-Act Cycle of the Cognitive Processor	Contents in Working Memory initiate the actions they are associated with in Long-Term Memory. Modifies Working Memory on every cycle of the Cognitive Processor.	
Encoding Specificity	The type of encoding operation after perception determines what is stored in Memory. Different retrieval cues are variably effective in affording access to what is stored.	
Discrimination	Memory retrieval difficulty is determined by what is in Memory and the relation to retrieval cues used.	

*data from (Card et al., 1983, p. 27)

Appendix B. SPMNet Formal Definition

A SPMNet graph S is defined as,

$$S = (P, T, C, E),$$

where,

$$P = \{P_{ab}, P_{at}, P_p, P_d\},$$
$$T = \{T_I, T_O, T_{DO}\},$$
$$C = \{C_r, C_c\},$$
$$E = \{EI_I, EI_O, EO_I, EO_O, EDO_I, EDO_O, EDOI_I\},$$

with a set of places (P),

- P_{ab} = abstract activity,
- P_{at} = atomic activity,
- P_p = software product,
- P_d = decision,

a set of constraints (C) associated with each activity,

- C_r = resource constraints,
- C_c = complexity constraints,

a set of transitions (T),

- T_I = input transitions to activity places,
- T_O = output transitions from activity places,
- T_{DO} = output transitions from decision places

and a set of arcs (E),

- EI_I = input arcs to transitions T_I from product places,
- EI_O = output arcs from transitions T_I to activity places,
- EO_I = input arcs to transitions T_O from activity places,
- EO_O = output arcs from transitions T_O to product or decision places,
- EDO_I = input arcs to transitions T_{DO} from decision places
- EDO_O = output arcs from transitions T_{DO} to product places
- $EDOI_I$ = input arcs to transitions T_I from decision places

Appendix C. Cluster Sub-Group Label Abbreviations

Abbreviation	Meaning
AB	Availability Bias
BDM	Behavioral Decision-Making
BPM	Business Process Models
BS	Behavioral Science
CA	Cognitive Architectures
CC	Complex Cognitive
CM	Computational Models
CP	Cognitive Processes
CS	Cognitive Science
DM	Decision Model
DMP	Decision-Making Processes
FCM	Fuzzy Cognitive Maps
HCI	Human-Computer Interaction
IE	Internal and External
IL	Individual Level
IS	Information Systems
IT	Information Technology
ML	Machine Learning
PC	Project Cost
PE	Parameter Estimation
PF	Planning Fallacy
PMC	Probabilistic Models of Cognition
PMR	Project Management Research
PP	Psychological Processes
PR	Project Risk
PS	Problem Solving
RA	Resource Allocation
RL	Reinforcement Learning
SD	Software Development
SDP	Software Development Projects
SE	Software Engineering
SPM	Software Project Management
SSP	Support Software Projects

Abbreviation	Meaning
TE	Task Environment
TP	Theory of Project

Appendix D. Python Source Code

D-1 Data Cleaning

```
import pandas as pd
from bs4 import BeautifulSoup
import requests
import re

# Load the table into a pandas dataframe
df_train = pd.read_csv('../Docs/training_set_cleaned.csv')
df_all = pd.read_csv('../Docs/labels_50239.csv')
df_only_true = pd.read_csv('../Docs/labels_50239__only-true-
predictions.csv')


def fix_empty_author_and_doi(df):
    # Create empty columns for author(s) and DOI
    df['author'] = ''
    df['doi'] = ''

    # Loop through each row in the DataFrame
    for index, row in df.iterrows():
        # Get the title and journal from the current row
        title = row['title']
        journal = row['journal']

        # Construct the query URL
        query = f"{title} {journal} doi"
        url = f"https://www.google.com/search?q={query}"

        # Make the request to the query URL
        response = requests.get(url)

        # Parse the HTML response using BeautifulSoup
        soup = BeautifulSoup(response.content, 'html.parser')

        # Find the first search result link
        link = soup.find('a')
```

```
        if link:
            # Get the URL of the first search result
            url = link.get('href')

            # Check if the URL contains "doi.org/"
            if "doi.org/" in url:
                # Extract the DOI from the URL
                doi = url.split("doi.org/")[1]

                # Update the DOI column in the DataFrame
                df.at[index, 'doi'] = doi

                # Make a request to the DOI URL
                doi_url = f"https://doi.org/{doi}"
                doi_response = requests.get(doi_url)

                # Parse the HTML response using BeautifulSoup
                doi_soup = BeautifulSoup(doi_response.content,
'html.parser')

                # Find the author(s) information
                author_tags = doi_soup.find_all('meta', attrs={'name':
'citation_author'})

                # Extract the author(s) information
                authors = [tag.get('content') for tag in author_tags]

                # Join the author(s) into a single string
                author_string = ', '.join(authors)

                # Update the author column in the DataFrame
                df.at[index, 'author'] = author_string

    # Save the updated DataFrame to a new CSV file
    df.to_csv('../Docs/labels_50239_with_author_and_doi.csv',
index=False)


def fix_empty_abstracts(data):
    for i, row in data.iterrows():
        # Check if the abstract is missing or null
        if pd.isnull(row['abstract']):
            # Extract the title and journal from the current row
            title = row['title (1)']
            journal = row['journal']
```

```
            # Build the query URL
            query = f'{title} {journal}'
            url =
f'https://scholar.google.com/scholar?q={query}&hl=en&as_sdt=0,5'

            # Send the request to Google Scholar and parse the response
            response = requests.get(url)
            soup = BeautifulSoup(response.text, 'html.parser')

            # Find the first search result and extract the abstract
            result = soup.find('div', {'class': 'gs_ri'})
            if result:
                abstract = result.find('div', {'class': 'gs_rs'}).text
                data.loc[i, 'abstract'] = abstract.strip()

    # Save the updated table to a new CSV file
    return data.to_csv('../Docs/train_table_with_abstracts.csv',
index=False)


def clean_citations_from_abstrackr(data):
    # drop unnecessary columns
    drop_columns = ['consensus', 'pubmed id', 'keywords', 'authors',
'tags', 'notes', 'labeled_at', '(source) id']
    data = data.drop(columns=drop_columns, errors='ignore')
    # rename columns
    data.rename(columns={'labeled_at.1': 'labeled_at', 'aaronalt':
'relevance', '(internal) id': 'citation_id'},
                inplace=True)
    for i, row in data.iterrows():
        if pd.isnull(row['title']):
            continue
        data.loc[i, 'title'] = re.sub(r'[^\w\s-]', '', row['title'])
        data.loc[i, 'title'] = data.loc[i, 'title'].strip().lower()
    # fix duplicates
    data = data.drop_duplicates(subset='title')
    # separate rows with '0' label into a new dataframe
    data_zeros = data[data['relevance'] == 0]
    data = data[data['relevance'] != 0]
    # clean empty abstracts, journals
    for i, row in data.iterrows():
        # Check if the abstract is missing or null
        if pd.isnull(row['abstract']):
            # Extract the title and journal from the current row
            title = row['title']
            journal = row['journal']
```

```
            # Build the query URL
            query = f'{title} {journal}'
            url =
f'https://scholar.google.com/scholar?q={query}&hl=en&as_sdt=0,5'
            # Send the request and parse results
            response = requests.get(url)
            soup = BeautifulSoup(response.text, 'html.parser')
            # Find the first search result and extract abstract
            result = soup.find('div', {'class': 'gs_ri'})
            try:
                if result:
                    abstract = result.find('div', {'class':
'gs_rs'}).text
                    data.loc[i, 'abstract'] = re.sub(r'[^\w\s-]', '',
row['abstract'])
                    data.loc[i, 'abstract'] = abstract.strip().lower()
            except TypeError:
                continue
    # Fill null values
    data['relevance'] = data['relevance'].fillna(0)
    # Join with Abstrackr predictions
    df_pred = pd.read_csv('../Docs/predictions_50239.csv')
    df_pred = df_pred.drop(columns='title', errors='ignore')
    df_pred.rename(columns={'(internal) id': 'citation_id'},
inplace=True)
    data = pd.merge(data, df_pred, on="citation_id", how="left")
    data = data.loc[:, ~data.columns.duplicated()].copy()
    data = data.dropna(subset=['abstract', 'title'])
    # Save the updated tables to new CSV files
    data.to_csv('../Docs/labels_50239_cleaned.csv', index=False)
    data_zeros.to_csv('../Docs/labels_50239_zeros.csv', index=False)


clean_citations_from_abstrackr(df_all)
fix_empty_author_and_doi(df_only_true)
```

D-2 Springer Link CSV to Bibtex

```
import springer_link_csv_to_bibtex_parser
import argparse


parser = argparse.ArgumentParser(description='Convert a SpringerLink
auto-generated CSV references file to Bibtex')
parser.add_argument('-i','--input', help='Provide the path to your input
csv file', required=True)
parser.add_argument('-o','--output', help='Provide the path to your
output folder', required=True)
```

```
parser.add_argument('-s','--source', help='Source of csv creation:
springer, orange, zotero...', required=True)
args = vars(parser.parse_args())


csv_to_bibtex_parser =
springer_link_csv_to_bibtex_parser.CsvToBibtexParser(args['input'],
args['output'],

args['source'])
csv_to_bibtex_parser.convert_csv_to_bibtex()
```

D-3 Springer Link CSV to Bibtex Parser

```
import pandas as pd
import re
import requests
from bibtexparser.bwriter import BibTexWriter
from bibtexparser.bibdatabase import BibDatabase
from bs4 import BeautifulSoup



def split_camel_case_joined_names(joined_camel_case_names):
    individual_camel_case_names = re.finditer('.+?(?:(?<=[a-z])(?=[A-
Z])|(?<=[A-Z])(?=[A-Z][a-z])|$)',
                                             joined_camel_case_names)
    return [name.group(0) for name in individual_camel_case_names]


def split_joined_names(joined_names, name_type):
    individual_names = str()
    if name_type == 'camel':
        individual_names = re.finditer('.+?(?:(?<=[a-z])(?=[A-
Z])|(?<=[A-Z])(?=[A-Z][a-z])|$)', joined_names)
    elif name_type == 'semicolon':
        individual_names = re.finditer(re.escape("[^;\s][^\;]*[^;\s]*"),
joined_names)
    return [name.group(0) for name in individual_names]


def join_names_as_camel_case(name):
    names_list = re.split('([^a-zA-Z\u00C0-\u024F\u1E00-\u1EFF])', name)
    first_name_lower_case = names_list[0].lower()
    other_names_camel_case = [name.capitalize() for name in
names_list[1:] if name.isalnum()]
    camel_case_list = [first_name_lower_case] + other_names_camel_case
    camel_case = ''.join(camel_case_list)
    return camel_case
```

```
class CsvToBibtexParser:
    """ Given a CSV file path to a SpringerLink auto-generated refer-
ences CSV and an output_file_path, provide the
     functionality to parse the CSV into an equivalent bibtex (.bib)
format """

    def __init__(self, csv_file_path, output_file_path, file_source):
        self.csv = pd.read_csv(csv_file_path)
        self.output_path = output_file_path
        self.source = file_source
        if self.source == 'springer':
            self.fix_empty_abstracts()

    def fix_empty_abstracts(self):
        # Create empty column for abstract
        self.csv['Abstract'] = ''
        for i, row in self.csv.iterrows():
            # Extract the title and journal from the current row
            title = row['Item Title']
            journal = row['Publication Title']
            # Build the query URL
            url = row['URL']
            # Send the request to Google Scholar and parse the response
            response = requests.get(url)
            soup = BeautifulSoup(response.text, 'html.parser')
            # Find the first search result and extract the abstract
            result = soup.find('div', {'id': 'Abs1-content'})
            # abstract = result.find('div', {'id': 'Abs1-content'}).text
            try:
                self.csv.at[i, 'Abstract'] = result.strip()
            except TypeError:
                self.csv.at[i, 'Abstract'] = result
            except AttributeError:
                self.csv.at[i, 'Abstract'] = result
        print(self.csv.head())

    def convert_csv_to_bibtex(self):
        csv_dict = self.csv.to_dict('records')
        writer = BibTexWriter()
```

```
        with open(self.output_path, 'w', encoding="utf-8") as
bibtex_file:
            for csv_entry in csv_dict:
                bibtex_entry = self.convert_csv_entry_to_bibtex_en-
try(csv_entry)
                bibtex_file.write(writer.write(bibtex_entry))


    def convert_csv_entry_to_bibtex_entry(self, document_record):
        bibtex_key = self.create_bibtex_entry_key_from_csv_entry(docu-
ment_record)
        bibtex_entry = BibDatabase()
        authors_list = self.get_authors_from_csv_entry(self.source, doc-
ument_record)
        formatted_authors_list = self.remove_braces_and_quotes_from_au-
thors_list(authors_list)
        if self.source == 'orange':
            bibtex_entry.entries = [
                {'journal': str(document_record['journal']),
                 'title': str(document_record['title']),
                 'abstract': str(document_record['abstract']),
                 'keywords': str(document_record['relevance']),
                 'ENTRYTYPE': 'Article',
                 'ID': str(document_record['citation_id'])}
            ]
        elif self.source == 'springer':
            bibtex_entry.entries = [
                {'journal': str(document_record['Publication Title']),
                 'title': str(document_record['Item Title']),
                 'author': formatted_authors_list,
                 'year': str(document_record['Publication Year']),
                 'abstract': str(document_record['Abstract']),
                 'doi': str(document_record['Item DOI']),
                 'url': str(document_record['URL']),
                 'journalVol': str(document_record['Journal Volume']),
                 'journalIssue': str(document_record['Journal Issue']),
                 'ENTRYTYPE': str(document_record['Content Type']),
                 'ID': bibtex_key}
            ]
        else:
            bibtex_entry.entries = [
                {'journal': str(document_record['Publication Title']) or
str(document_record['journal']),
                 'title': str(document_record['Title']) or str(docu-
ment_record['title']),
```

```
                'author': formatted_authors_list,
                'year': str(document_record['Publication Year']),
                'doi': str(document_record['DOI']),
                'url': str(document_record['Url']),
                'ENTRYTYPE': str(document_record['ItemType']),
                'abstract': str(document_record['Abstract Note']),
                'pages': str(document_record['Pages']),
                'issue': str(document_record['Issue']),
                'volume': str(document_record['Volume']),
                'ID': str(document_record['Key'])}
            ]
        return bibtex_entry

    def create_bibtex_entry_key_from_csv_entry(self, csv_entry):
        if self.source == 'springer':
            document_authors = self.get_authors_from_csv_en-
try(self.source, csv_entry)
            print(csv_entry)
            first_author = document_authors[0]
            first_author_camel_case = join_names_as_camel_case(first_au-
thor)
            document_year = csv_entry['Publication Year']
            return first_author_camel_case + str(document_year)
        else:
            print(csv_entry)

    @staticmethod
    def get_authors_from_csv_entry(source, csv_entry):
        if source == 'springer':
            document_authors = str(csv_entry['Authors'])
            document_authors_list = split_camel_case_joined_names(docu-
ment_authors)
            return document_authors_list
        document_authors_list = list()
        document_authors = str()
        try:
            document_authors = str(csv_entry['Authors'])
        except KeyError:
            document_authors = str(csv_entry['Author'])
        finally:
            if document_authors.find(';'):
                document_authors_list = split_joined_names(document_au-
thors, 'semicolon')
```

```
            elif not document_authors.find(';'):
                document_authors_list = split_joined_names(document_au-
thors, 'camel')
            return document_authors_list



    @staticmethod
    def remove_braces_and_quotes_from_authors_list(authors_list):
        authors_list_without_braces = str(authors_list)[1:-1]
        authors_list_without_braces_or_quotes = str(authors_list_with-
out_braces).replace("'", "")
        return authors_list_without_braces_or_quotes
```

INTERNATIONAL
UNIVERSITY OF
APPLIED SCIENCES

Declaration of Authenticity

I hereby declare that I have completed this Master's thesis on my own and without any additional external assistance. I have made use of only those sources and aids specified and I have listed all the sources from which I have extracted text and content. This thesis or parts thereof have never been presented to another examination board. I agree to a plagiarism check of my thesis via a plagiarism detection service.

Seattle, WA - 31.08.2023

Place, Date **Signature**