Organizing Tagged Knowledge: Similarity Measures and Semantic Fluency in Structure Mining

Thurston Sexton[1]
Systems Integration Division,
Engineering Laboratory,
National Institute of Standards and Technology,
Gaithersburg, MD 20871
e-mail: thurston.sexton@nist.gov

Mark Fuge
Department of Mechanical Engineering,
University of Maryland,
College Park, MD 20742
e-mail: fuge@umd.edu

Recovering a system's underlying structure from its historical records (also called structure mining) is essential to making valid inferences about that system's behavior. For example, making reliable predictions about system failures based on maintenance work order data requires determining how concepts described within the work order are related. Obtaining such structural information is challenging, requiring system understanding, synthesis, and representation design. This is often either too difficult or too time consuming to produce. Consequently, a common approach to quickly elicit tacit structural knowledge from experts is to gather uncontrolled keywords as record labels—i.e., "tags." One can then map those tags to concepts within the structure and quantitatively infer relationships between them. Existing models of tag similarity tend to either depend on correlation strength (e.g., overall co-occurrence frequencies) or on conditional strength (e.g., tag sequence probabilities). A key difficulty in applying either model is understanding under what conditions one is better than the other for overall structure recovery. In this paper, we investigate the core assumptions and implications of these two classes of similarity measures on structure recovery tasks. Then, using lessons from this characterization, we borrow from recent psychology literature on semantic fluency tasks to construct a tag similarity measure that emulates how humans recall tags from memory. We show through empirical testing that this method combines strengths of both common modeling paradigms. We also demonstrate its potential as a preprocessor for structure mining tasks via a case study in semi-supervised learning on real excavator maintenance work orders.

[DOI: 10.1115/1.4045686]

Keywords: cognitive-based design, expert systems, machine learning, systems engineering

1 Introduction

Many engineering and design tasks rely on having an accurate representation of a system's structure. This *structured knowledge*, made up of concepts and concept relations, can then be used to create more reliable models for engineering learning tasks. For example, such structures include ontologies for the industrial data and reliability analysis [1–3], design structure matrices for quantitative design of complex systems [4–6], or rule sets for normalizing reliability data, e.g., survival analysis [7–9]. Although some effort has been spent automating the process of building these "knowledge structures" [10], even these cases require significant manual effort to collate validated vocabularies and syntactical rules; in general, obtaining such structured knowledge can be challenging since closed form descriptions and characterizations of structure are often either too difficult or too time consuming to produce. Manual construction of bespoke, application-specific engineering ontologies are often cost prohibitive to create and maintain, and the use of general purpose concept networks [11–13] often lack needed domain knowledge.

In light of these difficulties, many researchers have realized a need to rapidly acquire this structured knowledge from their staff's expertise, whether through elicitation [14] or by *learning* from their data (i.e., historical records). The latter is often easier or more reliable to collect from experts when time constraints and demanding responsibilities play a significant role in data creation. This process of learning structured data from written historical records is often referred to as *structure mining*, or in the machine learning community, a special case of representation learning on discrete data (e.g., graphs) [15,16].

In technical fields like engineering, design, and manufacturing, performing structure learning faces two key difficulties, which are focused in this study. First, performance of existing structure learning approaches hinges on an appropriate definition of similarity among concepts. As we describe in Sec. 2, common choices for this similarity fall into two camps—correlation versus conditional strength. This paper compares the merits of both approaches and demonstrates conditions under which both struggle to accurately infer ground truth structure (Sec. 4). Second, available historical data are often difficult to use directly; the domain experts creating it generally assume that it will be read and adapted by colleagues or other experts in their own field. This means an analyst cannot simply use, e.g., written lab notebooks, technical reports, or maintenance work orders (MWOs), taking them at face value; words and concepts with more general meaning to the layman will have domain-specific meaning.

This paper addresses this problem by adapting models of memory recall in psychology to posit a statistical model that accounts for how experts may generate tags given prior experience or context (Sec. 3). This model also forms a middle-ground between existing similarity measurement tools and sheds light on the differences among those models.

The following sections describe our perspective on the use of structure learning while dealing with tags and historical records—i.e., using maintenance work order to infer system structure.

[1] Corresponding author.

Contributed by the Design Automation Committee of ASME for publication in the Journal of Mechanical Design. Manuscript received June 14, 2019; final manuscript received December 10, 2019; published online December 12, 2019. Assoc. Editor: James T. Allison.

We use this concrete example to highlight why structure learning is difficult, what practical issues one faces when evaluating such techniques, and then summarize the paper's key research questions.

1.1 Example of Maintenance Work Orders and Tags.

In contexts where annotation is costly, significant research has been done to empower casual annotators and to understand how natural classification and labeling schemes arise in social communities. When restricted vocabularies and categories for record annotation are not available or practical, users are often allowed to assign uncontrolled keywords to a record, a process referred to as "tagging." This allows concepts to be derived freely in the course of work, as repeated and cross-contextual usage, often among multiple users, leads to a naturally arising set of useful, domain-specific concepts [17–20].

Historical Record (MWO) Annotation Comparison
"*Hydraulic Leak at cutoff unit; Missing fitting replaced*"

Categorization:	
Subsystem	`142_HYD_SYSTEM`
Error Code	`ERR_142A`
Action Taken	`PART_ORDERED`
Tags:	
objects	cutoff_unit, hydraulic, fitting
problems/actions	leak, replace

This freedom implies that "tags" have not been directly controlled—that is, picked from a fixed list known ahead of time. They lack a designed model of individual tag *relationships*. Therefore, the crucial step required to use tags for structure mining is to determine relationship strength: mathematically modeling pairwise tag similarities (or conversely, distances). As will be discussed in Sec. 2, methods for approximating concept relationships in unstructured multisets like tags vary widely and have a variety of implications.

1.2 Evaluating Similarity Measures Between Tags.

Because so many downstream structure mining and analysis tools require some underlying assumption of what makes concepts similar, it is important to consider the impact of selecting a similarity model.

How does one evaluate whether a given similarity measure is "good" for a given problem? To unpack common evaluation measures, assume we can represent our system structure as a weighted graph $G = \{V, E\}$, where the node set V represents concepts in our system (assumed to be known, one for each tag), and the edges E are weighted based on similarity between these. Our modeling assumptions can influence two key properties of E that are crucial to a successfully recovered structure:

Precision and Recall: Detected relationships should be distinctly recognizable, and those detections should be reliably useful. This implies graph sparsity and ensures that indirect similarity through an intermediary concept is not conflated with true similarity. Any nonzero edge weight should therefore correspond to a real concept connection of some kind. In other words, *of the detected edges, most should be relevant*, and *of relevant edges, most should be detected.*

Robustness: Since structure is not known a priori, some amount of filtering on edge weights will take place to enforce the previous properties. The quality of a recovered structure should be robust to changes in filter strictness. This implies that *relevant edges should not be quickly lost as an increased edge-weight thresholds remove unwanted detections.*

1.3 Research Questions.

This paper investigates the performance of two common similarity measures with respect to these traits: co-occurrence frequency-based (typified by cosine similarity) and conditional sequence probabilities (typified by the kth-order Markov chains). We then show that, while each has strength in one of the above desideratum, the other can be lacking. This necessitates a hybridized approach to "interpolate" between two measures. To accomplish this, we frame the act of tagging historical records, typified by MWOs, as a type of semantic memory recall from within the expert's internal "knowledge graph"—this leverages the concept of *semantic fluency*, which we define in Sec. 3. Specifically, we investigate the following:

R1 Whether incorporating mechanisms for non-Markovian jumps improve the precision and recall of structure recovery compared with frequency-based or Markovian relationship measures

R2 Whether the relationship graph learned through this model shows improved accuracy of learning tasks that require an assumed similarity measure, compared with the traditional measures.

We empirically test (1) precision and recall for learned similarity measures from multiple synthetically generated tag data sets (using several known structures) via a memory recall model; and (2) semi-supervised concept classification using tagged maintenance work orders from a mining excavator operation, not having a previously known concept relationship structure.

In both cases, we show that by building a probabilistic model that accounts for (and subsequently learns) how experts structure their implicit knowledge of a domain, one can achieve significantly better performance (as measured by precision and recall) than the existing methods of relationship recovery.

2 Related Work

Using data to infer the underlying structure of a complex system is a long-standing goal within domains that depend on accurate network recovery, such as biological systems and disease transmission vector modeling [21,22]; uncovering economic interactions and social networks [23,24]; inferring physical models by learning governing equations [25]; or even description generation in computer vision, and quantifying how humans reason about belonging and causality in ambiguous images or contexts [12,13]. It is beyond the scope of this work to exhaustively compare state-of-the-art in representation learning[2]; still, a common theme found among these techniques is an assumed definition for the "distance" between the observed data. For numerical data, a common assumption is that distance between observations with N features is an L-norm between the N-dimensional vectors (e.g., Euclidean distance being the L_2-norm), although often a more robust characterization of distances exists on a lower dimensional manifold embedding within that space [26].

Learning useful structures from nonnumerical data, like tags or networks, is a rapidly progressing research area. Recent works include extraction of latent taxonomies from tagged documents [27], extracting interconnected term and topic hierarchies through nested stochastic block models [28], or use of hyperbolic embeddings to recover latent hierarchy in similarity data [29,30]. Once again, all of these tools assume an a priori estimate of what being "related" means: how similarity and distance are defined in the latent feature space. Therefore, to make the best use of these burgeoning tools, it is paramount to characterize the impacts of one's chosen similarity measure and to ensure that the choice matches well with properties of the data and subsequent models being used.

2.1 Global Frequency and Context.

A common way to encode similarity between observations with discrete-valued features (whether tags, graphs, or natural language documents) starts with making the intuitive assumption that features occurring

[2] Readers are directed to the literature review by Bengio et al. [16].

across similar contexts *are similar*. This style of similarity measure naturally arises when using frequency-based mathematical representations of text via natural language processing (NLP). These include "bag-of-words" weightings [31], topic models [32,33], or semantic vector embedding [34,35]. In these vector representations of an observation, then, the similarity between two observations is less about how "close together" the co-occurrence frequency magnitudes are and more about occurrence frequency correlations—the vector *direction* similarity. This is encoded in the cosine similarity measure, i.e., the cosine of the angle between the vectors.

Rather than a corpus of documents, we are concerned specifically with the set of tags assigned to records. This set of tags, especially when created by multiple users, is commonly referred to as a *folksonomy*, a portmanteau of "folk" and "taxonomy" [36]. Because folksonomies generally ask users to determine minimal representative labels rather than strict classifications (i.e.,tags), each label can be seen in multiple contexts, much like words in text. The predominant way to analyze tag similarity, then, is by their co-occurrences with each other [37,38]. If, over a set of C records, tag t_k has binary vector $u_k = \{\mathbf{1}_c(t_k) : c \in C\}$, then the cosine similarity s between the binary occurrence vectors of the tags t_1, t_2 is defined as follows:

$$s(t_1, t_2) = \frac{u_1 \cdot u_2}{\|u_1\| \|u_2\|} \tag{1}$$

This measure is applied across many NLP and folskonometric methods to structuring relationships between tagged concepts in useful ways, including the taxonomy extraction and hyperbolic embedding work mentioned earlier [27,30,39]. For this work, while significant advances have been made in contextual, set-based measures on, e.g., topic models or semantic embeddings, the latent relations being "learned" are quite often difficult to interpret for humans [40], stemming from the so-called black box nature of these models. Therefore, we make use of Eq. (1) for ease of interpretation and broad acceptance.

The power of cosine similarity comes from its computational simplicity and an ability to deal with high-dimensional feature sets (e.g.,the set of all unique tags in a folksonomy). These context-based similarity measures (which also include Jaccard similarity, mutual information, and the like) base their approach on treating tags as unordered sets. This has the distinct advantage of picking up on unobvious relationships between tags that co-occur in wildly varying contexts, quickly recovering global scale structures with minimal observations. We should expect that most relevant relationships are quickly retrieved this way, i.e., cosine similarity typically exhibits a high *recall* score in structure recovery.

However, one can imagine adding a tag to a document that is related to, say, the previously added tag, but not necessarily to the first tag added; so when is co-occurrence a coincidence? This line of reasoning implies a separate model, where annotating each tag implies a probability to use or not use some subsequent tags.

2.2 Local Sequence Probability.

On the opposite side of treating tags or text as an unordered set, one might think of tagging as a sequential stream of tag additions. Once again taking a cue from NLP, one might assume that each subsequent concept written in text is directly conditional on what was written previously. Predicting the probability of observing a word based on the previous, n locally observed words (in order) is known as an nth-order language model [41].

For tags, say assigning a tag to a document is equivalent to being in that tag's "state," and the relations between states is the probability of transitioning between those states. Assigning tags would then be a process satisfying the Markov property; thus, for an nth-order tag Markov model, the probability of observing any ith tag in a sequence is $P(t_i|t_{i-1}, \ldots, t_{i-n})$. In practice, given a data set of observed tag sequences, this means finding the maximum likelihood estimate for transition probabilities between tags in the form of conditional probability tables.

This is a powerful (although oversimplifying) model, and many techniques seek to apply a similar reliance on the sequential nature of textual language or tagging to predict subsequent relevant tags. Hidden Markov models, for instance, treat each state as a *distribution* of tag "emmission" probabilities and train to find transitions between these distributions. These are often used for both tag recommendation and predicting other system feature relationships with tags or keywords [42]. Other success has been found using recurrent neural networks as language models, which are capable of storing sophisticated, long-distance contextual information while predicting a sequence [43].

Because the intuition behind these sequence-based models comes from nearby tags having a strong influence on each other, one way to quickly estimate the relationship strength of two tags is to estimate the probability of observing them in sequence:

$$s(t_1, t_2) = \max[P(t_1|t_2), P(t_2|t_1)] \tag{2}$$

This preserves symmetry in the similarity measure, allowing us to compare it with the cosine similarity above. Since our similarity is calculated from a sequence, and the model is estimated only from observed sequences, we expect a high fraction of total predicted relationships to be truly relevant, i.e., precision score should be high.

Still, what if tag relationships exist that are rarely observed directly, due to an third, highly common tag? What if there are biases in tag ordering due to quirks of user reporting? Rather than having to choose between skewing toward recall or precision, is there a model that more naturally fits the mechanics of tagging to avoid systematic failure to improve either metric?

3 Modeling Tags as Memory Recall

As discussed earlier, common techniques for discovering structural relationships in the tagged data primarily rely on either frequency and co-occurrence information or conditional sequence probabilities of discrete objects/concepts. These are powerful and easy-to-apply models used ubiquitously for speech or the written word and can also lead to systematic misbehavior under the conditions that user taging presents.

Instead, this paper tries to address shortcomings in relationship recovery by explicitly emulating the dynamics of how humans might recall concepts from memory and apply this memory recall to estimate tag relationship structures.

This section first describes the concept of *semantic fluency tests* —an existing tool in psychology literature for testing concept relationship recall—and how the surrounding theory relates to tagging engineering records. We then describe a computational method to implement the concept of semantic fluency using initial-visit emitting random walks (INVITE) [44]—a non-Markovian probabilistic model for sampling semantic fluency type data from an underlying concept relationship network.

3.1 Semantic Fluency.

When a user begins to tag a record, they try to search their memory for concepts that are relevant to the record itself, in the context of the engineered system it pertains to. In the interest of recovering latent relationships between system components as understood by, e.g., a technician, we restrict our discussion on tags to ones representing objects/items directly (although they may additionally concern problems that were encountered with some items or how other items were used to solve these problems [20]).

The exact psychological mechanisms by which a person searches through their memory is still an active area of research and has been modeled in various ways. Some recent studies [45] propose that concepts are recalled sequentially by foraging in "semantic patches"—in brief, that humans sequentially recall concepts that are "near" each other in some person-specific semantic space built through experience.

Specifically, these patches are thought of as existing in a high-dimensional concept-space,[3] and the likelihood that some concept is recalled next is based on combining both associative and categorical knowledge into a similarity measure between the current recalled concept and the next. By thresholding this high-dimensional association "map," binarizing it as "is related"/"is not related," we can represent this map as a graph,[4] where concepts are nodes and an edge represents "is related." Memory recall, then, consists of a sort of "walk" along this graph.

A classic psychological experiment to measure what such a graph might look like is the semantic (or, verbal) fluency test. Given an object type (e.g., animal):

(1) Recall and record an object of that type.
(2) Record the next object of this type you think of.
(3) Continue recording for the remaining time.

The reader is encouraged to try this process out for themselves. One advantage of this test lies in not restricting (or having to specify a priori) the relationship between objects required to record subsequent ones. For example,

$$\text{dog} \rightarrow \text{cat} \rightarrow \text{lion} \rightarrow \text{tiger} \rightarrow \text{elephant} \rightarrow \text{wolf} \cdots$$

As in this example, it is common for animal-based semantic fluency lists to start with household pets, potentially switching to unrelated categories like "large cats," for further exploration, before either retracing back to a previous category (e.g., canines to "wolf" via "dog") or onward via new similarities (e.g., African animals to "elephant" via "lion").

Altering the scope of such a task to "system object that is relevant to a given record" instead of "object that is an animal" represents a task that is remarkably similar to how the user tagging task was construed in Sec. 1. In this model, each subsequent tag assigned to a record constitutes a"jump" in the user's internal "tag network," which depends in some way on previous tag jumps for that record.

Thus, any attempt to recover the associative strength between concepts should necessarily incorporate these context "jumps" (canines, big cats, household pets, African animals) in a way that allows for "retracing your steps" to previous concepts when exploring some new context. One model that incorporates these precise features mathematically is the recently proposed INVITE model [44].

3.2 Initial-Visit Emitting Random Walks.

The described semantic fluency model for tagging boils down to two key components of a user's cognitive task when recalling relevant tags:

- They submit tags sequentially, as they recall **unique** defining concepts related to the record.
- They recall each concept by traversing relationship links between **it** and **any** recently recalled concepts.

Figure 1 illustrates such traversals by using a drivetrain component network from the study by Walsh et al. to stand in for a user's latent understanding of a system's structure. In that figure, each "MWO" begins with a some initially sampled tag, with subsequent tags potentially stemming from a "jump" to distant (non-adjacent) nodes in the network. This illustrates hidden jumps due to initial-visit censoring. The resulting tags could be still reasonable for a MWO where those components were involved: Example #1 could represent the text "Had to replace bearing retainer; bearing balls showed excess wear. Inner and outer bearing races cleaned." Despite not being directly connected, they share a common region in the graph, with each subsequent tag accessible in memory from *one of the previous tags*.

This differs from a standard bag-of-words model—where all tags are assumed to be linked through co-occurrence on a record (i.e., only global graph topology matters), and from nth-order Markov models—where tag relations are limited to the nearest (or, previous) n entities (i.e., only local sequences of observed tags matter). In addition, in neither of these models are tags explicitly modeled as unique within the record.

This nicely illustrates the trade-off between categorical and associative memory foraging that [45] discusses at length and is precisely the feature of tagging we investigate when extracting a more realistic representation of tag relationships through the mathematical framework of initial-visit emitting random walks.

Say the set of components or concepts that have a corresponding tag in our system is denoted by the node set N. A user-given set of T[5] for a specific record can be denoted as a random walk trajectory $\mathbf{t} = \{t_1, t_2, t_3, \ldots t_T\}$, where $T \leq N$. This limit on the size of T assumes that tags are a set of unique entries: any transitions between previously visited tags in $\mathbf{t}$ will not be directly observed, making the transitions observed in $\mathbf{t}$ strictly non-Markovian and allowing for a *potentially infinite* number of possible paths to arrive at the next tag *through previously visited ones*.

Instead of directly computing over this intractable model for generating $\mathbf{t}$, the key insight from the original INVITE paper [44] comes from partitioning $\mathbf{t}$ into $T-1$ Markov chains with absorbing states, where previously visited tags are "transient" states, and unseen tags are "absorbing." It is then possible to calculate the absorption probability into the kth transition ($t_k \rightarrow t_{k+1}$) using the *fundamental matrix* of each partition. If the partitions at this jump consist of q transient states with transition matrix among themselves $\mathbf{Q}^{(k)}_{q\times q}$, and r absorbing states with transitions into them from q as $\mathbf{R}^{(k)}_{q\times r}$, the Markov transition matrix $\mathbf{M}^{(k)}_{n\times n}$ has the form

$$\mathbf{M}^{(k)} = \begin{pmatrix} \mathbf{Q}^{(k)} & \mathbf{R}^{(k)} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \tag{3}$$

where $\mathbf{0}$ and $\mathbf{I}$ represent lack of transition between/from absorbing states. It follows from Ref. [48] that the probability P of a chain starting at t_k being absorbed into state $k+1$, letting $\mathbf{N} = (\mathbf{I} - \mathbf{Q})^{-1}$, is given as follows:

$$P(t_{k+1}|t_{1:k}, \mathbf{M}) = \mathbf{N}^{(k)} R^{(k)}\Big|_{q,1} \tag{4}$$

The probability of being absorbed at $k+1$ conditioned on jumps $1:k$ is thus equivalent to the probability of observing the $k+1$ INVITE tag. If we approximate an a priori distribution of tag probabilities to initialize our chain as $t_1 \sim \text{Cat}(n, \theta)$ (which could be empirically derived or simulated), then the likelihood of our observed tag chain $\mathbf{t}$, given a transition matrix, is as follows:

$$\mathcal{L}(\mathbf{t}|\theta; \mathbf{M}) = \theta(t_1) \prod_{k=1}^{T-1} P(t_{k+1}|t_{1:k}; \mathbf{M}) \tag{5}$$

Finally, if we observe a folksonomy of tag lists $\mathbf{C} = \{\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_c\}$ and assume θ can be estimated independently of $\mathbf{M}$, then we can frame the problem of structure mining on observed INVITE data as a minimization of negative log-likelihood of our folksonomy given $\mathbf{M}$:

$$\mathbf{M}^* \leftarrow \underset{\mathbf{M}}{\text{arm}\min} \quad \sum_{i=1}^{C} \sum_{k=1}^{T_i-1} -\log \mathcal{L}\left(t^{(i)}_{k+1}|t^{(i)}_{1:k}, \mathbf{M}\right) \tag{6}$$

3.3 Implementation.

As formulated in Eq. (6), the optimization is constrained: in addition to requiring row-stochasticity, the

[3] Although less applicable in technical or domain-specific corpuses where examples are too few and far between, this is the intuition that leads to the success of vector-based semantic embeddings like `gloVe` or `word2vec` [34,35].

[4] Also called an *associative network* [46].

[5] While some sources use "tagging" as a proxy for a set of strictly unordered labels (as in multilabel classification), we preserve the mechanism by which the tags were generated in the first place, i.e., in a *specific* order.

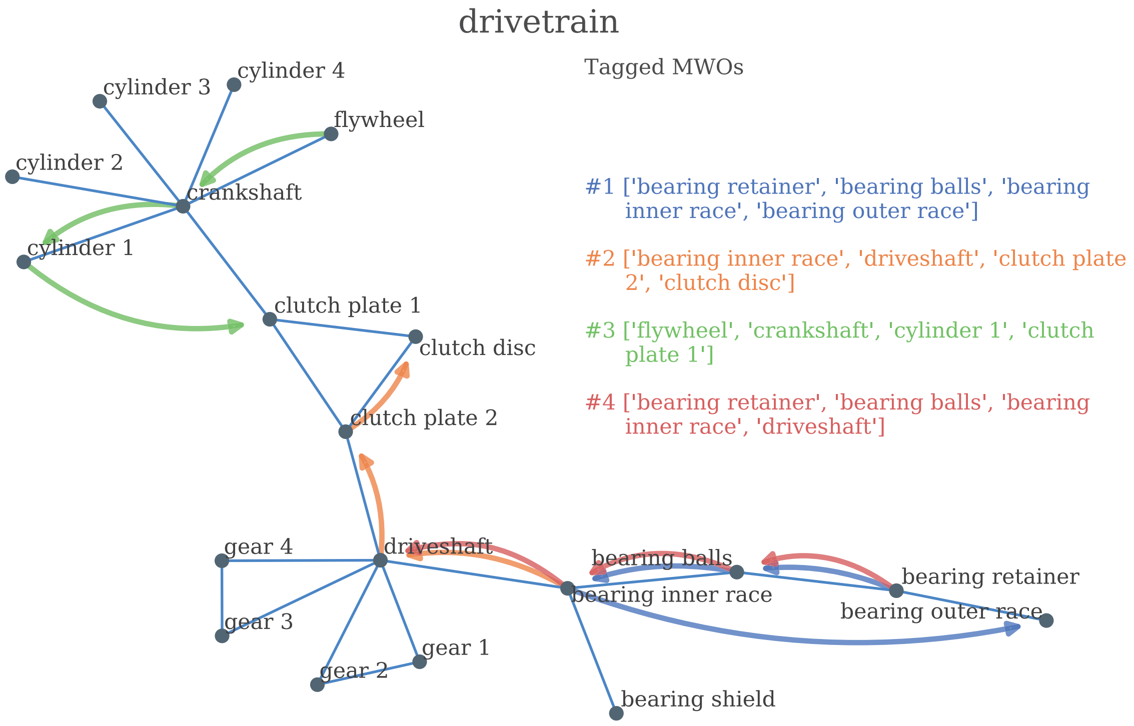
drivetrain
Tagged MWOs
#1 ['bearing retainer', 'bearing balls', 'bearing inner race', 'bearing outer race']
#2 ['bearing inner race', 'driveshaft', 'clutch plate 2', 'clutch disc']
#3 ['flywheel', 'crankshaft', 'cylinder 1', 'clutch plate 1']
#4 ['bearing retainer', 'bearing balls', 'bearing inner race', 'driveshaft']
cylinder 3
cylinder 4
flywheel
cylinder 2
crankshaft
cylinder 1
clutch plate 1
clutch disc
clutch plate 2
gear 4
driveshaft
bearing balls
bearing retainer
bearing inner race
bearing outer race
gear 3
gear 1
gear 2
bearing shield

Fig. 1 Example observations of INVITE samples on a drivetrain network model from the studies by Haley and Walsh et al. [47,50]

matrix N is only guaranteed to exist if self-transitions are disallowed, as proved in Ref. [44]. Similar to that implementation, we introduce a softmax reparameterization of $\mathbf{M}$ that allows the optimization to be unconstrained in $\mathbb{R}^{n\times n}$, and guaranteeing row stochasticity.

$$M_{i,j} \leftarrow \frac{\exp\left(M_{i,j}\right)}{\left[\sum_j \exp\left(\mathbf{M}_i\right)\right]_j}$$

We introduce a modification to this reparameterization. Equation (6) implies that $\mathbf{M}$ represents a *directed* graph. Although we model each tag as being generated conditional on preceding tags alone, we wish to preserve the intuition that relationships between tags are still assumed to be bidirectional, while not strictly enforcing $\mathbf{M}$ to be symmetric (undirected) while learning from samples, as in Ref. [49]. Put simply, one-directional relationships can be useful to model when they are largely the case (e.g., cat → lion), but we may not wish to encourage one-directional relations that are quirks of imbalanced data and how people talk (gear_1 ↔ gear_2). To speed up the recovery of what we assume is a "symmetry-dominant" $\mathbf{M}$, we can bias the optimization toward symmetry via an update to each entry prior to the softmax step:

$$M_{i,j} \leftarrow \max\left\{M_{i,j}, M_{j,i}\right\} \tag{7}$$

In folksonomies where the recovered weights in each direction are known to be meaningful, this can be skipped.

4 Experiments

Per the above discussion, the following experiments and case studies are done by comparing the recovered similarity measures, in the form of tag relationship graphs, between a cosine similarity measure, first- and second-order Markov chain models, and the proposed INVITE-based similarity model.

To address R1 from Sec. 1.3, the first experiment demonstrates the effectiveness of incorporating mechanisms from the INVITE model when the tag-style data are generated in the manner of semantic fluency tests. We synthesize tagged records as censored random walks on a sample of random small-world networks, as well as on networks representing real engineering systems, as described in Ref. [50].

We use these synthetic tags to (1) measure the network recovery accuracy of the various similarity measure models using standard information retrieval metrics, (2) determine the ability of INVITE-based similarity to hybridize precision and recall efficiency of other models, and (3) illustrate qualitatively the key failure modes of various modeling assumptions when INVITE mechanics are not taken into account.

In the second experiment, addressing R2, we determine the performance of the similarity measures as preprocessing steps to accomplish a semi-supervised tag classification task. We utilize a folksonomy of real, tagged excavator MWOs, for which a "true" underlying system structure is not known a priori. Classification scores and divergence from true multinomial tag classification distributions are presented.

For all experiments, we address the way in which different models perform under similarity *thresholding*. Thresholding is important since, as it is universally the case in representation learning, we do not generally have a "ground truth" representation to tune parameters against. As described briefly in Sec. 1.2, it is the performance characteristics over a *range* of thresholds that we seek to improve. After normalizing the relationship strength of any given edge into the range $M^*_{i,j} \in [0, 1]$, we threshold $\mathbf{M}$ such that, for a given threshold value $\sigma \in [0, 1]$, the entries of a thresholded similarity matrix $\mathbf{M}^\sigma$ are given by

$$M^\sigma_{i,j} = \begin{cases} 1, & \text{if } M^*_{i,j} \geq \sigma \\ 0, & \text{otherwise} \end{cases}$$

These networks should be sparse, to be informative about the existence of important relationships while ignoring noisy ones. This implies class-imbalance between edges and nonedges as target predictions. For imbalanced learning problems like this, precision (P, the ratio of true-positive edges to total detected edges) and recall (R, the ratio of true-positive edges to total true edges) at each threshold can elucidate model robustness under varying threshold

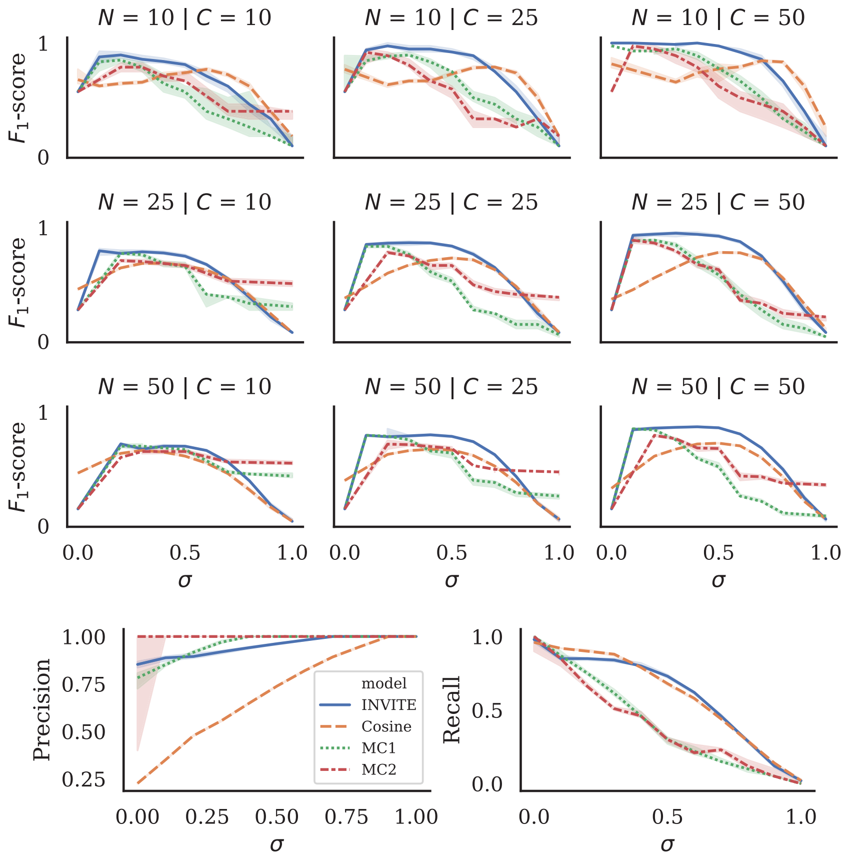

Fig. 2 Top: F_1 scores for various combinations of network size N and number of "tagged records"/random walks C, shown as a function of similarity threshold σ. Median over ten trials for each setting, shown with 95% confidence interval (1000 bootstrap samples). Bottom: Precision and recall across all 90 trials, for all nine setting combinations.

sensitivities[6] [51]. Combining both into a single metric for balancing these two desirable traits is primarily done using an F_β *measure*:

$$F_\beta = (1+\beta^2)\frac{PR}{\beta^2 P + R} \tag{8}$$

In this paper, we use the most common case of $\beta = 1$ to equally balance the importance of precision and recall.

Because of the alterations described in Sec. 3.3, the analytic gradient for the INVITE loss function described in Ref. [44] no longer applies; instead, we make use of automatic differentiation as a means to ensure accurate gradient calculations under these modifications. The package PYTORCH [52] was used for optimization with automatic differentiation, in the PYTHON programming language. For calculating maximum likelihood estimates for the Markov chain models, we have made use of the ,PYTHON package POMEGRANATE [53]. Code will be made available in an associated repository for reproducibility.[7]

4.1 Experiment 1: Recovering Known Networks.

To validate the ability of our method to accurately reconstruct engineering networks compared with other methods, we first synthesize censored tag lists from true tag relationship networks under a variety of conditions.

Randomized Graphs: Random graphs were generated, consisting of Watts–Strogatz randomized connections between $N \in \{10, 25, 30\}$ nodes. For the purposes of comparison across networks, the mean degree was set as $K_{WS} = 4$ with the re-wiring coefficient set[8] to $\beta_{WS} = 0.166$ [54]. Then, synthetic folksonomies were generated consisting of $\|C\| \in \{10, 25, 30\}$ "tagged documents" (i.e., censored random walks on a given graph). In this experiment, each document/random walk was assigned $\|T\| = 4$ tags. The median F_1 score across the for ten different graphs are shown for each N/C combination in Fig. 2. The precision and recall curves are also shown, collapsed over all 90 random graphs.

Discussion: As measured by F_1 score, the INVITE-based similarity measure consistently outperforms both the Markov chain and cosine similarity measures across a wide range of thresholds for all graph/random walk settings. More interesting, and more useful for practitioners in an unsupervised setting, is the *shape* of these curves and how they change. For low-complexity networks, cosine similarity is relatively stable over all thresholds. Then, as complexity increases, much more filtering has to take place (higher σ) before it reaches best performance. This is contrary to the sequence-based Markov chains, which show dramatically better performance at thresholds *barely* above 0, but suffering at higher specificity.

Meanwhile, the INVITE-aware similarity shows a sharp increase at low σ, like the Markov model, while retaining the smoothness of the cosine model as σ is tuned higher. This tendency to capture strengths of each is more clear if precision and recall are shown separately, as in the bottom of Fig. 2, where the precision behavior

[6] Recall is alternatively known as *sensitivity*, while precision is alternatively known as positive predictive value.

[7] https://github.com/tbsexton/organizing-tags

[8] This Watts–Strogatz setting, while not necessary for the purposes of our experiment, can give networks with experimentally similar properties to real cognitive associative networks; see Ref. [49].

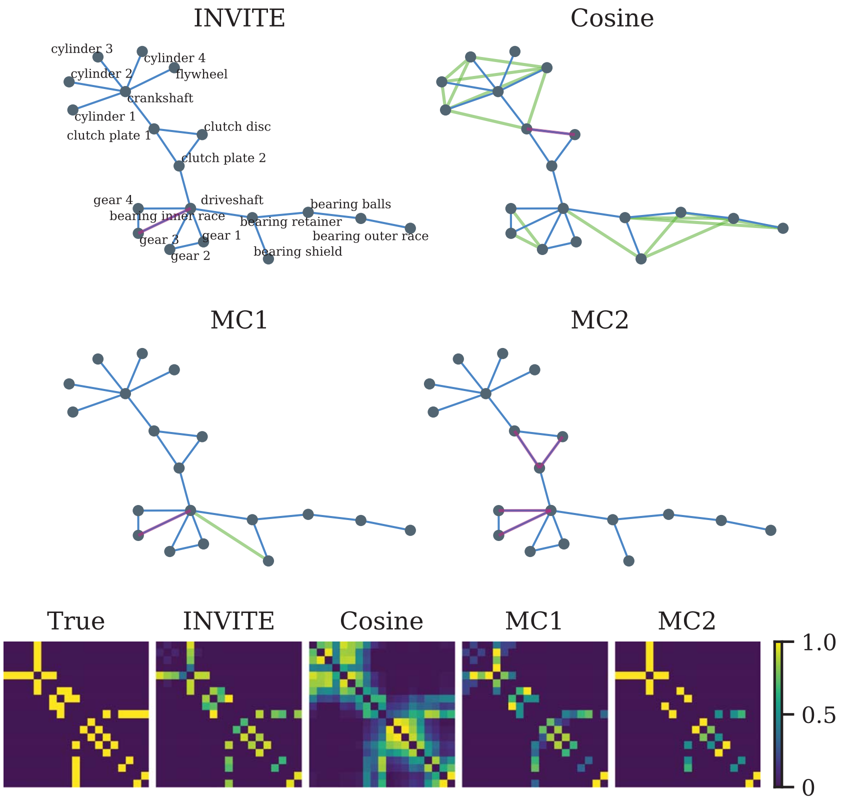

Fig. 3 F_1-optimal thresholded networks, with unthresholded adjacency-matrix representations of M^*

of the INVITE model matches that of the Markov similarity (its presumed strength; recall Sec. 1.2). At the same time, its recall behavior more closely resembles that of the cosine similarity model (again, the strong suit of that paradigm).

This dynamic, the trade-off between models that favor recall versus precision, can be made clearer with a concrete example.

Real System Networks: To qualitatively understand the underlying failure/success modes of each measure, we turn to the real system networks presented in Refs. [47,50]. We start with their drivetrain model ($N=18$), which is simple enough for visualization while demonstrating common patterns in engineered systems. We sample $C=20$ random walks of length $l=4$, some of which were used in Fig. 1. These settings were chosen intentionally as low performing to illustrate failure modes in each model type in Fig. 3, where optimal F_1 thresholded networks are shown with false positive (green) and false negative (red) edge predictions, along with the unthresholded similarity matrix $\mathbf{M}^*$.

The cosine similarity model has quickly detected *all* relevant edges, as seen in the un-thresholded matrix, but it has also overestimated the connectivity of local communities. Engineered systems often display hierarchical connectivity patterns, where many low-level parts are only similar indirectly because of their connection to a key higher level component (e.g., all cylinders and the flywheel to the crankshaft, or all gears and bearing inner race to the driveshaft). Because the observed transitions are censored from seeing previously visited nodes—just like a user only tags each concept once, even while they continue to use it to recall other concepts—the cosine model sacrifices higher component connections to preserve the perceived frequency with which low-level tags co-occur.

The Markov models, on the other hand, demonstrate remarkably few false positives. Instead, accuracy is limited by the number of available observations for each sequence of two or three tags. Since the number of possible paths is so large, true relationships might only be realized as a direct sequence a single time, or not at all, when so little data are available. This means true edges are quickly lost when the model's certainty about their existence is no better than false edges detected in a censored INVITE jump. The INVITE model balances aspects from both models, by quickly gaining certainty about the overall structure, while still allowing for exploration to re-route potential connections through edges that make more sense *sequentially*.

For the interested reader, the same exercise was performed on a reduced version of the Airplane network presented by Walsh et al. [47,50]. Due to complexity of the visualization, nodes with identical names (barring a numerical identifier) were merged into a single concept tag. Figures made available in the Supplemental Material on the ASME Digital Collection show F_1 scores, precision-recall curves, and average precision scores for this more complex network. Readers will note that once again certain desirable behaviors of the Cosine model are exhibited by INVITE (e.g., a smooth rise in F_1 over a wide range of middling thresholds, with maximum at a mid-to-high value), along with desirable traits of the Markov model (e.g., significant F_1 at near-zero thresholds).

4.2 Experiment 2: Real-World Excavator Maintenance Work Orders. Unlike the previous synthetic experiments, one does not in general have access to a ground truth network that validates any chosen similarity measure. Not having labeled data or targets to supervise the learning process is one of the key difficulties in representation learning [16]. To assess the applicability of the INVITE-based similarity measure to real-world scenarios, we apply our model to tags annotated for a mining dataset pertaining to eight similar sized excavators at various sites across Australia [7,55].

The tags were created by a subject-matter expert spending 1 h of time in the annotation assistance tool NESTOR [56] using a methodology outlined in a previous benchmarking study for that annotation method [9].

That work compared the ability of tags to estimate survival curves and mean time-to-failure, when compared with a custom-

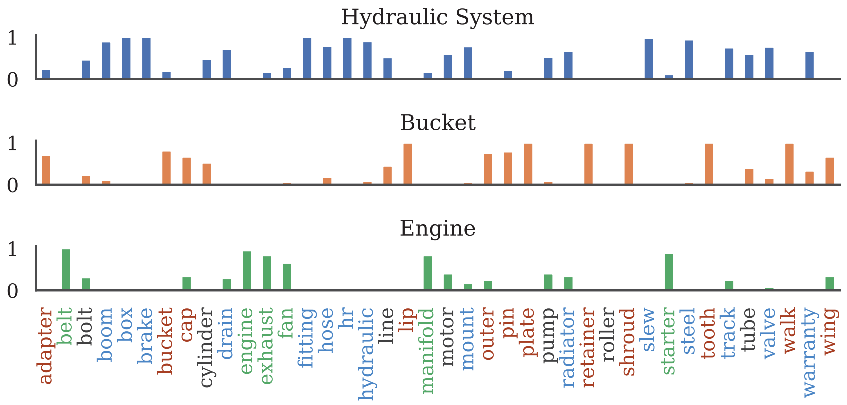

Fig. 4 Ground truth tag multinomial distributions across the top three subsystems. "Unknown subsystem" tags are shown in black.

designed keyword extraction tool based on classifying the maintenance issues by subsystem. While certain sets of tags were able to predict time to failure with high accuracy for certain subsystems, a key problem identified in that work is in knowing beforehand "which tags best represent a given subsystem?"

Some tags are sufficient-but-unnecessary conditions to represent a subsystem—e.g., the "hydraulic" tag indicates a Hydraulic System MWO, but so might a "valve," such that hydraulic is implied but not present. Consequently, we can treat the problem of assigning tags to a subsystem as a semi-supervised multi-class classification problem: given a few known tag → subsystem assignments, and a similarity value between all pairs of tags, *classify each unassigned tag as belonging to a subsystem.*

To test the ability of the similarity measures to accomplish this, the top three most common subsystems in the data were used as classes, namely, hydraulic system, engine, and bucket. The tags "hydraulic," "engine," and "bucket" were assigned to those subsystems as known labels, respectively. Tags were filtered to only include ones of high importance and sufficient information: only work orders containing at least three unique tags, and only tags that occurred at least ten unique times within the those work orders, were included for this analysis ($C = 263$ MWOs, $N = 40$ tags). Then, the number of occurrences for every tag can be compared across subsystems, giving each tag a ground truth multinomial (categorical) probability distribution for occurring within each subsystem, as shown in Fig. 4. We determine ground truth classification labels as subsystems that account for ≥60% of each tag's occurrences. Tags more balanced than that are considered "unknown subsystem."

Implementation: We proceed in a similar way as before in training the similarity measures for each tag. Note that the tagging annotation process used by Ref. [56] assigns tags when they are recognized in raw text through one of many alias'. Therefore, the ordering of tags for these MWOs is strictly based on the order in which English is written—this makes the order any pair's occurrence quite meaningful. As discussed in Sec. 3.3, we skip the symmetrization step of Eq. (7) until after training is complete.

To perform semi-supervised classification on the recovered relationship graphs, we use a label-spreading algorithm described in Ref. [57], which itself was inspired by spreading activation networks in experimental psychology [58,59]. The result of this algorithm is tags having a score for each class, with the classification being the maximally scored class for that tag. These class assignments can then be compared with the ground truth labels, which we have done by weighted macro-averaging of the F_1 score (see the top of Fig. 5).

5 Discussion

The classification of the INVITE-based similarity measure far outperforms the other measures as a preprocessor for label spreading, when measured by average F_1 score. However, since these "classifications" are actually thresholded multinomial distributions (with some tags regularly occurring across multiple subsystems), how do we know if an underlying structure has actually been recovered, rather than simply a black box classifier that happens to perform well at this setting?

To begin answering this question, we might ask whether the relative scores returned by label spreading are similar to the original multinomial distributions themselves, rather than the overall classification. To find out, we use softmax normalization[9] to transform each tag's scores into a "predicted multinomial," before finally calculating the Kullback–Leibler divergence (KLD) between the true and predicted multinomials for every tag. The total KLD, summed over all tags, is also shown in Fig. 5, along with positions of each tag's multinomial as projected onto the two-simplex for the true and F_1-optimal predicted distributions. Once again, the INVITE performs much better at this task over a wide range of σ (lower is better).

A reason for the performance disparity can be seen in the simplex projections: recovered topology via INVITE similarity does a much better job of separating the three classes, while not letting any single tag overcompensate by dominating a subsystem's area. Even the "unknown" tags are correctly placed roughly between Bucket and Hydraulic System regions, reflecting the true topology of the system. Interested readers are encouraged to find the best-performing recovered networks visualized in Supplemental Figure 9, further demonstrating how the properties of each similarity measure behave radically differently.

One other point of note is the number of tags-per-MWO: these results were calculated using MWOs with at least three tags each, but the majority of documents in this dataset had fewer than this. The same similarity measures were calculated using more data (having at least two tags each), and performance decreased *across the board*. INVITE-based similarity still performed best, with Cosine similarity now closer to it. This decrease indicates a base level of noise in common, catch-all tags that actually reduces the amount we learn about structure from data. In a sense, quality may beat quantity for some types of representation learning. Interested readers can find all additional results in Supplemental Figure 9.

6 Conclusions and Future Work

This paper presented a method to recover a structured representation of engineering knowledge from unstructured written documents (specifically, manufacturing work orders), based on

[9] For visualization, a temperature parameter was added to softmax, and this was optimized for minimum KLD via Brent's method [60] for each similarity measure independently to provide an equal footing for comparison.

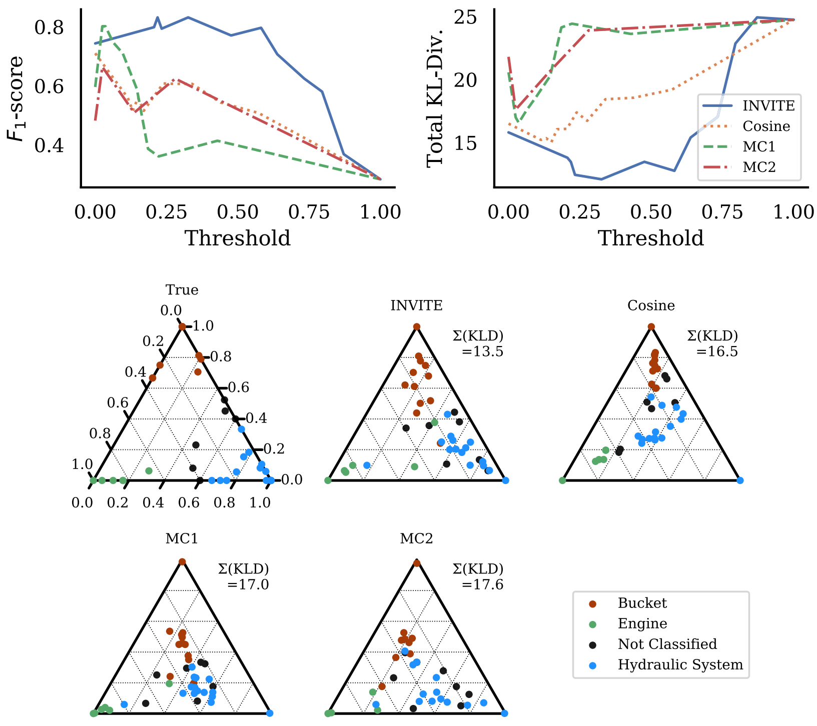

Fig. 5 Top left: (weighted) macro-averaged F_1 scores for semi-supervised classification (label spreading) for using each similarity measure as a preprocessor. Top right: total Kullback–Leibler divergence for each method between ground truth multinomial and class score distribution after softmax normalization. Bottom: tag multinomials projected onto a two-simplex, colored by ground truth classification.

INVITE. Compared with previous methods, our technique preserves local connectivity structures, even in locally hierarchical communities. This can lead to better preprocessing for downstream structure mining and representation learning tasks, as well as for analytics or predictions that better map to expert users' intuitions about how concepts within a system are organized. Both of these have the opportunity to increase trust in data-driven decision support systems, which are increasingly adopted and used without necessarily considering how humans will interact with them [61].

Plenty of work remains to be done to achieve these goals. While the INVITE-based similarity measure performed quite well in our tests, there are still discrepancies between the model it adheres to, and what one might observe in a real folksonomy. For instance, if a "hydraulics" tag is considered too general or abstract for a team that concerns itself largely with hydraulic work, this tag may be skipped as being implied through context. INVITE requires tags to be observed at least once in a record to be reached, but a better method might account for hidden paths or extra, unseen nodes that greatly improve the model's likelihood, much like a form of the "Steiner-tree" problem [62].

In addition, such a similarity measure could be used for knowledge-structuring-assistance more generally, e.g., in an active learning context. Such a tool could additionally benefit from a recent explosion in interest for preserving hierarchical and knowledge graph relationships in vector space, e.g.,via Poincaré and "Box-lattice" embeddings [29,63]. Care must be taken to allow flexible annotation of *different kinds* of relationship strengths,[10] while INVITE assumes a single, generic "similarity." Such a system should allow for multiple (potentially disagreeing) annotators, occasionally suggesting detected relationship types for review to become accepted as ground truth. We envision a type of "topic model" over the space of knowledge graphs [28], or relationship graphs a combination of independent "graph components" that maximally explain the distribution of edge types in a community [64].

Overall, the model we describe here can enable experts and novices alike to benefit from tacit expertise contained within frequently unused mountains of tagged technical records, by quickly prototyping quantitative representations of this knowledge as concept relationship graphs for downstream usage in analysis pipelines. We believe that by explicitly incorporating cognitive theories into our modeling assumptions about how users might represent and then recall their knowledge while tagging, we can accelerate the training and the use of unsupervised data-driven expert systems in engineering design.

Acknowledgment

Thanks to Dr. Melinda Hodkiewicz (University of Western Australia) for providing and annotating the excavator data, for providing feedback on the "Nestor" tagging process, and for many enjoyable and enlightening discussions on this topic.

Disclaimer

The use of any products described in this paper does not imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that products are necessarily the best available for the purpose.

References

[1] ISO/TS 15926-8:2011, 2011, "Industrial Automation Systems and Integration—Integration of Life-Cycle Data for Process Plants Including Oil and Gas Production Facilities—Part 8: Implementation Methods for the Integration of Distributed Systems: Web Ontology Language (OWL) Implementation," Standard, International Organization for Standardization, Geneva, CH.

[2] Batres, R., West, M., Leal, D., Price, D., Masaki, K., Shimada, Y., Fuchino, T., and Naka, Y., 2007, "An Upper Ontology Based on ISO 15926," Comput. Chem. Eng., **31**(5–6), pp. 519–534.

[10] For example, Walsh et al. actually construct three types of structured system representations in their paper: functional, parametric, and component (which we use here).

[3] Klüwer, J. W., Skjæveland, M. G., and Valen-Sendstad, M., 2008, "ISO 15926 Templates and the Semantic Web," Position Paper for W3C Workshop on Semantic Web in Energy Industries; Part I: Oil and Gas, Houston, TX, Dec. 9–10.

[4] Eppinger, S. D., and Browning, T. R., 2012, *Design Structure Matrix Methods and Applications*, MIT Press, Cambridge, MA.

[5] Browning, T. R., 2016, "Design Structure Matrix Extensions and Innovations: A Survey and New Opportunities," IEEE Trans. Eng. Manage., **63**(1), pp. 27–52.

[6] Ellinas, C., Allan, N., Durugbo, C., and Johansson, A., 2015, "How Robust Is Your Project? From Local Failures to Global Catastrophes: A Complex Networks Approach to Project Systemic Risk," PLoS One, **10**(11), p. e0142469.

[7] Hodkiewicz, M., and Ho, M. T.-W., 2016, "Cleaning Historical Maintenance Work Order Data for Reliability Analysis," J. Qual. Maint. Eng., **22**(2), pp. 146–163.

[8] Ho, M., 2015, "A Shared Reliability Database for Mobile Mining Equipment," Ph.D. thesis, University of Western Australia, Crawley, Western Australia.

[9] Sexton, T., Hodkiewicz, M., Brundage, M. P., and Smoker, T., 2018, "Benchmarking for Keyword Extraction Methodologies in Maintenance Work Orders," PHM Society Conference, Philadelphia, PA, Sept. 24, Vol. 10.

[10] Kumar, N., Kumar, M., and Singh, M., 2016, "Automated Ontology Generation From a Plain Text Using Statistical and NLP Techniques," Int. J. Syst. Assur. Eng. Manage., **7**(1), pp. 282–293.

[11] Miller, G. A., 1998, *WordNet: An Electronic Lexical Database*, MIT Press, Cambridge, MA.

[12] Speer, R., Chin, J., and Havasi, C., 2017, "Conceptnet 5.5: An Open Multilingual Graph of General Knowledge," Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, Feb. 4–9.

[13] Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.-J., Shamma, D. A., Bernstein, M. S., and Li, F.-F., 2017, "Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations," Int. J. Comput. Vision, **123**(1), pp. 32–73.

[14] Pantförder, D., Schaupp, J., and Vogel-Heuser, B., 2017, "Making Implicit Knowledge Explicit–Acquisition of Plant Staff's Mental Models as a Basis for Developing a Decision Support System," International Conference on Human-Computer Interaction, Vancouver, CA, July 9, Springer, New York, pp. 358–365.

[15] Hadzic, F., Tan, H., and Dillon, T. S., 2010, *Mining of Data with Complex Structures*, Vol. 333, Springer, New York.

[16] Bengio, Y., Courville, A., and Vincent, P., 2013, "Representation Learning: A Review and New Perspectives," IEEE Trans. Pattern Anal. Mach. Intell., **35**(8), pp. 1798–1828.

[17] Strohmaier, M., Körner, C., and Kern, R., 2012, "Understanding Why Users Tag: A Survey of Tagging Motivation Literature and Results From an Empirical Study," Web Semant. Sci., Serv. Agents World Wide Web, **17**(Knowledge Technologies), pp. 1–11.

[18] Macgregor, G., and McCulloch, E., 2006, "Collaborative Tagging as a Knowledge Organisation and Resource Discovery Tool," Lib. Rev., **55**(5), pp. 291–300.

[19] Huang, Y.-M., Huang, Y.-M., Liu, C.-H., and Tsai, C.-C., 2013, "Applying Social Tagging to Manage Cognitive Load in a Web 2.0 Self-Learning Environment," Interac. Learn. Environ., **21**(3), pp. 273–289.

[20] Sexton, T., Brundage, M. P., Hoffman, M., and Morris, K. C., 2017, "Hybrid Datafication of Maintenance Logs From AI-Assisted Human Tags," 2017 IEEE International Conference on Big Data (Big Data), Boston, MA, Dec. 11, IEEE, New York, pp. 1769–1777.

[21] Guimerà, R., and Sales-Pardo, M., 2009, "Missing and Spurious Interactions and the Reconstruction of Complex Networks," Proc. Natl. Acad. Sci., **106**(52), pp. 22073–22078.

[22] Gomez-Rodriguez, M., Leskovec, J., and Krause, A., 2012, "Inferring Networks of Diffusion and Influence," ACM Trans. Knowl. Discovery Data (TKDD), **5**(4), p. 21.

[23] Linderman, S., and Adams, R., 2014, "Discovering Latent Network Structure in Point Process Data," International Conference on Machine Learning, Beijing, China, June 21–26, pp. 1413–1421.

[24] De Paula, Á., Rasul, I., and Souza, P., 2018, "Recovering Social Networks From Panel Data: Identification, Simulations and an Application to Tax Competition," CEPR Discussion Paper No. DP12792.

[25] Raissi, M., Perdikaris, P., and Karniadakis, G. E., 2017, "Machine Learning of Linear Differential Equations Using Gaussian Processes," J. Comput. Phys., **348**(C), pp. 683–693.

[26] Chen, W., Fuge, M., and Chazan, J., 2017, "Design Manifolds Capture the Intrinsic Complexity and Dimension of Design Spaces," ASME J. Mech. Des., **139**(5), p. 051102.

[27] Heymann, P., and Garcia-Molina, H., 2006, "Collaborative Creation of Communal Hierarchical Taxonomies in Social Tagging Systems," Stanford University, Stanford, Technical Report.

[28] Gerlach, M., Peixoto, T. P., and Altmann, E. G., 2018, "A Network Approach to Topic Models," Sci. Adv., **4**(7), p. eaaq1360.

[29] Nickel, M., and Kiela, D., 2017, "Poincaré embeddings for Learning Hierarchical Representations," Advances in Neural Information Processing Systems, Long Beach, CA, Dec. 4–9, pp. 6338–6347.

[30] Nickel, M., and Kiela, D., 2018, "Learning Continuous Hierarchies in the Lorentz Model of Hyperbolic Geometry," arXiv:1806.03417.

[31] Robertson, S., 2004, "Understanding Inverse Document Frequency: On Theoretical Arguments for IDF," J. Doc., **60**(5), pp. 503–520.

[32] Steyvers, M., and Griffiths, T., 2007, "Probabilistic Topic Models," Handb. Latent Semant. Anal., **427**(7), pp. 424–440.

[33] Blei, D. M., Griffiths, T. L., and Jordan, M. I., 2010, "The Nested Chinese Restaurant Process and Bayesian Nonparametric Inference of Topic Hierarchies," J. ACM (JACM), **57**(2), p. 7.

[34] Mikolov, T., Chen, K., Corrado, G., and Dean, J., 2013, "Efficient Estimation of Word Representations in Vector Space," arXiv:1301.3781.

[35] Pennington, J., Socher, R., and Manning, C., 2014, "Glove: Global Vectors for Word Representation," Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, Oct. 25–29, pp. 1532–1543.

[36] Vander Wal, T., 2007, Folksonomy, http://vanderwal.net/folksonomy.html, Accessed May 5, 2019.

[37] Specia, L., and Motta, E., 2007, "Integrating Folksonomies With the Semantic Web," European Semantic Web Conference, Innsbruck, Austria, June 3, Springer, New York, pp. 624–639.

[38] Mousselly-Sergieh, H., Egyed-Zsigmond, E., Gianini, G., Döller, M., Kosch, H., and Pinon, J.-M., 2013, "Tag Similarity in Folksonomies," INFORSID, Vol. **29**, pp. 319–334.

[39] Henschel, A., Woon, W. L., Wachter, T., and Madnick, S., 2009, "Comparison of Generality Based Algorithm Variants for Automatic Taxonomy Generation," International Conference on Innovations in Information Technology, Al Ain, Dec. 15, New York, pp. 160–164.

[40] Chang, J., Gerrish, S., Wang, C., Boyd-Graber, J. L., and Blei, D. M., 2009, "Reading Tea Leaves: How Humans Interpret Topic Models," Advances in Neural Information Processing Systems, Vancouver, BC, Canada, Dec. 6–9, pp. 288–296.

[41] Lv, Y., and Zhai, C., 2009, "Positional Language Models for Information Retrieval," Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, Boston, MA, July 19–23, ACM, pp. 299–306.

[42] Bergamaschi, S., Guerra, F., Rota, S., and Velegrakis, Y., 2011, "A Hidden Markov Model Approach to Keyword-Based Search Over Relational Databases," International Conference on Conceptual Modeling, Brussels, Belgium, Oct. 31, Springer, New York, pp. 411–420.

[43] Mikolov, T., Karafiát, M., Burget, L., Černockỳ, J., and Khudanpur, S., 2010, "Recurrent Neural Network Based Language Model," Eleventh Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, Sept. 26–30.

[44] Jun, K.-S., Zhu, X., Rogers, T. T., Yang, Z., and Yuan, M., 2015, "Human Memory Search as Initial-Visit Emitting Random Walk," Advances in Neural Information Processing Systems, Montreal, Canada, Dec. 7–12, pp. 1072–1080.

[45] Hills, T. T., Todd, P. M., and Jones, M. N., 2015, "Foraging in Semantic Fields: How We Search Through Memory," Top. Cognit. Sci., **7**(3), pp. 513–534.

[46] Schvaneveldt, R. W., Durso, F. T., and Dearholt, D. W., 1989, "*Network Structures in Proximity Data*," *Psychology of Learning and Motivation*, Vol. 24, Academic Press, New York, pp. 249–284.

[47] Haley, B. M., Dong, A., and Tumer, I. Y., 2016, "A Comparison of Network-Based Metrics of Behavioral Degradation in Complex Engineered Systems," ASME J. Mech. Des., **138**(12), p. 121405.

[48] Doyle, P. G., and Snell, J. L., 2000, "Random Walks and Electric Networks," arXiv:math/0001057.

[49] Zemla, J. C., and Austerweil, J. L., 2018, "Estimating Semantic Networks of Groups and Individuals From Fluency Data," Comput. Brain Behav., **1**(1), pp. 36–58.

[50] Walsh, H. S., Dong, A., and Tumer, I. Y., 2019, "An Analysis of Modularity as a Design Rule Using Network Theory," ASME J. Mech. Des., **141**(3), p. 031102.

[51] Saito, T., and Rehmsmeier, M., 2015, "The Precision-Recall Plot Is More Informative Than the Roc Plot When Evaluating Binary Classifiers on Imbalanced Datasets," PLoS One, **10**(3), p. e0118432. An Optional Note.

[52] Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A., 2017, "Automatic Differentiation in Pytorch," NIPS 2017 Workshop Autodiff, Long Beach, CA, Dec. 9.

[53] Schreiber, J., 2018, "Pomegranate: Fast and Flexible Probabilistic Modeling in Python," J. Mach. Learn. Res., **18**(164), pp. 1–6.

[54] Watts, D. J., and Strogatz, S. H., 1998, "Collective Dynamics of 'Small-World' Networks," Nature, **393**(6684), p. 440.

[55] Hodkiewicz, M. R., Batsioudis, Z., Radomiljac, T., and Ho, M. T., 2017, "Why Autonomous Assets Are Good for Reliability—The Impact of 'Operator-Related Component' Failures on Heavy Mobile Equipment Reliability," Annual Conference of the Prognostics and Health Management Society 2017, St. Petersburg, FL, Oct. 2–5.

[56] Sexton, T. B., and Brundage, M. P., 2019, "Nestor: A Tool for Natural Language Annotation of Short Texts," J. Res. NIST, **124**, Article No. 124029.

[57] Zhou, D., Bousquet, O., Lal, T. N., Weston, J., and Schölkopf, B., 2004, "Learning With Local and Global Consistency," Advances in Neural Information Processing Systems, Vancouver, Canada, Dec. 13–18, pp. 321–328.

[58] Anderson, J. R., 2013, *The Architecture of Cognition*, Psychology Press, London.

[59] Shrager, J., Hogg, T., and Huberman, B. A., 1987, "Observation of Phase Transitions in Spreading Activation Networks," Science, **236**(4805), pp. 1092–1094.

[60] Brent, R. P., 1971, "An Algorithm With Guaranteed Convergence for Finding a Zero of a Function," Comput. J., **14**(4), pp. 422–425.

[61] Brundage, M. P., Sexton, T., Hodkiewicz, M., Morris, K., Arinez, J., Ameri, F., Ni, J., and Xiao, G., 2019, "Where Do We Start? Guidance for Technology Implementation in Maintenance Management for Manufacturing," ASME J. Manuf. Sci. Eng., **141**(9), pp. 1–24.

[62] Ivanov, A. O., and Tuzhilin, A. A., 1994, *Minimal Networks: The Steiner Problem and Its Generalizations*, CRC Press, Boca Raton, FL.

[63] Vilnis, L., Li, X., Murty, S., and McCallum, A., 2018, "Probabilistic Embedding of Knowledge Graphs With Box Lattice Measures," arXiv:1805.06627.

[64] Park, B., Kim, D.-S., and Park, H.-J., 2014, "Graph Independent Component Analysis Reveals Repertoires of Intrinsic Network Components in the Human Brain," PLoS One, **9**(1), p. e82873.