Integrating LLMs and Psychometrics: Global Construct Validity

Completed Research Paper

Kai R. Larsen
University of Colorado
995 Regent Dr., Boulder, CO 80303
kai.larsen@colorado.edu

Sen Yan
University of Colorado
995 Regent Dr., Boulder, CO 80303
sen.yan@colorado.edu

Roman Lukyanenko
University of Virginia
140 Hospital Dr., Charlottesville, VA 22904
romanl@virginia.edu

Abstract

The psychometric approach in IS offers a foundational framework for a broad spectrum of research endeavors, typically relying on construct validation to confirm that a series of indicators accurately measures the intended construct. However, a longstanding issue with construct validity, unaddressed since its introduction by Cronbach and Meehl in 1955, is that it is evaluated using study-specific response data without comparison to constructs outside the study. This oversight (or, rather, incapability) has significant implications. We introduce a large language model combined with principal components analysis (PCA) and develop the *Validity Lodestar* application. This approach lays the groundwork for developing more accurate and reliable theoretical models, marking a significant leap forward in the IS discipline's methodological capabilities, making IS the first psychometric discipline with the capability to properly evaluate construct validity.

Keywords: Construct validity, Global construct validity, Validity Lodestar, Principal Components Analysis (PCA), Large Language Models (LLM).

Introduction

For research to be insightful and actionable, we must be able to conceptualize and measure the phenomena of interest appropriately. Hence, social sciences, including Information Systems (IS), postulate conceptual entities known as theoretical constructs (Danziger 1994). Theoretical constructs are *representations* of the phenomena of interest – such as the traits, beliefs, or behaviors of individuals, collectives, or social, physical, and artificial entities presumed to exist. Direct observation cannot verify many such phenomena (Bagozzi and Fornell 1982; Hitchcock and Nastasi 2011).[1] In the IS discipline, much of what is known about systems' design, use, and impact is expressed through a network of such constructs, known as a *nomological network* (Cronbach & Meehl, 1955).

Each discipline employs patterns and approaches to evaluate its constructs (Hoyningen-Huene, 1993). The approaches employed in psychometrics are those of content and construct validity, where content validation ensures the measurement is representative of the construct we hope to measure (Schmitz & Storey, 2020). In contrast, construct validity ensures independence between purportedly different sets of indicators and

[1] We note that the ontological nature of constructs is a subject of debate (Bagozzi and Fornell 1982; Bollen and Diamantopoulos 2017; MacKenzie 2003; MacKenzie et al. 2011; Weber 2021).

the interrelationships between the indicators in each set (Cronbach & Meehl, 1955). Unfortunately, although these validation approaches have supported psychometric research, they suffer from significant problems that have not been solved within any psychometric discipline.

We argue that some of these challenges stem from current approaches to validating constructs being narrowly scoped, typically within a single study and for a given theory. We call the constructs within a single study and for a given theory *a local construct network*. We refer to current approaches to validation based on data collected within a single or a group of related studies as *local construct validation*.

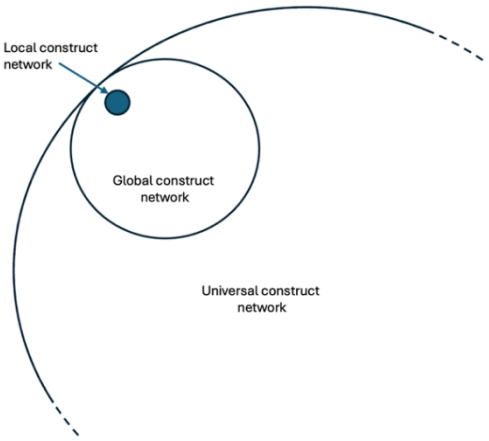

Figure 1. Local vs. Global construct networks.

In contrast, and as evident by notions of exploring or expanding the nomological network of a theory (c.f., Galperin 2012), a more extensive network must exist beyond a given theory proposed and evaluated by a researcher. Because current construct validation approaches evaluate only a small set of constructs included in one focal study (see Figure 1), they are missing the broader knowledge network within a discipline, or *global construct network*. *We define* a *global construct network as a set of constructs within a single area or discipline of research*. We further assume that a larger construct network exists beyond a discipline (universal construct network), but this is beyond our current scope.

We focus this article on problems stemming from the tension between local and global construct validation, which suggests that the social sciences are not living up to their potential because it is not currently possible to properly scope the boundaries of the constructs, which are thus allowed to tap into the domain of other constructs. For example, the construct validity of the self-efficacy construct in Social Cognitive Theory (Bandura 1977), still in use in contemporary IS work (e.g., Yoo et al. 2020), is in question because it is entangled and confounded with motivation (Williams and Rhodes 2016). Notably, in the case of self-efficacy, early detection of the overlap with motivation, which is not part of Social Cognitive Theory, could have been addressed with minor adjustments leading to appropriate differentiation (Williams and Rhodes 2016). As such distinction was not made promptly, more than forty years of empirical work is tainted.

Complementing local construct validation with global construct validation has significant value. Such validation, which does not exist in the literature today, would evaluate construct validity against the other constructs in the same study and against the constructs in a discipline. Such a concept would be of particular value for IS. Our discipline is diverse and addresses constant technological change. This makes it especially pertinent to build bridges among the disparate information systems communities (Goes 2013).

In this study, we develop global construct validity as a lodestar for research. We detect patterns of stable construct representations that are available for researchers to build into theories without overlapping different constructs. The intention is not to restrict the creativity of behavioral researchers, who will still be free to propose new constructs. They will receive guidance about the common names of the proposed constructs, definitions, typical measurement instruments, and whether their own constructs cross-load on multiple existing construct components. Articulating these patterns provides evaluative clarity, increases research efficiency, promotes the sharing of best practices, facilitates cumulative science and meta-analysis, and contributes to greater public trust in science. We conducted a user study with three renowned IS scholars to evaluate these claims.

Next, we review existing construct validity research and establish the need for global construct validation. Second, we develop a model for global construct validation and assure its criterion validity. Third, we develop a website through which the global construct validation model will be accessible to researchers in the discipline. Fourth, in the evaluation section, we evaluate the model against local construct validity for a recently published paper and establish model and ecological validity through an applicability check with experts in the IS discipline. Finally, we conclude by discussing contributions, weaknesses, and next steps.

CONSTRUCT VALIDITY EVALUATION

Current Practice

Most psychometric literature, including that published in IS journals, relies on language artifacts such as definitions and measurement indicators to derive and validate constructs and their relationships (Rajamanickam 2001; Trochim et al. 2015). Language has "a particular discriminatory value and, hence, a vital role in creating a complicated construct system" (Noaparast 2000, p. 69). Although researchers use their prior knowledge to propose constructs, the result is contingent on the extensiveness and depth of this knowledge, which naturally varies between researchers. Significant progress has been made in improving our understanding of methods for establishing construct validity. At the same time, the extensive literature on construct validity contains persistent calls to deepen our understanding of the relationship among constructs and to continue improving construct validity methods (e.g., MacKenzie et al. 2011).

Extant construct validation relies on various statistical and empirical methods. These techniques seek to understand and evaluate the similarity between groups, objects, or responses and contribute to the IS literature by providing better-validated items, constructs, and theories. As part of such techniques, the researcher must assess that the constructs under evaluation are distinct. This distinction is easy to address at the local level but currently unassailable at the global level. We are aware of no major innovations in this area during the last 50 years.

Other validities are sometimes touted as construct validity approaches, including content validity, nomological validity, and criterion-related validity (Messick, 1998). However, two primary techniques have been developed and are widely used in IS and other areas for construct validation: Multi-trait-multimethod matrix (MMTM) (Campbell & Fiske, 1959) and confirmatory factor analysis (Jöreskog, 1969). Despite many advantages, they share four key limitations. First, current methods mainly operate within the local construct network. Second, the methods require pre-existing respondent data. Third, most methods are based on statistical properties of data, rather than semantic or lexical properties. The names, definitions, and indicators for such constructs are arguably linguistic entities, even if the derived relationships may not be (though some findings indicate otherwise, c.f., Gefen and Larsen, 2017; Arnulf et al., 2019). Finally, the popular construct validation approaches are laborious and do not scale (e.g., card sorting, pre-test). We seek to address these limitations as we develop our approach.

Problems in Current Practice

Reliance on unscalable data-gathering techniques such as surveys (Revilla and Ochoa 2017) is a fundamental limitation of current validation practices. Criterion-related validity, which encompasses both predictive and concurrent validity, is a crucial aspect of construct validation, assessing the extent to which test scores correlate with relevant outcome measures. However, pragmatic and theoretical challenges limit its incorporation into construct validation processes. Conducting criterion-related validity studies requires access to relevant outcome data, which may not always be readily available or involve lengthy follow-up periods to collect, particularly for predictive validity. This can significantly delay the validation process and increase its complexity and cost. Moreover, identifying suitable and universally accepted criterion measures is challenging, as such measures must be theoretically justified and practically obtainable. Finally, such measures should never be collected using the same research mode because common method variance creates false relationships, even for our most frequently researched theories (Sharma et al., 2009)

Construct validations are capped by human capabilities that have profound implications for the validity of our constructs. Hinkin and Tracey (1999) found that even major, high-attention, and carefully validated psychometric instruments suffer misspecified item memberships. Researchers employ similarity measures during the content validity evaluation (LeBreton and Senter 2008; MacKenzie et al. 2011) but operate without the benefit of a deeper understanding of the global construct network. Furthermore, in many techniques used for construct validation, a single-study context remains the default.

Given the previously stated inability of survey data collection methods to remain bias-free for large data collection and validation efforts and the quadratic growth of full mesh networks, such as construct and item networks, it becomes clear that current approaches have failed to account for validity accurately. For example, an examination of 10 constructs in a project produces knowledge on $\frac{n*(n-1)}{2}$ relationships, or 45 construct relationships. By comparison and by our accounting, a sliver of the discipline's journals, *MIS*

Quarterly, Information Systems Research, and *Journal of MIS,* between 2011 and 2020 published articles that reported 2,367 constructs, introducing approximately 2,800,000 construct relationships for which less than 0.085% of the construct pairs have been validated.[2] This percentage decreases quadratically the larger the set of constructs one considers so that at the discipline level, construct validity asymptotically approaches zero, a problem that increases with the research volume (Larsen et al., 2013).

However, constructs do not exist in isolation. In identifying a new construct, it is imperative to ensure that this construct does not overlap with other constructs. Constructs exist in a nomological network and must be connected to be useful. "The aim of construct validation is to embed a purported measure of a construct in a nomological network, that is, to establish its relation to other variables with which it should, theoretically, be associated" (Westen and Rosenthal 2003, p. 608). For such exercises to be meaningful, researchers must *first "specify how the construct will differ from other potentially similar constructs*" (Schwab 1980, p. 25). This statement acknowledges the importance of carefully evaluating the overlap between constructs. An approach that would allow the unification of evidence for constructs that address the same latent construct could turn the quadratic growth of relationships around, for example, for the example of the 2,367 constructs in a 10-year period, if most of these constructs were found to represent even 300 latent dimensions, the number of relationships the discipline would have to manage would go from 2.8 million to 45,000, a reduction so drastic that it would make IS research viable as an incremental science.

Proposed Solution

We propose a novel approach to address the construct validation problem: global construct validation. Unlike the prevailing approaches, ours goes beyond local validation testing, that is, testing of a set of constructs in a study against each other. Instead, it evaluates the construct validity in the focal study in the context of the common factors derived from some of the most popular constructs in the IS discipline over a 40+ year period. While we do not propose to replace the previous approaches, which we term local validation, we argue that a global approach ensures that the constructs in a study are those claimed relative to the global construct network in a discipline and measure this phenomenon at the construct level. Therefore, we take the name, definition, and indicators of each construct in a study into consideration and demonstrate how each construct loads or cross-loads against other constructs' latent dimensions in the whole discipline, allowing a researcher to address weaknesses in the construct identity, definitional accuracy, and indicator specification even before collecting data.

As we discuss the process of creating the global construct validity application, it is essential to note that this paper is the culmination of a 15-year project where the creation of this application has been the primary goal. After attempting many times and often being defeated by the lack of accuracy of language models and mathematical incompatibilities of the components, the recent introduction of large language models and our experimentation to solve the inherent incompatibilities of language models and PCA finally combined to solve this longstanding problem. The resulting model and application have the potential to move the IS discipline a step closer to an operational implementation of Cronbach and Meehl's (1955) vision of implicit construct definitions by more fully revealing the latent constructs in the IS nomological network and a method for specifying whether a new article builds on these latent constructs.

We next describe the model's development and formative validation before moving to summative validation. The approach draws on design science (e.g., Hevner et al. 2004), but the goal is not to create explicit design propositions but rather to solve a wicked problem in a replicable and extensible way and carefully evaluate the results using contemporary design validation approaches (Larsen et al., 2020).

DEVELOPING A MODEL FOR GLOBAL CONSTRUCT VALIDITY

Lodestar Application

We first share exemplar screenshots of the website of our proposed Lodestar application to support the reader's understanding of the pipeline that led to the application. The system was designed to validate a set of user-provided constructs against each other and a set of existing constructs in the discipline. The user is asked to provide the construct names, definitions, and indicators in their focal study and upload them. Once

[2] These numbers represent the total pairs in each paper (12,609) as a percent of the maximum pairs across the papers (2,800,161).

they have provided this information, their constructs appear as rows in the matrix. Figure 2 shows one of the demo papers in our system, Hoehle *et al.* (2022). Their first construct, named *Continued shopping intentions,* was defined as "The extent to which a respondent intends to continue shopping for Target's products" and operationalized through indicators such as "I intend to continue shopping for Target's products."

In the application, this construct is shown to load on latent dimension *4: Behavioral Intention to Use*, in Lodestar defined as "The degree to which an individual plans or commits to using a specific technology system or tool in the foreseeable future." It also loaded on dimension *207: Continued Shopping Intentions*, defined as "The extent to which a person plans to persist in purchasing products from a specific vendor or brand." This seems reasonable, given that an intention to buy from a website is both an intention to purchase and an intention to use the technology. Yet, both loadings slightly failed our entropy test, as explained later, suggesting that the authors might have benefitted from examining their construct before data collection. Their two types of *expected compensation* constructs loaded on dimension *483: Compensation Expectations*, which also seems reasonable.

For the most part the paper fared well in the analysis except for both *procedural justice* and *distributive justice* constructs loading on dimension *33: Procedural Justice*. If available to the authors before data collection, they may have considered more clearly distinguishing these constructs from each other. Another cross-loading that might have been useful to the authors before their data collection was that their construct *distributive justice* cross-loaded on dimensions *33: Procedural Justice* and *116: Fairness Of Outcomes*. Did they really measure distributive justice? This would at least be worth a discussion among the authors.

These examples show the value of moving from local validation (the traditional approach, only for constructs within a focal paper) to global validation, where information is provided not only about the relationships between constructs in the focal paper but also about the other disciplinary constructs. This was done by examining an uploaded set of constructs for cross-loadings against multiple latent dimensions for the research domain, which, if found, would suggest poor global construct validity. Next, we describe how we detected the latent dimensions.

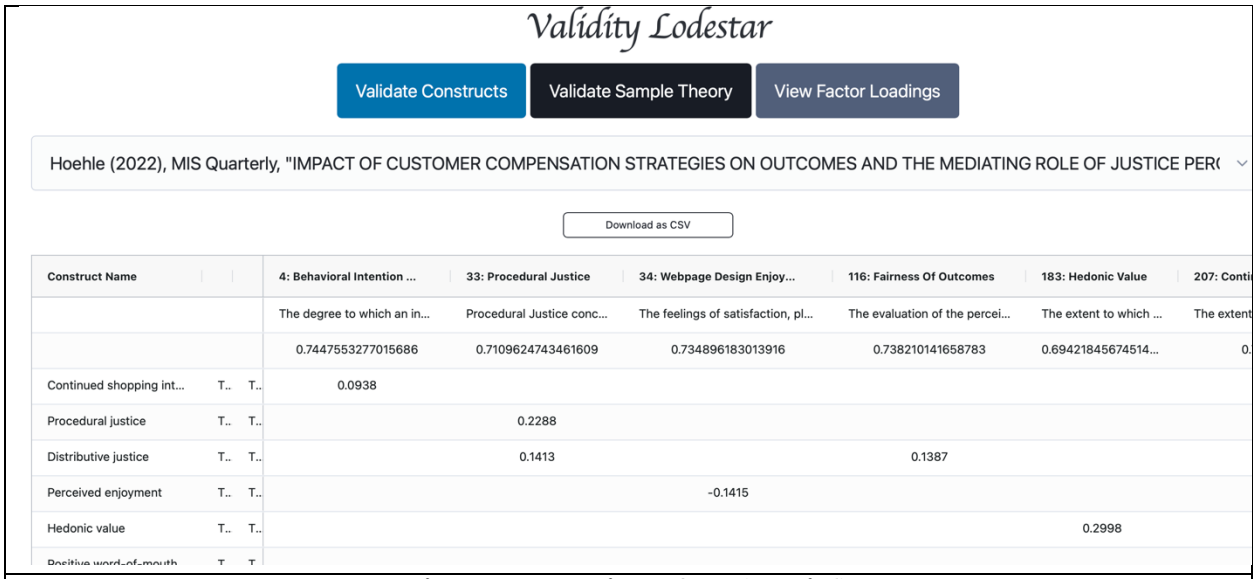

Figure 2. User interface (partial)

Before we explain the science behind our novel artifact, we start with a small cut-out of our whole global validity matrix, with the first eight latent dimensions outlined in Figure 3. There are a total of 550 latent dimensions (columns), each the result of a principal components decomposition of over 4,000 constructs published in top IS journals. Here, we show a sample of the loadings of the first four constructs and the dimensions on which they load. For example, while we show only five constructs that load on dimension *1: Technology Usefulness*, the full loading matrix contains 56 rows of constructs with names for the most part

being (perceived) usefulness, but also usefulness constructs with names such as *performance expectancy, job fit, performance impact of computer systems, performance,* and *utility for work-related use*.

The rows show the constructs and their definitions. To produce the matrix in Figure 2, uploaded constructs are projected into this matrix, and only those latent dimensions that a user's constructs load on above a cutoff are displayed. The diagonal values show the loadings of each construct on the left on the latent dimensions represented as columns. The numbers are lower than most PCA users may be used to, but is as expected for a large matrix like this (4,419 constructs × 550 latent dimensions). The negative vs. positive values are irrelevant, given that each set of constructs that load on a latent dimension are either positive or negative. The whole matrix is available here: (Larsen, Kai R. et al., 2024).

					1: Technology Usefulness	2: Computer Efficacy	3: System Ease Of Use	4: Behavioral Intention To Use Technology	5: Strategic Sourcing Intent	6: Trust In Web Vendors	7: Social Influence	8: System Support
Row ID	Construct ID	Construct Name	Construct Definition	Construct Items	The extent to which an individual believes that utilizing a specific	An individual's belief in their ability to successfully complete specific	The extent to which an individual believes that interacting with a specific	The degree to which an individual plans or commits to using a specific	The proactive intention of a firm to engage in sourcing specific resources and	The extent to which consumers believe that web vendors and their websites are	The degree to which an individual perceives that important others	The degree to which individuals perceive that organizational and technical support,
1	136	Perceived usefulness	The degree to which a persor	my effectiveness in	0.136							
2	74	Perceived usefulness	The degree to which a persor	would enable me to	0.1346							
3	33145	Perceived usefulness	The prospective user's subje	improves my productivity	0.1315							
4	27280	Perceived usefulness	The degree to which a persor	improve my performance	0.1291							
5	376	Perceived usefulness	The prospective user's subje	my job performance. ////	0.1288							
57	89529	Computer self efficacy	An individual's belief about h	using the hospital		-0.1534						
58	90564	Excel self efficacy	A learner's judgments of thei	//// I cannot yet use Excel		-0.15						
59	25035	Computer self-efficacy	Users' prior beliefs regarding	using the software if there		-0.1467						
60	89329	Computer self efficacy	The judgment of one's ability	using the software		-0.1466						
61	36153	Self-efficacy	A judgement of one's ability t	function if there was no		-0.1456						
109	23855	Perceived ease of use	The degree to which an indiv	cumbersome to use. //// It			0.1273					
110	271	Perceived ease of use	The degree to which a persor	system is clear and			0.1269					
111	23730	Perceived ease of use	The extent to which a person	system is clear and			0.1266					
112	119	Perceived ease of use	The degree to which a persor	system is clear and			0.1257					
113	33225	Perceived ease of use	The degree to which a persor	easy for me. //// I find it			0.1245					
172	27296	Behavioral intention	Users' intentions to use the s	in my course. //// I intend				0.1426				
173	25955	Behavioral intention	Intentions to use informatior	this term. //// I intend to				0.1425				
174	23840	Behavioral intention to use	Individual's intention to use t	spreadsheet] rather than				0.1422				
175	25143	Behavioral intention	Behavioral intention to use t	for communicating with				0.1418				
176	24011	Behavioral intention to use	Behavioral intention to use a	my studies. //// I intend to				0.1409				
226	82291	Replacement intention	A belief by those who are res	system with a competing					-0.2305			
227	94251	Replacement intentions	Indicate that an organizatior	system with another					-0.19			
228	95832	Multihoming Intent	The developer's intent to por	near future, to port this					-0.1815			
229	27206	Intention to reuse e-customer service	Users' behavioral intentions	online complaint, similar					-0.1676			
230	80918	Software-as-a-service continuance int	Continued information syste	using the SaaS-based					-0.1529			
231	26029	Firm's strategic intent	Firm's intent to use sourcing	executing this process. ////					-0.1068			
232	26691	Trusting beliefs in linker	Whether the consumer belie	can be relied upon to be						0.2492		
233	26695	Trusting beliefs in linkee	Whether the consumer belie	be relied upon to be						0.2393		
234	27187	Trusting beliefs in information technolo	The degree to which users pl	competent and effective in						0.2148		
235	27321	Trust in e-government Web site	Trusting beliefs that an e-gov	trustworthy. //// This Web						0.2114		
236	21811	Trust	A user's beliefs about the rel	confidence in the						0.1986		
255	26068	Social influence	The extent to which users be	to me would want me to							0.1892	
256	106	Social influence	The degree to which an indiv	behavior think that I							0.189	
257	93549	Social Influence	The degree to which an indiv	behavior think that I							0.1855	
258	81406	Social influence	The extent to which consum	to me think that I should							0.1779	
259	50053	Interpersonal influence	The relative influence of inte	following source influence							0.1749	
284	92085	Facilitating conditions	The degree to which an indiv	necessary to use the								0.2268
285	95617	Facilitating Conditions	The degree to which an indiv	necessary to use the								0.2266

Figure 3. First eight constructs of the 550-dimensional discipline-wide validation matrix

We now describe the pipeline that leads to the matrix in Figure 3, which we then project new constructs into to evaluate global construct validity, as depicted in Figure 2.

Design Overview

Figure 4 outlines the Lodestar design pipeline. As with most innovations, there are many candidate design options in each pipeline component, requiring formative criterion evaluations throughout. We employed the Human-in-Loop approach to pick the optimal combination of parameters. Each pipeline element accepts gradually transformed information on a construct, starting with its name, definition, and indicators to evaluate its global construct validity. Each pipeline element is outlined as a subsection in the rest of this section. We start with a corpus of constructs, finetune several candidate LLMs with the help of a construct taxonomy, and employ principal components analysis for dimensionality reduction with rotation before employing a human-in-the-loop design to find the optimal combination of hyperparameters, including dimensionality.

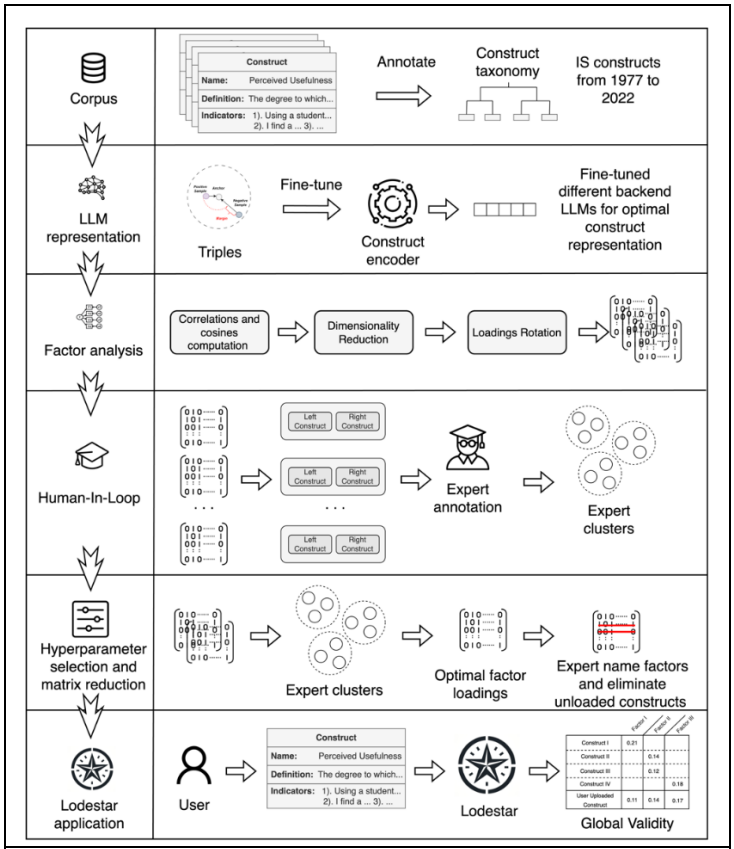

Figure 4. Application Design Framework

Corpus and Taxonomy

The original data for this project was provided by the Human Behavior Project (HBP) at the University of Colorado. They collected constructs from *MIS Quarterly* and *Information Systems Research* from 1983 – 2009. The project conducted a categorization of these constructs and made them available as part of the article on the Construct Identity Fallacy (Larsen & Bong, 2016). The process for data collection and categorization is provided in Appendix B of Larsen and Bong (2016).

For this project, we replicated the procedures described in Larsen and Bong (2016), including training and auditing. We expanded the data collection conducted by HBP to include the Journal of MIS and the years 1977 to 2020 for all three journals (from journal inception up to 2020) and partial data for 2021-2022. The name, definition (when available), and all reported indicators for each construct were collected and blended with the original data in a new database. Table 1 reports on the characteristics of the data collected, split by journal and year range into the training, validation, and holdout samples. Each is split by year to enable true predictive validity evaluation. Only constructs for which the original author reported a definition and at least one indicator were included in the training, validation, and holdout samples.

Journal	Training	Validation	Holdout	Inference
MIS Quarterly [id: 3]	1977-2005: 665 constructs	2006-2007: 190 constructs	2008-2009: 187 constructs	1977-2022: 1,552 constructs
Information Systems Research [id: 29]	1990-2005: 471 constructs	2006-2007: 103 constructs	2008-2009: 108 constructs	1990-2021: 935 constructs
Journal of MIS [id: 1620]				1984-2020: 1,932 constructs[3]

Table 1. Training, Validation, Holdout, and Inference Data.

LLM Representation

The objective of the Construct Encoder, the core model in the LLM representation component, is to learn the semantic representations of constructs from the same pool so that representations of constructs are closer in Euclidean space than representations of constructs from different pools.

Construct Encoder. Rather than the classification, generation, and other machine learning tasks; we focus on how to get embeddings of the given constructs that represent the underlying semantic information in their names, definitions and indicators. To align with our objective, we leverage Deep Metric Learning (Kaya & Bilge, 2019) to fine-tune Large Language Models (LLMs). Deep Metric Learning is a learning strategy to learn the hidden representation of objects based on the distance among them, which is aligned with the objective of the Construct Encoder in Lodestar. More details about leveraging this strategy to fine-tune LLMs as the Construct Encoder are demonstrated as follows.

As shown in Figure 5, in the training process, we chose triples (sets of three) of constructs as inputs consisting of one anchor point (a randomly selected construct), one positive point (a construct categorized as similar), and one negative point (a construct categorized into a different category from the first two). This is different from regular supervised learning methods, which take one data sample with its label as the input in training. Each data point in the triple refers to one construct with its name, definition, and indicators. Then, we use three identical backend LLMs to embed the three constructs in one triplet and compute the triplet loss based on the cosine distance among the three embeddings to optimize the parameters in the LLM. The three identical LLMs share the same parameters and get the same update during the gradient descending optimization procedure (Ruder, 2016). The objective of the optimization procedure is to minimize the triplet loss. Finally, the LLM with optimized parameters is the construct encoder we use to embed the constructs.

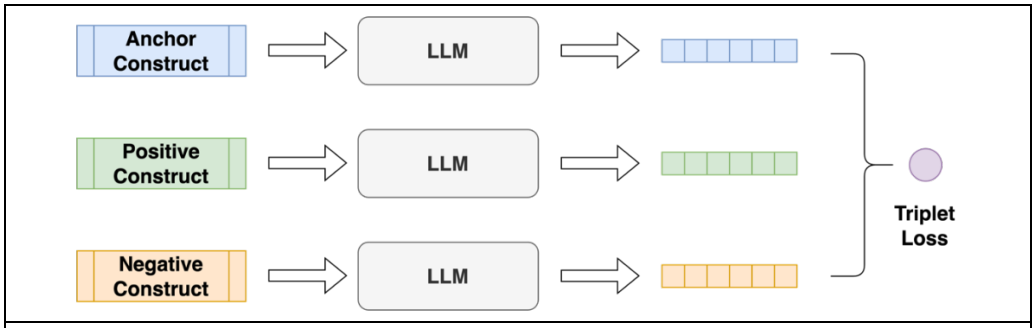

Figure 5. Illustration of Deep Metric Learning to Fine-Tune the LLMs.

Triplet Loss. The triplet loss is one of the standard loss functions in Deep Metric Learning, and this section will focus on the triplet loss used to fine-tune the parameters of LLMs. As mentioned, the triples of

[3] This set by accident contains 260 papers from the Journal of Applied Psychology. These will be removed in the next version of the paper and are only expected to improve the performance of the model.

constructs are the input for training the model, and we will demonstrate how to build the construct triples first. As illustrated in Figure , the objective of the triplet loss is to minimize the vector representation's distance between the anchor point and the positive point and maximize the distance between the anchor point and the negative point in the Euclid space. Finally, the formula of the triplet loss for one training data point is defined as follows:

$$loss = max\big(d\big(P_{pos}, P_{anchor}\big) - d\big(P_{neg}, P_{anchor}\big) + margin, 0\big)$$

Where the distance function $d(X,Y)$ is defined as the cosine distance and the formula is $d(X,Y) = \frac{XY}{||X||\,||Y||}$, the P_{anchor}, P_{pos} and P_{neg} refer to the anchor point, positive point and negative point in the triple, respectively. The margin is a predefined hyperparameter in triplet loss that encourages the model to differentiate the positive point from the negative point by a specified distance, i.e., the distance between the positive point and the negative point needs to be larger than the margin.

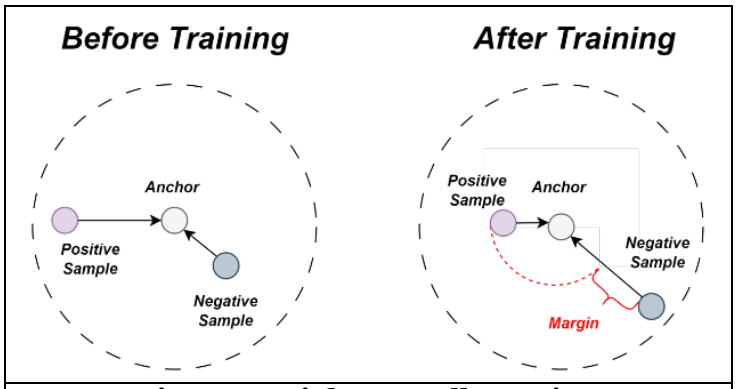

Figure 6. Triplet Loss Illustration

The model's performance based on triplet loss is largely based on the choice of negative points because there are too many candidates for potential negative points in the corpus (Xuan et al., 2020). The strategy for chooing the negative samples in each triple is derived from the Negative Mining Strategy (Schroff et al., 2015) and define three different types of negative points: *Hard Negative*, *Semi-Hard Negative* and *Easy Negative* as illustrated in Figure 7. Formally, the hard negative points refer to those negative points closer to anchor points in the Euclidian space than positive samples, and we name them as hard negative points because the Construct Encoder fails to encode them properly and mis-classify them as positive samples based on cosine distance. The semi-hard negative points refer to those points further than positive samples but less than the distance of positive samples plus the margin. We name these points as semi-hard samples because the Construct Encoder can distinguish them from the positive sample, but the distance is still not large enough, and the triplet loss is positive in a triple with semi-hard negative points. Lastly, the easy negative points refer to those points with further distance than the positive points' distance plus the pre-defined margin. Mathematically, the relationship between the hard negative, semi-hard negative, easy negative, and positive points can be defined as follows:

$$d\big(P_{hard\_neg}, P_{anchor}\big) \le d\big(P_{hard\_neg}, P_{anchor}\big) \le d\big(P_{semi\_hard\_neg}, P_{anchor}\big) \le d\big(P_{hard\_neg}, P_{anchor}\big) + margin \le d\big(P_{easy\_neg}, P_{anchor}\big)$$

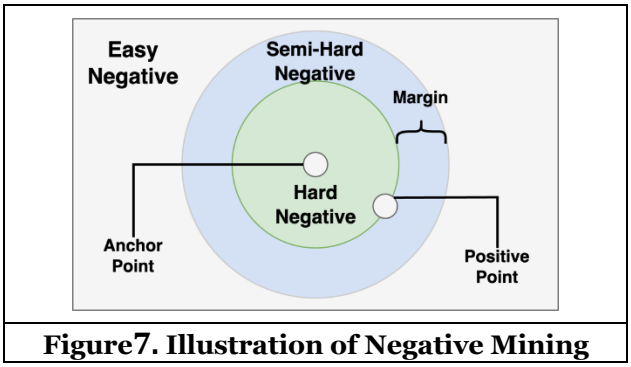

Figure7. Illustration of Negative Mining

For one anchor point, we build three triples with hard-negative points, three triples with easy-negative points, and three triples with semi-hard negative samples as the input for fine-tuning the backend LLM (Xuan et al., 2020). After each epoch of training, we update the representation of constructs based on the fine-tuned LLM from the last epoch and then re-compute the distance to update three types of negative points for each anchor point, which are chosen to construct the triples for the next training epoch. For the first training epoch, we use the pre-trained LLM without any fine-tuning process to construct triples because there are no fine-tuned versions of the backend LLM.

Backend LLM. We choose candidate LLMs from the Massive Text Embedding Benchmark (MTEB) leaderboard,[4] which aims to benchmark the text embedding performance of LLMs through 8 embedding tasks covering 58 datasets and 112 languages (Xuan, Stylianou and Pless, 2020). Considering both the number of parameters for the feasibility of fine-tuning and the performance on the leaderboard for efficacy of construct representation, we choose four backend LLMs as candidates including: 1. Salesforce/SFR-Embedding-Mistral,[5] 2. WhereIsAI/UAE-Large-V1,[6] 3. avsolatorio/GIST-large-Embedding-v0,[7] and 4. llmrails/ember-v1.[8] We compared these to zero-shot GPT-4 and few-shot GPT-4.

Training Details. We train for 10 epochs, select 500 anchor points in each epoch, and then select the fine-tuned LLM with highest F_1-score on the validation set as the final Construct Encoder in Lodestar. To efficiently fine-tune the backend LLMs, we adopt the LoRA (Hu et al., 2021) strategy and only fine-tune some parameters of the LLMs while freezing the rest of parameters during the training process. In this case, roughly 20 million parameters out of 7 billion were updated.

Formative Criterion Evaluation. In this section, we leverage the holdout dataset to evaluate the performance of the Construct Encoder with different backend LLMs. Because the test dataset is unbalanced, we evaluate the performance using the Macro-F_1, Precision, and Recall metrics. The Macro-F_1, Precision, and Recall are computed based on the regular F_1, Precision, and Recall for positive and negative classes, respectively, and average the metrics of the two classes.

As shown in Table 2, the best LLM is Salesforce/SFR-Embedding-Mistral, with the highest value in most metrics, including the Macro-F_1, Macro-Precision, and ROC-AUC. However, the differences among the LLMs are not significantly large. All exhibit relatively high performance in these metrics, which demonstrates the LLMs' capability to represent the construct information. This also suggests that our design could benefit from using multiple models or, to avoid designing a slow artifact, using a human-in-the-loop to pick the best LLM and hyperparameters.

[4] https://huggingface.co/spaces/mteb/leaderboard
[5] https://huggingface.co/Salesforce/SFR-Embedding-Mistral
[6] https://huggingface.co/WhereIsAI/UAE-Large-V1
[7] https://huggingface.co/avsolatorio/GIST-large-Embedding-v0
[8] https://huggingface.co/llmrails/ember-v1

LLM Name	Threshold	Macro-F1	Macro-Precision	Macro-Recall	ROC-AUC
bert-base-uncased	0.62	0.75	0.72	**0.80**	0.91
roberta-base	0.63	0.76	0.74	0.78	0.90
allenai/specter-2	0.64	0.74	0.74	0.76	**0.93**
llmrails/ember-v1	0.62	0.76	0.75	0.77	0.85
avsolatorio/GIST-large-Embedding-v0	0.61	0.76	0.75	0.77	0.89
WhereIsAI/UAE-Large-V1	0.62	0.76	0.76	0.76	0.87
Salesforce/SFR-Embedding-Mistral	0.63	**0.79**	**0.80**	0.78	**0.93**

Table 2. Large Language Model Representation Performance

Factor Analysis. We focus on explicating the hidden factors behind the IS constructs in the Factor Analysis element. There are three sub-steps, including 1) correlation computation, 2) dimensionality reduction, and 3) rotation. Similar to the four candidate backend LLMs in the Construct Encoder design, each component in the dimension reduction element may include multiple potential candidate algorithm options. We detail the choice of an optimal parameter option in the Human-In-Loop pipeline element.

Correlation and Cosine Similarity Computation. We first used the finetuned models to infer on all the constructs in the 'Inference data' in Table 1 to have a full set of vectors for all available constructs. We compute the similarity or correlation matrix as the first step. Such a matrix represents the relationship among the constructs, which can be leveraged to examine the underlying hidden factors via factor analysis. In the inference stage, we only compute the similarity or correlation vector for each new construct against the original set of constructs rather than computing the matrix again for each newly uploaded construct. Doing so avoids having to re-estimate the parameters in the dimensional reduction and Rotation components so that the expert-labeled hidden factors during the development process stay the same.

Human-in-the-Loop Hyper-Parameter Examination

The common approaches to dimensionality selection for PCA with varimax rotation include the Kaiser criterion, where components with eigenvalues less than 1 are discarded. Alternatively, Cattell scree test is sometimes used (Rust et al. 2021), but consistent cutoff rules are unavailable when dimensionalities are expected to be in the hundreds. Both approaches were inappropriate for the dimensionality section due to simultaneous selection between four language models: PCA based on cosine vs. correlation. We therefore tested all combinations with dimensionalities in steps of 50 (50, 100, 150, etc.) up to 1,000 dimensions. This yielded 160 rotated matrices. To select the optimal matrix among them as the final factor loadings used in our proposed lodestar system, we adopt the human in loop hyper-parameter examination procedure as illustrated in the following.

We adopted an algorithm, which will be demonstrated later in this section, to compute the similarity scores between two constructs based on the 160 factor loadings matrices. And then a website was developed for expert annotation which presented these pairs of constructs to an expert in the order of the similarity. Any time the expert agreed that the two constructs should load on the same dimension, those two constructs were clustered together. An examination of the concordance between the algorithm similarity score and the average decision of the expert for 3,291 expert decisions split into 33 groups ordered by similarity demonstrated a linear decrease in the expert coding a pair of constructs as concordant as the similarity score decreased. The concordance had an R^2 of .94, suggesting that the combined similarity based on the 160 rotated matrices performed much like the expert coder. At the end of the coding process, a dataset consisting of 9,761,571 construct pairs with expert codes was derived.

The similarity algorithm is motivated by TF-IDF, which is commonly used in the field of Information Retrieval, and is described as follows:

$$tf_{i,j}^{a} = \frac{f_{i,j}^{a}}{\sum_{m=1}^{n} f_{i,m}^{a}}$$

$$idf_{i,j}^{a} = ln\left(\frac{\sum_{k=1}^{4419} f_{k,j}^{a}}{f_{i,j}^{a}}\right)$$

$$tfidf_i^a = \left[tf_{i,1}^a \times idf_{i,1}^a, tf_{i,2}^a \times idf_{i,2}^a, \dots, tf_{i,n}^a \times idf_{i,n}^a\right]$$

Where $f_{i,j}$ denotes the i-th construct's value of loading on j-th factor, n denotes the number of factors in the specific factor loadings matrix. The $tf_{i,j}$ measures the ratio of loading value of i-th construct's on j-th factor relative to its factor loadings on all factors. The $idf_{i,j}$ measures the ratio of loading value of i-th construct's on j-th factor relative to all constructs' factor loadings on j-th factor. And $tfidf_i$ denotes the TF-IDF vector of i-th construct. We then create the similarity score matrix $S^a \in \mathbb{R}^{4419\times4419}$ derived from the a-th factor loading matrix via the cosine similarity following the formula:

$$S_{i,j}^a = cosine\_similarity\left(tfidf_i^a, tfidf_j^a\right)$$

Lastly, we average all the similarity score matrices derived from all 160 factor loading matrices to get the final similarity score matrices of construct pairs for expert annotations.

In the final similarity matrix, we get 9,761,571 ($\frac{4419\times(4419-1)}{2}$) construct pairs after removing both diagonal and duplicate construct pairs. To avoid evaluating all of these pairs, constructs in a cluster are required only one connection into any other construct in that cluster, which means that if construct A already exists in cluster 1, we skip any pairs containing construct A and any constructs in cluster 1; also, if experts have already disagreed any constructs in cluster 2 should load on the same dimension with any constructs in cluster 1 (where construct A exists) before, we skip any pairs containing construct A and any constructs in cluster 2.

An expert with 30 years of experience with behavioral IS constructs acted as coder. After evaluation and resolution of the first 13,000 pairs, we evaluate 160 rotated factor loadings with four different backend LLMs, correlation vs similarity and number of common factors from 50 to 1,000 in the step of 50. For the rotated factor loadings, we only use the eigenvectors from PCA as unrotated factor loadings and varimax as the rotation method. Given the incomplete annotation results, we propose a metric based on micro-recall and adjust it by the number of cross-loaded constructs, the number of constructs without loading on any common factors, and the average number of constructs in each common factor. Given a rotated factor loading matrix, the specific formula is defined as follows:

$$micro - recall = \frac{\sum_{i=1}^{m}\frac{1_i^{same}}{cl_i} + \sum_{i=1}^{n}\frac{1_i^{not\_same}}{cl_i}}{m+n} \times \left(1 - \frac{1-nn}{4419}\right) \times \frac{10}{l}$$

$$1_i^{same} = \begin{cases} 1 & if\ constructs\ in\ i-th\ pair\ with\ label\ "same"\ load\ on\ same\ factor \\ 0 & if\ constructs\ in\ i-th\ pair\ with\ label\ "same"\ do\ not\ load\ on\ same\ factor \end{cases}$$

$$1_i^{not\_same} = \begin{cases} 1 & if\ constructs\ in\ i-h\ pair\ with\ label\ "not\ same"\ do\ not\ load\ on\ same\ factor \\ 0 & if\ constructs\ in\ i-th\ pair\ with\ label\ "not\ same"\ load\ on\ same\ factor \end{cases}$$

Where m and n denote the number of pairs annotated as "same" and "not same" respectively. cl_i refers to the number of factors where the constructs in i-th pair load on, so we punish those constructs loading across different factors. *nn* refers to the number of constructs that do not load on any factors, and we punish those cases where lots of constructs do not load on any factors. *l* refers to the average number of constructs that load on each factor, and we punish those cases that too many constructs (more than 10) are loaded in the same dimension.

After evaluating these 160 rotated factor loadings with different cut-off points from 0.05 to 0.5 with the step 0.01, we find the optimal combination is: *llmrails/ember-v1*, *correlation*, *550 factors* and *0.08 cut-off point*. The micro-recall of the optimal factor loadings matrix is 0.5315.

To guide researchers when evaluating their own constructs, we leverage *entropy* (Shannon, 1948) and *Jensen-Shannon (JS) divergence* (Menéndez et al., 1997) from Information Theory to evaluate the loadings matrix of the user's uploaded constructs. Entropy measures the uncertainty of a given system, with higher entropy indicating greater uncertainty, i.e., more information. In the context of GCV, if a focal construct loads on multiple factors or if multiple constructs load on one latent dimension the uncertainty of the focal

construct or factor will be large, leading to higher entropy. Consequently, we compute the entropy for each uploaded construct and its loaded factors. Low entropy indicates that the uploaded construct is appropriately loaded without cross-loadings or that multiple uploaded constructs do not load on the same latent dimension. We use one standard deviation (std) above the average entropy computed based on the set of 72 constructs and 74 latent dimension entropies in the test dataset, respectively, as the threshold, which indicates an appropriate construct or factor. The entropy thresholds are *6.2732* and *0.7422* for construct and latent dimension, respectively, and any construct or latent dimension below this threshold can be considered as appropriate. JS divergence measures the similarity between two distributions, and we use this metric to assess the similarity of two constructs' loadings. Specifically, we apply the softmax function to convert the loadings of each construct into a probability distribution and then compute the pairwise JS divergence. In the context of GCV, a high JS divergence indicates that the two constructs are dissimilar, thus measuring distinct attributes. Similarly, we use one std below the average JS divergence computed based on the set of 530 construct pairs in the test datasets as the threshold. The threshold value is *0.0025,* and any construct pairs larger than this threshold can be considered as appropriate.

EXAMPLE USE CASE

We examined several papers published in 2021 and 2022 using the interface shown in Figure 2, allowing users to download the results as a .csv file. We use this functionality for the paper "Managing Collective Enterprise Information Systems Compliance," published in MIS Quarterly (Zhou et al., 2022). The paper contained six core constructs: *relational capital, structural capital, cognitive capital, collective enterprise information systems compliance, social context,* and *performance management context*. These constructs, their definitions, and a sample of their indicators are shown in the first three columns of Table 3. These six constructs load on five of the latent dimensions, 267: Task Impact on Work, 317: Relational Capital, 428: Cost of Noncompliance, 443: Interdepartmental Relations, and 513: Cognitive Capital.

		267: Task Impact On Work	**317: Relational Capital**	**428: Cost Of Noncompliance**	**443: Interdepartmental Relations**	**513: Cognitive Capital**	**Construct Entropy**
Construct Name	**Construct Definition**	The extent to which the task impacts the work and productivity of others within the organization, evaluated before and after the	Relational Capital refers to the value derived from established interpersonal relationships characterized by trust, respect,	The perceived negative consequences associated with not adhering to required information security policies.	The extent to which various departments within an organization such as functional units and information technology	Cognitive capital refers to the resources available within an organization or team that enable effective communication, shared	
	Dimension entropy	**0.6424**	**0.6129**	**0.5571**	**0.6727**	**0.5408**	
Performance management context	The extent to which the organization sets performance goals to motivate its members to meet expectations and strive for more ambitious goals.	0.0815					**6.0918**
Relational capital	The extent to which functional unit and information technology unit developed a solid social relationship.		0.1923		0.1254		**6.2568**
Collective enterprise information systems compliance	The aggregation of individual-level enterprise information systems compliance behaviors in a functional unit. Individual-level enterprise information systems compliance is measured as the extent to which an employee in a functional unit complies with EIS policies.			-0.354			**6.1047**
Structural capital	The extent to which functional unit and information technology unit developed a solid social relationship.				0.2581		**6.1069**
Cognitive capital	The extent to which functional unit and information technology unit developed a solid social relationship.					-0.413	**6.1173**
Social context	The extent to which management systems, processes, and actions provide trust and support to employees.						**6.1739**

Table 3. Global Construct Validity Use Case

While much work is still remaining to understand how our tool should be used by psychometricians, we observe that the relational capital construct loads on the two latent dimensions 317: Relational Capital and 443: Interdepartmental Relations, likely because it measures an organizational-level construct by elevating individual-level terms such as trust and respect to the department level. Also, both relational capital and structural capital load on the same dimension, 443: Interdepartmental Relations, likely because both constructs are about close interactions, trust, and the ability to interact. This should not be taken as a sign that the study is flawed but rather that these constructs, as measured, may be closer to each other than expected. In fact, our new entropy measures did not suggest any problem with this study. Nevertheless, the authors could have examined the names, definitions, and items and addressed the cross-loadings by adjusting the statistical model or by considering a different set of definitions or measurement indicators. For example, could structural capital be measured in objective terms by examining the extent to which members of the departments are on shared committees? Alternatively, *how often* do specific individuals interact? Finally, one construct did not load highly on any latent dimensions. This is as expected in cutting-

edge behavioral research, as one expects new constructs to be introduced when addressing novel questions. Finally, no construct pair in this paper had JS Convergence scores below or equal to 0.0025, so they were considered appropriate.

EVALUATION

To evaluate Lodestar, an applicability check was conducted (Rosemann & Vessey, 2008). An applicability check examines whether an artifact is important, accessible, and suitable for the problem. We conducted the check at a regional IS conference and recruited 22 researchers to evaluate an earlier 500-dimensional solution that was not optimal.

We first evaluated the model validity of our artifact by randomly selecting latent dimensions and selecting up to five constructs that loaded the most highly on these dimensions. Each dimension was interpreted and given a name by GPT-4 (Achiam et al., 2023) based on a sample of constructs that loaded on that dimension. Each participant was matched with a random dimension and asked the extent to which the constructs belonged together under the name and definition provided by GPT-4. This introduced another potential point of failure, but it was considered worthwhile to get an early sense of our ability to automatically name dimensions. In all, the participants rated 100 constructs in 22 dimensions and agreed that 75% fit together under the GPT-4 provided name and definition. The participant comments suggested that disagreements came from both the GPT-4 naming (which was found to have lost a big part of its prompt during implementation) and the PCA allocation. While we are not aware of an appropriate inter-rater metric for our approach, we believe that the fact that 500 dimensions are provided means that two constructs rated as "same" would be highly unlikely to load on the same dimension by chance, suggesting that the 75% would be equivalent to a kappa score that would be rated by Landis and Koch (1977) at the high end of "substantial" agreement between the algorithm allocation and the expert rater.

After a demonstration of the artifact, 17 participants filled out a survey containing an open-ended question and a set of questions about artifact usefulness and their intention to use, applying Likert-type scales from Venkatesh et al. (2003). The responses, both qualitative and quantitative, indicated that participants considered the Lodestar to be important, accessible, and suitable for meeting the identified needs of the community and believed it would be useful for their research.

Rosemann and Vessey (2008) define importance as research "that meets the needs of practice by addressing a real-world problem … in such a way that it can act as the starting point for providing a solution" (p. 3). We consider the process of evaluating this aspect to be an instance of criterion validity. In our context, DSR researchers are practitioners in the real world who themselves need to validate knowledge claims about artifacts. Accessibility is a criterion validity that "encompasses whether the research is understandable, readable and focuses on results rather than the research process" (p. 3). Finally, suitability is defined as the extent to which the research can "[meet] the needs of practice" (p. 3), which we take to mean the extent to which DSR researchers view the framework as appropriate for the target context. These three evaluations all address model validities and context validity, given their evaluation in a setting similar to the real world.

Participants agreed that the framework is important for clarifying and providing structure to the increasingly complex landscape of constructs and their validation. Comments such as "a fantastic resource with so much promise," "the concept is good," and "very useful and insightful!" evidenced the artifact's importance. Other comments proposed improvements, the most popular being that the constructs for each latent dimension that user-uploaded theory loads on are provided. This could potentially solve the pairwise construct search problem detected by Li et al. (2020). Two other participants wanted additional functionality that would provide feedback on the impact of indicators on the overall loading patterns, which is planned for a future version of the system. Finally, one respondent stating that it will be "nice to have a "Map" and see where own constructs are located in the big picture."

Participants were surveyed about the usefulness of Lodestar ($\mu = 5.51$; $\sigma = 1.04$), a common applicability check test in DSR (e.g., Li et al., 2020; Lukyanenko et al., 2019). They were also surveyed about their intention to use the framework ($\mu = 5.68$; $\sigma = 1.17$), another criterion validity that doubles as an ecological validity when evaluated in the target setting. All participants indicated that they intended to use the application once it became available, though two researchers marked slight middle-positive scores. We

therefore concluded that the applicability checks established *model validity* and *criterion validity,* and added initial, if incomplete, evidence of *ecological validity*.

DISCUSSION AND CONCLUSION

Since Cronbach and Meehl introduced construct validity in 1955, the concept has diffused into almost all papers employing the psychometric method. Yes, the concept is unworkable because it focuses on the local nomological network. We offer the first step towards a solution to this problem by distinguishing between local and global construct validity and introducing the first artifact for global construct validity: *Validity Lodestar*. Yet, the artifact is no panacea, and much work remains to address construct validity for the discipline fully. Additional work, including open-sourcing the final application, which is itself based on open-source LLMs will reduce the black-box nature of the approach.

While the Lodestar application arguably stands as an artifact with "much promise," several improvements are planned before we plan to make it widely available to behavioral IS researchers. First, researchers have repeatedly found that PCA has deep abilities to produce an understanding of behavioral constructs. Nevertheless, although our Lodestar application provides PCA dimensions that make deep sense, this should not be taken as evidence that these constructs exist objectively. Our model distinguishes between the textual artifacts representing the constructs on which we finetuned it. However, researchers should remain skeptical of any construct that has not been shown to exhibit true predictive validity. Likewise, we have yet to demonstrate that the approach will work on previously unpublished constructs.

Second, to provide a direct construct validity solution, the user must be given feedback on their naming, definition writing, and individual indicators so that they can change these before conducting their work. It will be especially important to provide users with information on the effect of removing each indicator on the overall factor loadings. For example, does removing the indicators in the *usefulness* construct that focuses on the ease of doing the job lead to a representative construct and less cross-loading?

The focus of *global* construct validity is an individual discipline, and we argue through this paper and hands-on artifact that the IS discipline may use our tool to get its house in order. Yet, there are many disciplines that rely on the psychometric method. It is our hope that in the future, other disciplines may use our approach to refine its theoretical substrate.

Our work aims to enable behavioral researchers to stay innovative and publish their research without getting "in the way." Lodestar will enable such researchers to examine whether their new novel constructs load in the anticipated fashion and, if they do not, provide the information necessary to change its name, definition, or indicators. Articulating cross-loading patterns provides evaluative clarity and research efficiency, promotes the sharing of best practices, facilitates cumulative science, and is perhaps a key step in fully automated meta-analyses (e.g., Bosco et al., 2019). The goal is to allow behavioral researchers to move their science from focusing on the local construct network to the global construct network.

REFERENCES

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., & Anadkat, S. (2023). Gpt-4 technical report. *arXiv Preprint arXiv:2303.08774*.

Arnulf, J. K., Dysvik, A., & Larsen, K. R. (2019). Measuring semantic components in training and motivation: A methodological introduction to the semantic theory of survey response. *Human Resource Development Quarterly*, *30*(1), 17–38.

Bosco, F. A., Field, J. G., Larsen, K. R., Chang, Y., & Uggerslev, K. L. (2019). Advancing Meta-Analysis With Knowledge-Management Platforms: Using metaBUS in Psychology. *Advances in Methods and Practices in Psychological Science*, 2515245919882693.

Campbell, D. T., & Fiske, D. W. (1959). Convergent and discriminant validation by the multitrait-multimethod matrix. *Psychological Bulletin*, *56*(2), 81.

Cronbach, L. J., & Meehl, P. E. (1955). Construct validity in psychological tests. *Psychological Bulletin*, *52*(4), 281.

Gefen, D., & Larsen, K. R. (2017). Controlling for Lexical Closeness in Survey Research: A Demonstration on the Technology Acceptance Model. *Journal of the Association for Information Systems*, *18*(10), 727–757.

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design Science in Information Systems Research. *MIS Quarterly*, *28*(1), 75–105.

Hoehle, H., Venkatesh, V., Brown, S. A., Tepper, B. J., & Kude, T. (2022). Impact of Customer Compensation Strategies on Outcomes and the Mediating Role of Justice Perceptions: A Longitudinal Study of Target's Data Breach. *Mis Quarterly*, *46*(1).

Hoyningen-Huene, P. (1993). *Reconstructing scientific revolutions: Thomas S. Kuhn's philosophy of science*. University of Chicago Press.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., & Chen, W. (2021). Lora: Low-rank adaptation of large language models. *arXiv Preprint arXiv:2106.09685*.

Jöreskog, K. G. (1969). A general approach to confirmatory maximum likelihood factor analysis. *Psychometrika*, *34*(2), 183–202.

Kaya, M., & Bilge, H. Ş. (2019). Deep metric learning: A survey. *Symmetry*, *11*(9), 1066.

Landis, J. R., & Koch, G. G. (1977). The Measurement of Observer Agreement for Categorical Data. *Biometrics*, *33*(1), 159–174.

Larsen, K. R., & Bong, C. H. (2016). A Tool for Addressing Construct Identity in Literature Reviews and Meta-Analyses. *MIS Quarterly*, *40*(3), 529–551; A1–A21.

Larsen, K. R., Lukyanenko, R., Mueller, R. M., Storey, V. C., VanderMeer, D., Parsons, J., & Hovorka, D. S. (2020). *Validity in design science research*. 272–282.

Larsen, K. R., Voronovich, Z. A., Cook, P. F., & Pedro, L. W. (2013). Addicted to constructs: Science in reverse? *Addiction*, *108*(9), 1532–1533.

Larsen, Kai R., Yan, Sen, & Lukyanenko, Roman. (2024). *Global Construct Validity Matrix* [Dataset]. https://mega.nz/file/DaJByaga#EjB9JqReutYNIlqI6DRCOUx_Kx04OqKyOd8jRC-lCw4

Li, J., Larsen, K. R., & Abbasi, A. (2020). TheoryOn: A Design Framework and System For Unlocking Behavioral Knowledge Through Ontology Learning. *MIS Quarterly*, 1–55.

Lukyanenko, R., Parsons, J., Wiersma, Y., & Maddah, M. (2019). Expecting the Unexpected: Effects of Data Collection Design Choices on the Quality of Crowdsourced User-generated Content. *MIS Quarterly*, *43*(2), 634–647.

Menéndez, M. L., Pardo, J., Pardo, L., & Pardo, M. (1997). The jensen-shannon divergence. *Journal of the Franklin Institute*, *334*(2), 307–318.

Messick, S. (1998). Test validity: A matter of consequence. *Social Indicators Research*, *45*, 35–44.

Muennighoff, N., Tazi, N., Magne, L., & Reimers, N. (2022). MTEB: Massive text embedding benchmark. *arXiv Preprint arXiv:2210.07316*.

Rosemann, M., & Vessey, I. (2008). Toward improving the relevance of information systems research to practice: The role of applicability checks. *MIS Quarterly*, 1–22.

Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv Preprint arXiv:1609.04747*.

Schmitz, K., & Storey, V. C. (2020). Empirical test guidelines for content validity: Wash, rinse, and repeat until clean. *Communications of the Association for Information Systems*, *47*(1), 64.

Schroff, F., Kalenichenko, D., & Philbin, J. (2015). *Facenet: A unified embedding for face recognition and clustering*. 815–823.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, *27*(3), 379–423.

Sharma, R., Yetton, P., & Crawford, J. (2009). Estimating the effect of common method variance: The method—Method pair technique with an illustration from TAM Research. *MIS Quarterly*, *33*, 473–490.

Venkatesh, V., Morris, M. G., Davis, G. B., & Davis, F. D. (2003). User acceptance of information technology: Toward a unified view. *MIS Quarterly*, 425–478.

Xuan, H., Stylianou, A., & Pless, R. (2020). *Improved embeddings with easy positive triplet mining*. 2474–2482.

Zhou, J., Fang, Y., & Grover, V. (2022). Managing Collective Enterprise Information Systems Compliance: A Social and Performance Management Context Perspective. *MIS Quarterly*, *46*(1).