A Network-Graph Based IT Artifact Aiding the Theory Building Process

Michael Senft^, Edward Corrado*
^Department of Computer Science
*Dudley Knox Library
Naval Postgraduate School, Monterey, CA, USA
{michael.senft, emcorrad}@nps.edu

Abstract

To support theory building, we introduce a network-graph based IT artifact to provide high recall during exploratory searches and high precision using knowledge gained through the literature discovery process. The use of network graphs, where all data is represented as a node, relationship, or property of either, offers a flexible and tailorable methodology able to accommodate the highly iterative process of theory building. This IT artifact was developed to enable aggregation and normalization of data from varied sources and formats to support the acquisition and assessment of literature needed throughout this process. Our goal in presenting this IT artifact is to promote an accessible and pragmatic approach addressing the varied challenges of Information Systems researchers during the information seeking process.

1. Introduction

In the theory building process, the Information System (IS) researcher is like the ant described by Herbert Simon, generating an irregular path in traversing wind-swept terrain while moving from starting point to their goal [1]. Despite both having a general idea of where this goal lies, the path of the ant and IS researcher are never straight lines. Each are faced with obstacles and deep detours as they navigate their respective environments. For the IS researcher, successful exploration of their environment and movement toward their goal of theory development, first requires acquisition of a breadth and depth of knowledge on their topic of interest.

Literature reviews are essential in establishing a foundation of understanding for a given phenomenon, as well as situating the phenomenon within existing theoretical frameworks. In developing this foundation there is a critical need for greater replicability and repeatability in the literature review process, which is complicated by the substantial volume and continued grown of scientific literature [2,3]. The documents selected for review have a strong effect on the theories and relationships identified, making the need to start with a broad set of data critical [4]. Breslin and Gatrell describe the utility of literature reviews across a range of purposes and knowledge domains in what they term the Miner-Prospector Continuum [5]. This continuum highlights different paths available to researchers for knowledge acquisition along with the risks and rewards associated with them in the process of theorizing [5]. These risks and rewards are also emphasized in earlier research concerning challenges involved in theory integration between and within scientific disciplines along with opportunities missed due to their poor integration [6,7]. Challenges in identifying all relevant literature during the literature review process can also introduce bias, which has been previously highlighted in IS research. [8, 9].

The critical importance of the literature review process to reach erudition, a depth and breadth of understanding in a given topic, is highlighted by Rivard in presenting a spiral model of the theory-building process [10]. The spiral model reflects a highly iterative approach to the process of theory building with erudition as the desired outcome of the first iteration [10]. Reaching this level of knowledge requires identification and classification of relevant literature, which is also highlighted by Larsen et al. as the steps of boundary identification and corpus construction [9,10].

The challenges of identification and classification faced by IS researchers are frequently framed in terms of a recall-precision tradeoff, where a search tool must balance retrieval of all potentially relevant literature against retrieval of only literature that is actually relevant [9,11,12]. Previous works exploring theory integration using co-citation and metadata analysis have relied heavily on labor intensive methods for article discovery and extraction of pertinent content,

URI: https://hdl.handle.net/10125/80136
978-0-9981331-5-7

which have limited the utilization of the techniques described to wider audiences. With our network-graph based IT artifact we offer a lower complexity approach in addressing these challenges by combining the use of open-source resources, the Community Edition of Neo4j, a graph database, Jupyter Notebook, a web-based document development environment and the Python programming language [13,14]. This IT artifact is also able to quickly integrate heterogeneous literature data from any source accessible to the IS researcher.

2. Related Work

The use of network graphs to explore scientific literature has rich history, with computer-based techniques employed as early as 1974 that mapped relationships across scientific literature [15,16]. These works painstakingly mapped linkages using co-citation to identify similarities between documents to group and visualize these connections [15,16]. Network graphs were also used in heavily cited work mapping the development of DNA theory [17]. Contemporary examples expand on the scope, scale, and accessibility of these techniques across multiple scientific domains [18,19]. Recent work has also provided IS researchers with an interactive tool, RelPath, to explore citation networks and citation paths [20].

Diverse approaches in addressing the recall and precision challenges faced by IS researchers in identifying and classifying relevant research during the literature process have been offered recently through several IT artifacts. Litbaskets.io is a publicly available exploratory literature search tool developed to assist researchers with targeted searches of IS journals within Scopus [11]. DISKNET is also a publicly available online platform, which leverages structural equation modeling techniques to explore relationships between theoretical constructs [21]. TheoryOn provides a robust search engine allowing IS researchers to directly query constructs and relationships and enable theory integration through the construction of theoretical networks [12]. Seeking to improve the ability for IS researchers to identify relevant literature, a technique for Automated Detection of Implicit Theory (ADIT) was developed and evaluated on its ability to provide precision and comprehensiveness [9]. These works provide an abundance of insights in the development and application of IT artifacts supporting IS researchers confronted by an immense and ever-growing volume of scientific literature.

The use of computational techniques to conduct systematic and rigorous literature reviews also informed the development of this IT artifact. Antons et al. (2021) provide a six-step roadmap outlining the computational literature review (CLR) process, which details roles of both human and machine with issues for consideration at each step [2]. Mortenson and Vidgen (2016) offer a CLR approach to compliment human researchers while investigating the technology acceptance model [22]. Their approach addressed issues relating to selecting, filtering, and analyzing content from an enormous number of published articles available [22]. Portenoy and West (2019) also investigated the use of automated literature reviews leveraging supervised learning techniques to provide new insights and help address challenges that created by the volume and complexity of scientific literature [23].

3. Exploratory Search Artifact Design

The exploratory search IT artifact was initially developed as a mechanism to connect literature search results from multiple sources, including Harzing's Publish or Perish software, for identification of relevant research in the cross-disciplinary field of deception [24]. The importance of effective literature reviews and the need to minimize bias introduced in literature selection are well documented, but limited tools are available to IS researchers to aggregate and analyze exploratory literature search results across multiple databases [8,9,11]. A highly flexible and accessible tool is needed to aide in the acquisition and analysis of the breadth and depth of literature required to begin the theory building process, which motivated the construction of the IT artifact.

Leveraging open-source resources, this IT artifact was developed using a combination of the Python programming language and Neo4j, a network graph database. Neo4j was selected as the database platform because a full open-source version is available for non-commercial applications [13]. Neo4j's scripting language, Cypher, and advanced functionality provided by APOC and Graph Data Science libraries deliver a wide range of computational techniques for analysis. Neo4j is available as a locally installed application or as a cloud-native service through Neo4j Aura [13]. Neo4j also is highly scalable, capable of easily supporting millions of nodes and relationships on commodity hardware [13]. For the example presented, 3 million nodes and relationships were created using desktop installations of Neo4j and Jupyter Notebook running on a laptop with a 2.2GHz Six-Core Processor and 16 Gigabytes of RAM.

Python is used as the scripting language to both query academic knowledge datasets with an available application programming interface (API) and to parse data manually downloaded from datasets without automated retrieval mechanisms. The Python driver available for Neo4j also serves as the interface to facilitate ingest of the data into the network graph database [14]. All scripting is written within Jupyter Notebook to facilitate easy sharing of code across a broad user community [25]. While methodology presented uses Cypher, Neo4j's native graph language, it can be applied to any network graph database platform by adapting the syntax provided [14].

The core of this IT artifact is the basic data model schema depicted in Figure 1, consisting of nodes representing entities associated with a document, and edges, which represent relationships between entities. The data model schema displayed in Figure 1 was developed to highlight the relationships between key entities associated with academic literature, the person(s) who wrote the document, the organization they are affiliated with, the journal the document is published in, any documents referenced by the original document, and any topic or keywords associated with the document. Similar schemas are used by Semantic Scholar and Microsoft Academic [26,27].

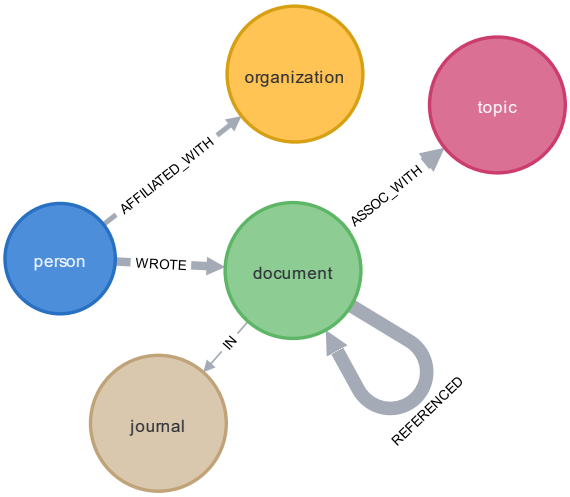

Figure 1. IT Artifact data model schema

Because all data is represented as a node, relationship, or property of either, construction of the data model schema within a network graph database is highly flexible. Unlike a relational database, the data model schema does not need to be pre-defined and thus can be modified or updated as new sources of information are added. Additionally, properties associated with nodes or relationships can be included as stand-alone search query parameters or in conjunction with relationship-based queries. The highly flexible nature of the Python scripting used in the tool combined with the customizable modeling of data within the network graph database provides a capability to enrich the academic research metadata with any other data in tabular/Comma Separate Value (CSV), eXtensible Markup Language (XML), or JavaScript Object Notation (JSON) formats.

4. Data Import

Highlighting the flexibility provided using network graphs in exploratory literature searches to support the theory building process, three techniques to import data into the IT artifact are discussed: (1) Manual, (2) Static, (3) Dynamic. These techniques support a range of search methods used to identify potentially relevant documents: informal approaches based on existing personal knowledge or personal contacts; protocol driven, pre-defined search strategies; and "snowballing" techniques pursuing references of references [28]. These techniques also support the distinct processes involved in creating a corpus of documents for analysis of boundary identification and corpus construction as described by Larsen et al. [9]. The use of network graphs provides transparency in separating these processes through the ease of manipulating and extracting data. Regardless of format, metadata from relevant documents identified through a comprehensive methodology can be imported into this IT artifact. Precision can be achieved through the wealth of capabilities provided by Neo4j to query and filter data. Corpus construction can be accomplished by retrieving documents that meet specific criteria or have been otherwise identified through the addition of a specific property to the document node.

4.1. Manual Data Import

While manual entry of data is not the primary technique to import data into the IT artifact, it may be useful as a starting point to include existing personal resources or knowledge or data from "gray literature" often not included in scientific literature datasets [9]. Serendipitous discovery of an unpublished draft located on a dusty library shelf can provide useful insights into the early development of theory. Creation of a new document node is simply achieved using the syntax below. The "d" and "p" letters used in these, and other examples are variable names and have no significance outside of the specific query.

```
CREATE (d:document {Author: "barton whaley", Title: "a reader in deception and counterdeception", Document_Type: "unpublished draft", Source: "library"})
```

Creation of a person node is accomplished using the same process, with a relationship created between author and document using the syntax below. The same process is used to create and link topic nodes for "deception" and "counterdeception".

```
MATCH (p:person),(d:document) WHERE p.Node_Key = d.Author CREATE (p)-[:WROTE]->(d)
```

The result is a graph depiction of the unpublished draft authored by Barton Whaley illustrated in Figure 2. Additional nodes, relationships and properties can be quickly added to capture key elements from personal resources and knowledge or "gray literature" that are not readily available in a standardized format or database.

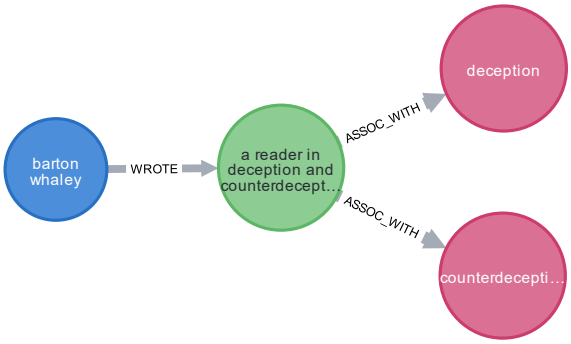

Figure 2. Graph representation of manually entered data

The same process described to manually create nodes, relationships and properties is used by the Static and Dynamic import options, with fields from the source spreadsheet, JSON, or XML formatted data paired with the desired node, relationship, or property. The IS researcher is able to modify the data model schema dynamically to best suit their requirements throughout the iterative process of theory building.

4.2. Static Data Import

Clarivate's Web of Science provides access to multidisciplinary research across more than 20,000 scholarly journals [29]. Currently, access to this data is only available using a web-based portal. Within the Web of Science Core Collection, several query options are provided, including "topic", which searches title, abstract and keywords [29]. This platform provides an example of how a protocol driven search technique, with use of specific terms defined at the beginning of a search effort, can be executed using this IT artifact [28].

A topic query on the term "deception" returned 12,787 records, covering a range of documents from journal articles to book chapters. These records, which include 67 data fields, can be exported as an Excel spreadsheet, with the limitation that no more than 1000 records can be exported at a time. All 12,787 records were exported as Excel spreadsheets in 13 separate manual downloads in accordance with the platform's terms of use. Using Python Using Python, the Excel spreadsheets were converted into CSV format for ingestion into Neo4j [30]. Of the 67 data fields available, four are depicted as nodes (Document, Authors, Journal, and Keywords) illustrated in Figure 3.

Figure 3. Graph representation of Web of Science data

Ten of the retrieved data fields are included as properties within the Document node depicted in Table 1 along with a "Source" property added to identify where the data was obtained from. Additional data fields could be included as either new nodes or properties within existing nodes or relationships. Within Web of Science, a field for Cited References was recently added, but the data provided consisting of author name, year, and journal created challenges in both retrieving and linking to these references. Uncertainty is present when matching records because the data elements provided for Cited Reference are not unique identifiers.

Table 1. Tabular representation of Web of Science data

Title:	"interpersonal deception theory"
Abstract:	"interpersonal deception theory (idt) represents a merger . . ."
Author:	["buller, db", "burgoon, jk"]
CitationCount:	540
Document_Type:	"review"
DOI:	"10.1111/j.1468-2885.1996.tb00127.x"
Journal_Name:	"communication theory"
Node_Key:	"WOS:A1996VH09800001"
Publish_Date:	1996-01-01
References:	[Bauchner J. E., 1977, COMMUNICATION YB, V1, P229; Bauman Richard., 1986, STORY PERFORMANCE EV;…]
Source:	"Web of Science"

4.3. Dynamic Data Import

To demonstrate the dynamic data import option, Microsoft Academic was used due to its free and publicly available API, which enables automated retrieval of document records. Use of this API through a combination of scripting in Jupyter Notebook and Neo4j provides an example of how a "snowballing" search technique can be executed for automated retrieval of references of references [28]. This API allows dataset queries on a wide range of parameters including topic, journal, and organization after free registration to obtain an API key [31]. The primary limitation on use of this API is only one query may be executed per second, but each query could potentially return thousands of records. Metadata on academic literature related to the field of deception was retrieved from Microsoft Academic using its API. As depicted in the first step of Figure 4, a request was sent to the API to return 12 fields of information for each of the over 20,000 publications tagged by Microsoft Academic's processing algorithms as being associated with the topic of deception. The metadata fields retrieved included the names of the authors, article title, journal name, type of document, and list of ID numbers for the publications referenced by each document. Through the Microsoft Academic API, it is possible to specify the fields of information returned in each query from the many fields available [32].

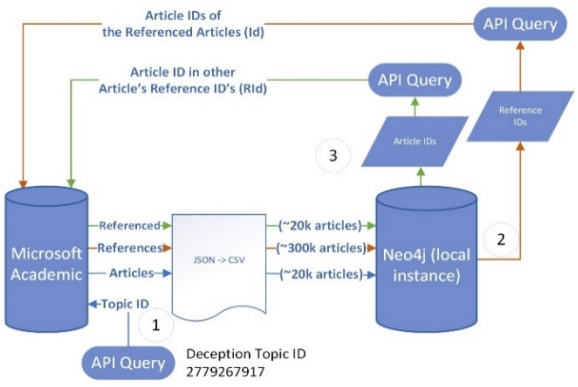

Figure 4. Dynamic data retrieval process

Because Microsoft Academic provides a list of ID numbers for the publications referenced by each document, it is possible to expand the metadata retrieved beyond the publications associated by Microsoft Academic's algorithms with deception to include articles both referenced and referenced by these original publications. Acquisition of metadata for documents referenced by the original set of publications retrieved is accomplished by executing queries using the ID numbers listed as references. This retrieval process is illustrated in Figure 4, with an example document reference list depicted in Table 2.

Queries to return the metadata for documents that referenced the original set of publications is achieved by executing queries to retrieve metadata for documents that include the ID numbers of the original set of publications within their list of references indicated in step 3 of Figure 4. The initial download of publications associated with the topic of deception retrieved metadata on 20,000 documents. Retrieving references from and for these publications resulted in metadata on a total of 330,000 documents. This information was used to construct the graph illustrated in Figure 5. Two key differences between Figure 3 and Figure 5 are the inclusion of author affiliation represented by an "AFFILIATED_WITH" relationship connecting to an "organization" node and inclusion of a document referenced by the original document and a subsequent document that referenced the original document.

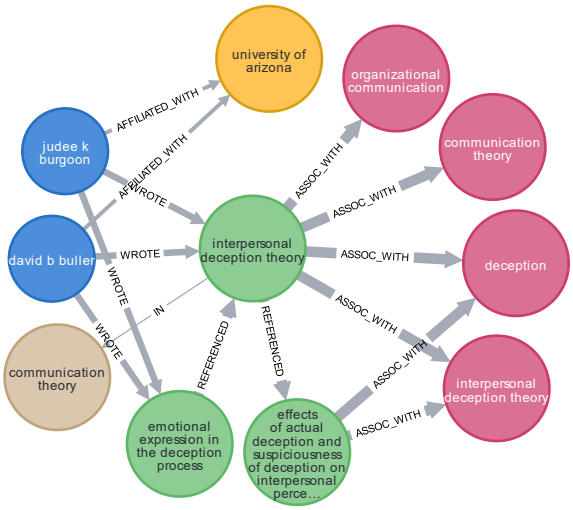

Figure 5. Graph representation of Microsoft Academic data

Like the previous example, ten of the retrieved data fields are included as properties within the Document

node depicted in Table 2, along with a "Source" property to identify from where the data were obtained. The use of a unique identifier to connect references enables the "snowball" effect to retrieve references of and references to a specific document.

Table 2. Tabular representation of Microsoft Academic Data

Title:	"interpersonal deception theory"
Abstract:	"interpersonal deception theory (idt) represents a merger . . ."
Author:	["david b buller","judee k burgoon"]
CitationCount:	838
Document_Type:	"journal article"
DOI:	"10.1111/J.1468-2885.1996.TB00127.X"
Journal_Name:	"communication theory"
Node_Key:	2163220194
Publish_Date:	1996-08-01
References:	["2159035740","2576297379","1606729893",…]
Source:	"Microsoft Academic"

4.4. Database Integration

Integration of data from multiple data sets is an immense challenge. This IT artifact addresses but does not fully resolve the issue of data integration. It does, however, provide the IS researcher with the ability to identify overlap between documents retrieved from multiple datasets. The examples provided for the static and dynamic import highlight duplication between Web of Science and Microsoft Academic for articles related to deception. Fortunately, Neo4j has native functionality that enables users to compare data across the spectrum desired precision as illustrated in Table 3. This table highlights the number of duplicate articles identified using a range of specificity from explicit match between document title and year published to a fuzzy match comparing document titles. The fuzzy match within Neo4j utilizes the LevenShtein algorithm to compare document text strings [33].

Table 3. Duplicate document comparison

Explicit Title and Publish Year Match	Explicit Title Match	Fuzzy Title and Publish Year Match	Fuzzy Title Match
1645	1855	4614	5337

Cypher syntax is provided for the most and least specific document comparisons to highlight the ease of either approach, which can be further refined to include use of additional properties or relationship to identify duplicates as needed to best meet the needs of the IS researcher. Document nodes with duplicate titles can be merged or deleted depending on the need of the user.

Cypher syntax to return the count of explicit title and publish year matches:

```
MATCH (d1:document),(d2:document) WHERE d1.Title =
d2.Title AND d1.Source = "Web of Science" AND d2.Source =
"Microsoft Academic"  AND d1.Publish_Date.year =
d2.Publish_Date.year RETURN COUNT(d1)
```

Cypher syntax to return the count of fuzzy title matches:

```
MATCH (d1:document),(d2:document) WHERE
apoc.text.fuzzyMatch(d1.Title,d2.Title) = TRUE AND d1.Source
= "Web of Science" AND d2.Source = "Microsoft Academic"
RETURN COUNT(d1)
```

Even with the least restrictive matching criteria, the gap in literature coverage between Web of Science and Microsoft Academic on the multidisciplinary topic of deception ranges is more than 50%. This introduces bias in the literature review process as described by Larsen et al. and vom Brocke et al., when a review is conducted using a set of articles not representative of the overall population of articles [8,9]. The ability to integrate and deduplicate data from multiple sources, when available, may help minimize bias introduced compared to data obtained from a single source.

5. Theory Ecosystem Exploration

In exploring relationships between theories associated with computer deception and cyber deception, a series of queries were created to investigate the article metadata. This example provides a basic demonstration using simple keywords, but the process presented is easily expandable to accommodate queries involving complex variables or constructs through the native functionality of Neo4j. In identifying documents specifically related to "computer deception" or "cyber deception", it is possible to query for these terms within the title or abstract for each document. Additionally, depending on how narrow or wide the desired exploration is, it is possible to query for these terms as phrases or as individual words. How these terms are queried has a dramatic impact on the returned results. Querying the metadata on approximately 330,000 documents for the terms "computer" and "deception" individually in the abstract property returns 910 documents, while the same search in the title property returns 172 documents. The syntax used for this query is below.

```
MATCH (d:document) WHERE d.Title CONTAINS "computer"
AND d.Title CONTAINS "deception" RETURN COUNT(d)
```

Similarly searching for the terms "cyber" and "deception" individually returns 283 documents with these terms in the title and 665 documents with these terms in the abstract. Searching for the exact phrases, however, can have a dramatic impact. Searching for the exact term "computer deception" returns a single result in the title and zero results within abstracts when querying all 330,000 documents. Searching for the term "cyber deception" returns 112 results in the title and 109 results in the abstract of the same corpus of documents.

The real power of this IT artifact is not in the keyword or key phrase search, but rather in the ability to query based on the relationships within the data being queried. In surveying the theories being leveraged for research in computer deception and cyber deception, it is possible to search for documents that have referenced articles with "theory" in their title by building on the previous query. The query below is designed to return a count of articles that contain the terms "computer" and "deception" in the title that referenced any article with the term "theory".

```
MATCH (d1:document)<-[:REFERENCED]-(d2:document)
WHERE d1.Title CONTAINS "theory" AND d2.Title CONTAINS
"computer" AND d2.Title CONTAINS "deception" RETURN
d1.Title, SIZE(COLLECT(d2.Title))
```

This query returned a total of 53 results, but only the top five theory articles by number of times referenced are show in Table 4 below.

Table 4 - Computer Deception Theory Connections

Title	Author(s)	Times Referenced
Interpersonal deception theory	D. Buller, J. Burgoon	32
Channel expansion theory and the experiential nature of media richness perceptions	J. Carlson, R. Zmud	10
Testing interpersonal deception theory the language of interpersonal deception	D. Buller, J. Burgoon, A. Buslig, J. Roiger	8
Resting media richness theory in the new media the effects of cues feedback and task equivocality	A. Dennis, S. Kinney	5
Information manipulation theory	S. Mccornack	5

The same query performed for articles with the terms "cyber" and "deception" in their title that referenced articles with "theory" in their title is shown in Table 5. Even though there are roughly double the number of articles related to "computer deception", articles related to "cyber deception" have significantly fewer references to articles concerning "theory".

Table 5 – Cyber deception theory connections

Title	Author(s)	Times Referenced
Toward a general theory of deception	B. Whaley	13
Game theory meets network security and privacy	M. Manshaei, Q. Zhu, T. Alpcan, T. Bacsar, J. Hubaux	9
A survey of game theory as applied to network security	S. Roy, C. Ellis, S. Shiva, Q. Wu D. Dasgupta, V. Shandilya	6
Physical intrusion games optimizing surveillance by simulation and game theory	S. Rass, A. Alshawish, M. Abid, S. Schauer, Q. Zhu, H. DeMeer	5
Interpersonal deception theory	D. Buller, J. Burgoon	4

In analyzing Table 4, the most referenced theories from articles related to computer deception cover a wide range of theories including interpersonal deception theory, channel expansion theory, media richness theory, and information manipulation theory. Interestingly, interpersonal is referenced more times than the other top 5 results combined. In contrast, Table 5 indicates for articles related to cyber deception, game theory is the most influential, but general deception theory and interpersonal deception theory are also referenced. With this information, it is possible to determine if any articles relating to cyber deception have referenced both general deception and interpersonal deception theories using a targeted query matching multiple property and relationship constraints. The syntax provided below uses the unique document identifier provided by Microsoft Academic for the deception theory articles written by Whaley and Buller and Burgoon represented by the "Node_Key" property. Any records returned need to reference both of these documents and be associated with the topic of deception.

```
MATCH (d1)<-[:REFERENCED]-(d)-[:REFERENCED]-
>(d2),(d)-[:ASSOC_WITH]-(p:topic) WHERE d1.Node_Key =
"2077375749" AND d2.Node_Key = "2163220194" AND
p.Node_Key = "deception" RETURN d
```

Only a single document within the dynamic data import over 300,000 articles from Microsoft Academic met these criteria. This article, titled "Online Social Deception and Its Countermeasures for Trustworthy Cyberspace: A Survey" was written by Guo et al. and published in 2020. The visual results of this query showing all articles are illustrated in

Figure 6. While this analysis of theory related to cyber deception is cursory, the capabilities of the IT artifact described provide a flexible approach in addressing the tradeoff between precision and recall during literature searches.

Figure 6 – Theory Reference Connections

6. Limitations

The primary limitation of this IT artifact is that use in its current form requires a working knowledge of basic programming concepts to effectively apply the import and analysis techniques described to explore boundary identification and corpus construction to the domain or phenomenon being studied. Working knowledge of the use of network graphs to store and visualize data is also needed to tailor the IT artifact to support specific use cases for IS researchers. Currently, parsers to import data into Neo4j have been written to support manually downloaded spreadsheets containing literature metadata from Clarivate's Web of Science and the Defense Technical Information Center and API connections to Microsoft Academic. Parsers to process additional data sources frequently used by IS researchers, including Google Scholar, Scopus, Semantic Scholar, and OpenCitations are needed to expand the utility of this IT artifact. Experimentation is also needed to assess both the recall and precision of the IT artifact in comparison with existing tools and the perceived utility of the IT artifact by practitioners engaged in literature review and theory building processes.

7. Future Work

One goal with this IT artifact is to release the Python and Cypher codebase used to import and analyze scientific literature data on a publicly available platform. The use of open-source resources for the construction of this IT artifact and capabilities enabled with the use of network graphs to store and visualize data offer new opportunities to increase automation and replicability in literature reviews. Additionally, the flexibility of this approach creates openings to integrate concepts and outputs from a wide range of existing IT artifacts and frameworks to increase the accessibility and utility of automation to support IS researchers. Collaboration with other researchers developing tools and techniques addressing recall and precision challenges experienced throughout the literature review and theory building processes is critically needed. Further refinement of the IT artifact also has potential to address several literature search challenges outlined by vom Brocke et al., including reducing the difficulty and time required to store and retrieve publication data and improving teamwork cohesion during literature reviews [8].

Neo4j also provides a Natural Language Processing (NLP) functionality that transforms raw text into network graphs using existing Python libraries in conjunction with Cypher scripting or by leveraging NLP resources offered by Amazon Web Services, Azure or Google Cloud [34].The network graphs generated by either method can be easily integrated into the processes using literature metadata described in this paper. The NLP functionality also provides the ability to automate the creation and categorization of constructs for both theory development and integration.

8. Conclusion

The primary contribution of this paper is a generalizable and flexible IT artifact capable of supporting a wide range of data sources during the corpus construction process and subsequent analysis of these documents during the iterative process of theory building. The trans-disciplinary nature of IS research compounded with the constantly increasing number of scientific articles produced requires automated tools to identify and analyze these works [35]. Automated tools are also needed for theory integration to address hurdles created by the fragmentation of research, changing variables, and evolution of terminology [6]. This IT artifact provides IS researchers with flexible data aggregation, automation, and analysis capabilities using open-

source platforms to support the theory building process without requiring extensive technical knowledge.

9. References

[1] Simon, H. *The Sciences of the Artificial (3rd Ed.).* The MIT Press, Cambridge, Massachusetts, 1996.

[2] Antons, D., Breidbach, C. F., Joshi, A. M., & Salge, T. O. "Computational Literature Reviews: Method, Algorithms, and Roadmap", *Organizational Research Methods*, 2021, pp. 1-32.

[3] Tauchert, C., Bender, M., Meshbah, N. & Buxmann, P., "Towards an Integrative Approach for Automated Literature Reviews Using Machine Learning", *Proceedings of the 53rd Hawaii International Conference on System Sciences*, 2020, pp. 762-771.

[4] Hovorka, D. S., Larsen, K. R. T., & Monarchi, D., "Conceptual Convergences: Positioning Information Systems Among the Business Disciplines", *ECIS 2009 Proceedings, 36,* 2009, pp. 1-14.

[5] Breslin, D., & Gatrell, C., "Theorizing Through Literature Reviews: The Miner-Prospector Continuum", *Organizational Research Methods,* 2020, pp. 1-29.

[6] Hovorka, D. S., Larsen, K. R., Birt, J., & Finnie, G., "A Meta-theoretic Approach to Theory Integration in Information Systems", *Proceedings of the 46th Hawaii International Conference on System Sciences,* 2013, pp. 4656-4665.

[7] Börner, K., "Making Sense of Mankind's Scholarly Knowledge and Expertise: Collecting, Interlinking, and Organizing What We Know and Different Approaches to Mapping (Network) Science.", *Environment and Planning B-Planning & Design 34*(5), 2007, pp. 808-825.

[8] vom Brocke, J., Simons, A., Riemer, K., Niehaves, B., Plattfaut, R., & Cleven, A., "Standing on the Shoulders of Giants: Challenges and Recommendations of Literature Search in Information Systems Research.", *Communications of the Association Information Systems* 37(9), 2015, pp. 205-224.

[9] Larsen, K., Hovorka, D., Dennis, A., & West, J., "Understanding the Elephant: The Discourse Approach to Boundary Identification and Corpus Construction for Theory Review Articles.", *Journal of the Association for Information Systems 20*(7), 2019, pp. 887-927.

[10] Rivard, S. "Theory Building Is Neither an Art nor a Science. It Is a Craft." *Journal of Information Technology 20(7)*, 2021, pp. 316–328.

[11] Boell, S., and Wang, B., "www.Litbaskets.Io, an IT Artifact Supporting Exploratory Literature Searches for Information Systems Research." 2019. *ACIS 2019 Proceedings. 71.*

[12] Li, J., Larsen, K., and Abbasi, A., "TheoryOn: A Design Framework and System for Unlocking Behavioral Knowledge Through Ontology Learning." *Management Information Systems Quarterly 44(4)*, 2020, pp. 1733–1772.

[13] Neo4j, *What is Neo4j*, 2021; Available from https://neo4j.com/.

[14] Github, *Neo4j Python Driver*, 2021; Available from https://github.com/neo4j/neo4j-python-driver.

[15] Small, H., & Griffith, B. C., "The Structure of Scientific Literatures I. Identifying and Graphing Specialties", *Social Studies of Science 4*(1), 1974, pp.17-40.

[16] Griffith, B. C., Small, H. G., Stonehill, J. A., & Dey, S., "The Structure of Scientific Literatures II: Toward a Macro- and Microstructure for Science", *Social Studies of Science 4*(4), 1974, pp. 339-365.

[17] Hummon, N. P. & Dereian, P., "Connectivity in a Citation Network: The Development of DNA Theory", *Social Networks 11*(1), 1989, pp. 39-63.

[18] Ahmed, A., Khan, M. F., Usman, M., & Saleem, K., "Analysis of Coauthorship Network in Political Science using Centrality Measures", *International Journal of Advanced Computer Science and Applications 9*(10), 2018, pp. 329-341.

[19] Min, C., Chen, Q., Yan, E., Bu, Y., & Sun, J., "Citation Cascade and the Evolution of Topic Relevance", *Journal of the Association for Information Science and Technology 72*(1), 2021, pp. 110-127.

[20] Guilarte, O. F., Barbosa, S. D. J., & Pesco, S., "RelPath: An Interactive Tool to Visualize Branches of Studies and Quantify the Expertise of Authors by Citation Paths", *Scientometrics 126*(6), 2021, pp. 4871-4897.

[21] Dann, D., Maedche, A., Teubner, T., Mueller, B., Meske, C., & Funk, B., "DISKNET – A Platform for the Systematic Accumulation of Knowledge in IS Research", *40th International Conference on Information Systems,* 2019, pp 1-9.

[22] Mortenson, M. J. & Vidgen, R., "A Computational Literature Review of the Technology Acceptance Model", *International Journal of Information Management: The Journal for Information Professionals 36*(6), 2016, pp. 1248-1259.

[23] Portenoy, J. & West, J., "Supervised Learning for Automated Literature Review.", *Proceedings of the 4th Joint Workshop on Bibliometric-enhanced Information Retrieval and Natural Language Processing for Digital Libraries (BIRNDL 2019)*, 2019, pp. 83-91.

[24] Harzing, A.W. (2007) *Publish or Perish*, Available from https://harzing.com/resources/publish-or-perish.

[25] Project Jupyter, *The Jupyter Notebook*, 2021; Available from https://jupyter.org/.

[26] Ammar, W., et al. "Construction of the Literature Graph in Semantic Scholar." *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 3 (Industry Papers)*, 2018, pp. 84–91.

[27] Sinha, A., et al. "An Overview of Microsoft Academic Service (MAS) and Applications." *Proceedings of the*

24th International Conference on World Wide Web, 2015, pp. 243–246.
[28] Greenhalgh, T., and Peacock, R., "Effectiveness and Efficiency of Search Methods in Systematic Reviews of Complex Evidence: Audit of Primary Sources." *BMJ*, vol. 331, no. 7524, 2005, pp. 1064–1065.
[29] Clarivate., *Web of Science.* Available from https://clarivate.com/webofsciencegroup/solutions/web-of-science/.
[30] Neo4j, *Load CSV*, 2021; Available from https://neo4j.com/docs/cypher-manual/current/clauses/load-csv/#load-csv-import-data-from-a-csv-file-containing-headers.
[31] Wang, K., Shen, Z., Huang, C., Wu, C.-H., Eide, D., Dong, Y., … Rogahn, R., "A Review of Microsoft Academic Services for Science of Science Studies", *Frontiers in Big Data 2*(45), 2019, pp. 1-15.
[32] Microsoft Corporation., *Project Academic Knowledge*; Available from https://docs.microsoft.com/en-us/academic-services/project-academic-knowledge/.
[33] Neo4j, *Text Functions*, 2021; Available from https://neo4j.com/labs/apoc/4.0/misc/text-functions/.
[34] Needham, M. *QuickGraph #7: An entity graph of TWIN4j using APOC NLP*, 2020; Available from https://www.markhneedham.com/blog/2020/05/05/quick-graph-building-entity-graph-twin4j-apoc-nlp/.
[35] Anisienia, A., Mueller, R. M., Kupfer, A., & Staake, T., "Research Method Classification with Deep Transfer Learning for Semi-Automatic Meta-Analysis of Information Systems Papers", *Proceedings of the 54th Hawaii International Conference on System Sciences*, 2021, pp. 6099-6108.