TimberSleuth: Visual Anomaly Detection with Human Feedback for Mitigating the Illegal Timber Trade

Information Visualization
XX(X):1–19

DOI: 10.1177/ToBeAssigned
www.sagepub.com/
SAGE

Debanjan Datta[1], Nathan Self[1], John Simeone[23], Amelia Meadows[2], Willow Outhwaite[5], Linda Walker[2], Niklas Elmqvist[4] and Naren Ramkrishnan[1]

Abstract

Detecting illegal shipments in the global timber trade poses a massive challenge to enforcement agencies. The massive volume and complexity of timber shipments and obfuscations within international trade data, intentional or not, necessitates an automated system to aid in detecting specific shipments that potentially contain illegally harvested wood. To address these requirements we build a novel human-in-the-loop visual analytics system called TIMBERSLEUTH. TimberSleuth uses a novel scoring model reinforced through human feedback to improve upon the relevance of the results of the system while using an off-the-shelf anomaly detection model. Detailed evaluation is performed using real data with synthetic anomalies to test the machine intelligence that drives the system. We design interactive visualizations to enable analysis of pertinent details of anomalous trade records so that analysts can determine if a record is relevant and provide iterative feedback. This feedback is utilized by the machine learning model to improve the precision of the output.

Keywords

Visual analytics, anomaly detection, active learning, machine learning.

Introduction

Illegal logging is estimated to be the third largest category of transnational crime, with an annual retail value estimated to be between $52 and $157 billion[1] and connections to illicit financial flows.[2] These unsustainable practices are not only detrimental to biodiversity[3] but negatively impact local economies and national security[4]. This activity poses an urgent problem for enforcement agencies and environmental conservation agencies. The trend of land use change and associated terrestrial biodiversity loss is particularly perceptible in tropical ecoregions and developing countries[5]. The United States, the world's largest importer of wood and forest products, imported $51.5 billion of solid-wood forest products in 2017 which accounted for 22% of all global imports. However, monitoring this trade remains a challenge due to *(a)* volume and complexity of trade data[6]; *(b)* short investigative window and high cost to detain cargo; and *(c)* no timber-specific tools for live targeting and long term trend analysis. Thus there is a critical need for a decision support system to aid enforcement agencies in detecting and acting upon shipments with potentially illegal timber.

The task of detecting and investigating trade in suspicious timber can be formulated as a *visual anomaly detection* task, akin to many fraud detection tasks. Anomaly detection has been proposed for detection of trade in other high-risk commodities to support customs enforcement.[7] However, even with interactive visualization support, the sheer scale of global shipping—hundreds of thousands of shipments per month—makes manual scrutiny and exploration to identify potentially illegal shipments infeasible.

We propose TIMBERSLEUTH (Figure 1), a scalable visual analytics approach that combines a machine learning model for anomaly detection, interactive visualization, and human input to inform the user rather than solely relying on human cognitive effort. Off-the-shelf machine learning models for anomaly detection do not tend to explain why an identified record is considered anomalous or suspicious, and also do not provide any feedback mechanism. In TimberSleuth, we provide explanations for the model output and solicit feedback from the user based on those explanations. Our contributions in this paper include

(I) An integrated visual anomaly detection system for detecting suspicious timber shipments that combines domain expertise, human-in-the-loop anomaly detection, and visual analytics.

(II) A scoring model for feedback driven anomaly detection using an off-the-shelf anomaly detection model that is demonstrated to perform better than standard approaches.

(III) An embedding-based approach to provide explanations for anomalies that can be utilized in the feedback process.

[1]Department of Computer Science, Virginia Tech, Virginia, United States
[2]World Wildlife Fund, Washington, DC, District of Columbia, United States
[3]Simeone Consulting, LLC, United States
[4]College of Information Studies, University of Maryland, College Park, College Park, Maryland, United States, [5]TRAFFIC International, Cambridge, Cambridgeshire, UK

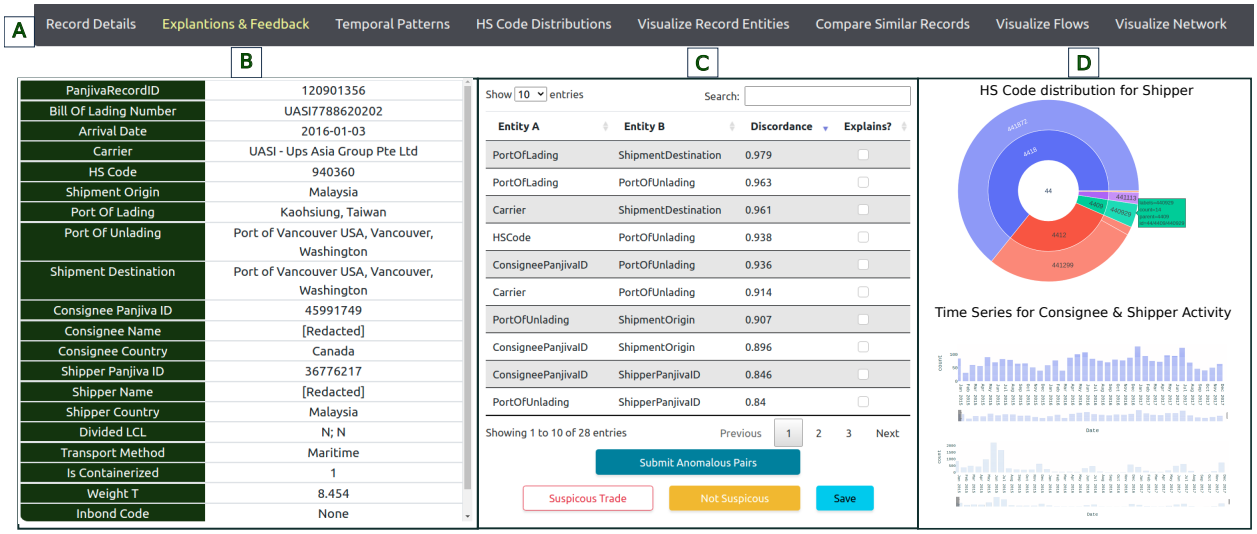

Figure 1. System overview. TIMBERSLEUTH detects suspicious timber trades, and provides explanations to solicit feedback for the visual anomaly detection system. (A) The persistent navigation bar at top allows quick navigation between visual components. (B) The details of the record shows the complete set of attributes to the analyst. (C) The explanations are provided as a ranked list to show the most probable reason for the record being judged anomalous. The interactive elements allow saving the input, and also provide overall confirmation. (D) A few of the visual components that are integrated into the system to allow users to investigate the record in detail.

(IV) Task-specific visual components that use scalable machine learning techniques to effectively aggregate and provide multiple views for trade data with high-dimensional categorical attributes based on domain expert feedback.

Related Work

Recent work has demonstrated how visual analytics can be utilized towards incorporation of human knowledge into machine learning systems.[8,9] We group our literature review into three subsections, each pertaining to research areas consistent with the components used in our overall system.

Visual Analytics for Anomaly Detection

One of the key goals of TimerSleuth is to provide visual analytics tools for the underlying machine learning task—i.e., unsupervised anomaly detection—to aid the end user to provide investigate individual records and provide feedback. Visual analytics has been utilized in multiple applications of anomaly detection to alleviate the lack of ground truth labels. Because anomalies are domain-specific, in this section we discuss some systems that have proposed approaches tailored to suit their respective nature of the data and its anomalies. Thom et al.[10] and Cao et al.[11] present visual anomaly detection systems for malicious activity on Twitter. *Voila*[12] is a system that performs interactive anomaly detection on spatiotemporal data obtained from a streaming source, allowing for a human in the loop approach. *Z-Glyph*[13] explores a family of glyphs with the intent to visualize outliers pertaining to multiple datasets, that help human judgement and interpretation of outliers in data. *Situ*[14] presents a visual analytics framework for cybersecurity. Xie et al.[15] present a visual analytics framework for detecting run-time behavior in high performance computing environments. Wilkinson[16] outlines specific approaches for understanding and visualizing outliers in large scale data. Ko et al.[17] focus on the integration of multiple visual analytics techniques for analysis and exploration of high-dimensional and multivariate network data in multiple domains including shipping and logistics. *OoDAnalyzer*[18] provides an interactive visual system to understand distribution samples for image data. These prior efforts rely on data sources that are distinctly different from large scale trade data, so the techniques do not translate to our use case. Moreover, they do not focus on specifically human-in-the-loop systems, which is imperative for our use case.

Visual Analysis for Tabular Data

Tabular data is difficult to comprehend due to the lack of implicit structure among the attributes, especially with categorical attributes—where there is no intra-attribute ordering. There have been prior works that present approaches to visualization of tabular data, given it's ubiquitous nature which we have briefly discussed below.

Bertifier presented by Perin et al.[19] visualizes numerical tabular data through simultaneous visual encoding of cell values and grouping rows and columns with similar values. Lex et al.[20] presents a visualization approach called *UpSets* for sets that can be extracted from tabular data. *VisBricks*[21] provides a framework to explore large heterogeneous data using clustering and aggregation of relationships between subsets. *Keshif*[22] is a framework for quickly exploring tabular data through summarization and aggregation characteristics based on data types.

SMARTExplore[23] is another recent framework that focuses on tabular data. Apart from user-driven exploration, it provides summary and descriptive statistics combined with intuitive visual representations and automated analysis to show outliers, clusters, and correlations. *Taggle*[24] follows the design paradigm of prior frameworks. It incorporates coordinated multiple views, with a variety of descriptive statistics and user driven operations for querying, filtering

and aggregation. Systems such as *Snap-together*[25] and *Improvise*[26] target expert users and developers, providing both flexibility and a wide range of use cases and patterns. While these works present a myriad of approaches, upon which we build, their exact use cases and target data types deviate from our intended objective.

Human-in-the-loop Anomaly Detection

The key impetus in improving anomaly detection systems through a human-in-the-loop (HITL) process is to bridge the gap between application specific interpretation of anomalies. The output of anomaly detection systems are based on expected patterns and the underlying statistical properties of data. In scenarios where the budget for labelling is very limited compared to the scale of data, *active learning* aims to obtain informative labels to iteratively train the underlying machine learning model. Pelleg and Moore[27] present an anomaly detection system where active learning is used to extract only useful anomalies although the scale, nature, and data complexity are of comparatively lower scale. Similar HITL systems have been proposed by Abe et al.[28] and He and Caronell[29]. Ghani and Kumar[30] present an interactive anomaly detection system for fraudulent insurance claims that uses human feedback in an active learning setting to train a classifier. *AI*^{2}[31] is a cybersecurity-specific HITL system that relies on large scale feedback to train supervised anomaly detection models, and is thus different from our setting.

Active Anomaly Detection (AAD)[32] is based on *LODA*[33] and tree-based ensembles such as that proposed by Das et al.[34] and has been shown to perform well for tabular data with numeric attributes. The approach presented by Siddiqui et al.[35] and *GLAD*[36] are frameworks for HITL anomaly detection based on online learning for weights of components of an ensemble. *OJRank*[37] uses the feedback regarding top anomalous instance to iteratively reweigh an ensemble of anomaly detectors. The objective of finding anomalies that are similar to ones already encountered as adopted OJRank is the same as in our case. However this approach uses a sampling based approach and can take feedback of a single instance only—which is not suitable for our use case. The concept of similarity or clustering among anomalies, which our model utilizes, has been explored in prior approaches presented by Ghani et al.[30] and Lamba and Akoglu[37]. However they are not directly applicable to our scenario. Ghani et al.[30] relies on iterative classification, however the there is sparsity of data obtained through feedback to train such classifiers. OJRank[37] as explained above does not directly satisfy our requirements, and also these models are intended for data with real-valued attributes. While categorical tabular data can be encoded as real valued input by methods like one-hot encoding, the high dimensionality of our target data makes it infeasible.

Kong et al.[38] presents a system for HITL anomaly detection for time series data that treats the underling anomaly detector as an off-the-shelf system and uses clustering (K-means) based approach that is not effective for high dimensional categorical data. The the task of HITL anomaly detection for high dimensional multivariate categorical data remains an unsolved challenge, which we attempt to address here. While our problem setting is similar to Kong et al.[38], the exact approach is not applicable.

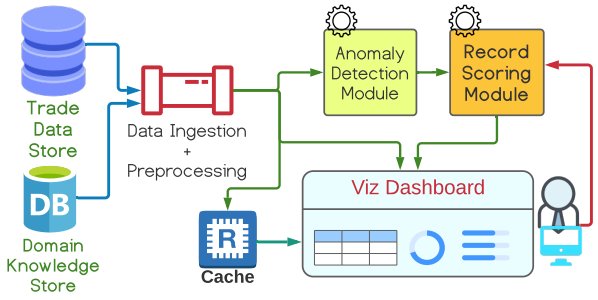

Figure 2. TIMBERSLEUTH system architecture comprising Anomaly Detection module, Record Scoring Model (re-ranking) module, and the Visualization dashboard.

System Design Process

We begin by formally defining the objective that is intended to be accomplished in this work:

Problem Description *Design a visual analytics system that can display details for a set of records that are flagged by an underlying anomaly detection system and provide insights so as to solicit iterative feedback in order to improve relevance of the records to the application scenario in subsequent iterations. The visual analytics system should have an active learning component that can learn patterns from user feedback to improve relevance of output iteratively.*

The current systems in use, while not directly accessible by researchers, are known to be not utilizing automated methods. In fact most of the checks to the best our knowledge are performed manually, using personnel on the ground and physical forms which are examined by personnel. Their knowledge of past infractions, known anecdotal prior evidence and expert judgement drives the process of flagging suspicious shipments. Our system was closely developed in collaboration with domain experts from ecological conservation agencies. The collaborating domain experts have significant experience in working with illegal timber trade and shipment records. They were chosen as collaborators due to their knowledge of how the intended end-users operate, and their in-depth knowledge of how potentially illegal timber trade practices, as well as their knowledge of endangered flora and fauna. It is also important to note that while experts have fine grained domain knowledge, obtaining security clearance in US to access their internal tools, processesm and documents is subject to certain restrictions and requires justifications that researchers and conservation organizations do not have access to.

Data Description

The design of a visual analytics system is connected to the underlying data, which is the case here as well. The data used in this system are shipments records, each of which is an individual transaction instance between companies, describing the type of goods and products traded. Our system is designed to work with a real-world trade dataset on United States trade imports from Panjiva trade data.[39] Shipment

RecordID	Origin	Shipper	Port Of Lading	Carrier	Port Of Unlading	HS Code	Consignee	Destination
10001	Xijiang	Shipper A	Shanghai	Maersk	Los Angeles	421022	Consignee A	Tahoe
10002	Borneo	Shipper X	Singapore	Hyundai	New York	440710	Consignee Y	North Carolina

Figure 3. Examples of trade records for U.S. import data. Entity names are anonymized by replacement.

data in the form of *Bill of Lading* manifests are utilized by customs enforcement agencies to regulate import and determine taxes. However, these shipment records contain important details that describe the different aspects of the global supply chain. Specifically the attributes of the data are (i) Consignee; (ii) Shipper; (iii) Port of Lading; (iv) Port of Unlading; (v) HS Code; (vi) Shipment Origin; (vii) Shipment Destination; and (viii) Carrier. Figure 3 demonstrates some example records with the schema. An important point to note is that all the attributes are categorical in nature. This directs our design and methodology choices.

The first stage of data ingestion involves selecting pertinent attributes from the raw tabular data. This is done through understanding the attribute semantics with the help of domain and data experts. This is followed by data cleaning and generally preprocessing the data to convert it to a format utilizable by a machine learning pipeline. It is important to note that this data is sizeable, on the order of 10^5 records per month and has significant complexity. In preprocessing the data, we follow methods adopted in prior literature such as those presented in Das et al.[40], where very sparse entities are discarded to prevent bias. We perform data curation specific to domain knowledge from external sources which is described in next sections.

Formative Study

We conducted a formative study with a sample of our domain expert audience to understand how to design our tool.

Participants. We engaged three domain experts with prior experience in dealing with illegal timber trade and forestry. They have been working with enforcement agencies over several years and belong to one of the most prominent conservationist organizations. They were well situated to communicate the issues faced by the intended end users—specifically the enforcement agencies such as U.S. Customs and Border Protection. Unfortunately, given the sensitive nature of the data and the project, we were not given access to the actual analysts "on the ground," but instead team members who had prior experience with these activities.

Method. Our formative study was performed as individual interviews with domain experts followed by a collaborative discussion with the whole group. Experts were presented with questions that related to (i) what are the key bottleneck faced by end users; (ii) what are the information communication modes (such as natural language, visual) would most likely assist in the given task; (iii) how difficult is the current manual inspection process; and (iv) what level of technical proficiency do the end users have.

Findings. We here summarize the findings that informed our the overall design. Firstly, participants felt that the core objective of such a system is to aid in the investigation of shipments of interest that can potentially contain illegal timber, allowing for greater efficiency and effectiveness for the end users. We present expert scores on a 1–5 Likert scale.

Experts noted that the system should be be usable and intuitive for analysts who have expertise in domain knowledge of illegal timber (score: 4) and sufficient technical expertise to use web-tools (score: 3). There is need for the system to have adequate visual cues, and provide potential explanations towards why records are highlighted as interested. These were rated on average at 4 and 5 respectively. Experts pointed out that it is especially important to have explanations, context, and appropriate visual tools to analyze why a record is highlighted, especially for seemingly legal timber products. There can be cases of blatant illegality, fraud (deliberate mislabelling),[41] or potential clandestine activity in a record, and records with similar attributes or context are important.

Design Requirements

Based on our formative study, the design requirements for the overall system can be summarized as follows:

R1 *Automatic anomaly detection:* Given the scale and scope of the data, experts felt that the proposed system must be based on an automated anomaly detection;

R2 *Explainability and transparency:* To improve accuracy and oversight, the proposed system should visually explain the rationale for its decisions to the user;

R3 *Human control and supervision:* Experts asked for a visual interface for the user to (a) view records ordered by score, (b) explore and investigate individual records, and (c) provide feedback and update the underlying model.

We note that our design rationale was founded on the principles of human-centered artificial intelligence (HCAI),[42] where the goal is to achieve **both** high automation as well as high human control.

Continuous Evaluation

We continued working with our expert panel even after completing the initial formative study throughout the duration of the project. However, we note again that these experts were part of our collaborating team and not the front-line specific analysts from enforcement agencies. Unfortunately, these analysts were far too busy with their day-to-day work to be able to participate in this study. Furthermore, the sensitive nature of the activities and the data also meant that these analysts were effectively barred from communicating with the research team.

Domain Knowledge Incorporation

Goods and products involved in global trade are tracked using the Harmonized Schedule (HS) code nomenclature and product classification system. Many countries add up to four additional digits (up to ten total) to further the specificity of product classifications based on HS codes. We utilize the first six digits of HS Code, which are globally standardized, and their associated descriptions. We obtain the ontology and data for HS Codes from open source repositories containing

text descriptions of products, which can include scientific names, family and common names of timber species.

We extract specific six-digit HS codes representing known high risk timber using text processing techniques such as regular expression matching and n-gram based keyword matches, as well as collaborating domain experts' inputs on HS Code text descriptions. All HS codes for solid wood and products containing solid wood (like furniture) are used to select trade records for our system. HS codes covered by legislation such as the Lacey Act and data on country-specific logging and export bans are also obtained. Although these curated HS Codes may contain high risk species, they can correspond to such a large number shipments that simple rule-set based matching is neither analyzable nor actionable by end users. Data from sources including CITES (Convention on International Trade in Endangered Species), IUCN (International Union for Conservation of Nature),[43] and WWF (World Wildlife Fund) contain scientific names, common names, and country or region of harvest information, which are extracted for the application. We match these to HS Codes, allowing us to flag HS Codes in records which are highlighted by the underling machine learning algorithm.

System Overview: TimberSleuth

TIMBERSLEUTH consists of two major components: the algorithmic pipeline and the interactive visual analytics interface. The algorithmic pipeline or the back-end, hereon referred to as the **machine intelligence**, comprises

(i) Domain data ingestion module;

(ii) Shipment data preprocessing module; and

(iii) Machine learning module for record scoring.

The user interface, or the front-end, hereon referred to as the **visualization dashboard**, consists of

(i) Data processing and caching for visual analytics;

(ii) Visual analysis component modules; and

(iii) Bridge modules to record feedback and synchronize with machine intelligence.

The overall architecture is shown in Figure 2 with the *backend* and *frontend* components and how they are connected. In Figure 2, the *domain knowledge store* is used primarily for storing the timber specific data from which we obtain the relevant HS Codes. The *trade data store* is a database for storing and retrieving the trade records.

The first step of the data processing is anomaly detection, where the records are assigned a real-valued anomaly score. This computation is performed once, and is the starting point for the subsequent steps. The *Record Scoring Module* and the visualization modules are the iterative parts of the system.

Overview. Below we first discuss machine intelligence followed by the human-in-the-loop framework. We then describe the interactive visual dashboard and the visualization components. We close with an example scenario.

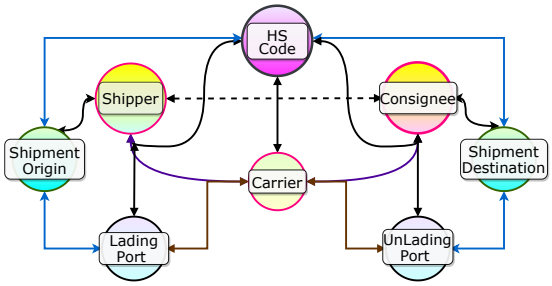

Figure 4. Trade records schema graph. Nodes represent the different domains and the relationships between them are represented as edges. Table 1 lists the metapaths constructed using this.

Machine Intelligence

The objective for the machine intelligence component is to:

(i) Perform anomaly detection on the trade records, including providing an initial ranking of records based on their anomaly score

(ii) Utilize human feedback to iteratively improve the ranking of the results in terms of their relevance, such that anomalous records similar to those that have been previously annotated as relevant are ranked higher in subsequent iterations.

These objectives pertain to all three design requirements **R1**, **R2**, and **R3**. We focus on *precision at the top* as presented in Kar et al.[44] and Lamba and Akoglu[37], where the objective is to improve precision in the top-ranked items (records) in the ranked list iteratively. Precision here refers to the ratio of relevant records (w.r.t application scenario) to the total number of records that are selected. In practical implementations of anomaly detection systems, a user-specified percentage or user-specified count of records are chosen for further investigation given capacity or budget constraints. The key challenge is improving the precision of such highly ranked records by utilizing continuous human feedback.

Data Model Preliminaries

Tabular Records. The shipment records are in tabular format, with each row describing an instance. Tabular data can be formally represented in terms of *domains* and *entities*. A *domain* or attribute U is defined as a set of elements sharing a common property, e.g. *Port*. The j^{th} domain U_j consists of a set of *entities*, denoted as e^i_j. For instance entities *New York* and *Houston* belong to the domain *Port*. The count of entities in a domain is termed as *arity* or *cardinality* of the domain. For instance, the cardinality for the domains *Consignee*, *Shipper* and *Carrier* are in order of 10^4, 10^4 and 10^2 respectively, for a month of shipment data. A *multi-relation* or *record* (r) is a tuple of entities, with one entity belonging to each of the l domains. *Context* is defined as the reference group of entities with which an entity occurs, implying an entity can be present in multiple contexts.

Network View of Tabular Data. An alternate intuitive representation of the *entities* and their relationships is in form of a Heterogeneous Information Network (HIN). These

relationships describe the entities that constitute the supply chain—such as shippers, ports, and destination and origin of commodities, and are understood with the help of domain experts. Such a network is heterogeneous since there are nodes representing multiple entity types (*domains*) and multiple relationships that exist between them, as shown in Figure 4.

Definition *A Heterogeneous Information Network is defined as a graph* $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$*, where* $\mathcal{V}$ *and* $\mathcal{E}$ *are sets of vertices and edges, respectively.* $e \in \mathcal{E}$ *belong to one of* r *types where* $\mathcal{E} = \{\mathcal{E}^1 ... \mathcal{E}^r\} \subseteq \mathcal{V} \times \mathcal{V}$.

Definition *A* ***metapath*** [45] *is a path defined on* $\mathcal{G}$ *of the form* $\mathcal{V}_i \leftrightarrow \mathcal{V}_j \ldots \leftrightarrow \mathcal{V}_k$*, where* i, j, k *are distinct node types, which defines a composite relation among the node types.*

The metapaths capture the semantic relationships between nodes of different types. We consider metapaths which are symmetric, since we consider the relationships to be undirected. The motivation for this use of metapaths is to capture inherent relationships and avoid edges that do not represent valid relationships. The list of relevant metapaths for United States trade data are shown in Table 1. We utilize both these views of data collaboratively in our system. HINs have been studied in context of relationship or pattern extractions, as well as for anomaly fraud detection.[46,47]

Detecting Initial Anomalous Records

Detecting sparse anomalies in tabular data is a challenging task, especially where the data objects are complex and do not have a single compact representation. For this task (**R1**) we adopt Multirelational Embedding based-Anomaly Detection (*MEAD*)[48], which is specifically designed for tabular categorical data with high cardinality. *MEAD* is an embedding based model that captures the likelihood of a record based on entities and its given context. *MEAD* is trained with a modified Noise Contrastive Estimation objective using negative samples, which is efficient and scalable. This provides a ranked list of records, ordered by likelihood scores, where lower scored records are deemed anomalous. Anomaly detection systems for categorical tabular data are based on approaches such as itemset mining[49], which are unsuitable for a deployed application that has upper bounds on model training time.

MEAD represents entities (of different domains) as low dimension embeddings. In MEAD the embedding vectors that represent commonly co-occurring entities are similarly oriented. This is because the model is trained with an objective such that the sum of vectors of entities corresponding to a record in the expected data distribution (training set) has a higher sum compared to negative samples or noise. More specifically, each score is modelled as the probability of such a record belonging to the data distribution through penalizing records where the entities do not co-occur in training data. If a record contains a set of entities that are not expected to co-occur, the vectors representing these entities are not oriented in the same direction as the context. We utilize this property for providing end users with interpretability as to why a record might be deemed relevant or anomalous with respect to the application. It is important to note that we assume that MEAD, being a state-of-the-art model, is effective in finding anomalies. Our main focus here is not the anomaly detection model but the later stages of the system that builds upon it.

Combining Multiple Models

Ensemble methods have been used in many machine learning systems to improve robustness and performance[50]. Combining outputs from multiple instances of the anomaly detection model with different key hyperparameters can potentially provide more robust results. An efficient approach to combine multiple ranked lists in an unsupervised manner is *Borda Count*,[51] which determines the final rank of objects based on their positions in the input rankings. The items in the combined output ranking from *Borda Count* are sorted according to the numbers of items that are ranked below them in the input ranking lists. Thus, records which are ranked high (anomalous) by multiple model instances will be scored high in the aggregated ranking.

To that end, we train multiple instances of MEAD with different embedding dimensions and combine their results into a single ranked list of records, where the highest ranked records are anomalous. It is a standard practice for unsupervised anomaly detection systems to use a user-provided threshold, such as $2nd$ or $5th$ percentile of the normalized scores, or to select a predefined user-specified number of records to select the highest ranked (lowest likelihood) scored samples, which are treated as anomalous. We adopt the second approach, choosing the k most anomalous records, where k is set to 5000. The ensemble potentially provides more robust anomaly scores, and this aids the feedback process.

Human-in-the-Loop Framework

The metric that we intend to optimize is precision at the top in the ranked list of anomalies, quantified by *precision@b* where b is a user-specified parameter, as specified in requirements **R1** and **R2**. It is infeasible to directly incorporate domain knowledge into off-the-shelf anomaly detection systems such as MEAD.

Highlighting similar records based on feedback reduces the cognitive load of the end user in investigating records, since patterns exist among anomalies and similar instances occupy the same region in the latent data space—as discussed by Ghani et al.[30] and He et al.[29] This offers a better (more precise) set of samples to provide positive feedback upon which can help the iterative process.

The objective here is to find a relative ordering among the output of the anomaly detection system, such that more relevant records are scored higher. In the first step, an application specific threshold (t) is used to select the highest scored records ranked by anomaly score, i.e. records with lowest likelihood from the output of the anomaly detection model as potentially anomalous. At each subsequent feedback iteration, these records are re-ranked by the scoring model.

Schema of Feedback

Let us suppose a record T1:{Consignee:C_1, Origin:O_1, HSCode:H_1, Destination:D_1, Shipper:S_1} is a relevant

Table 1. Metapaths overview. Metapaths used to capture relationship between entities, designed following the HIN schema view of the data. These are utilized for computing vector representations that enable measuring relative proximities between entities of the same and different domains (types).

Serial	Metapath Structure			
1	Consignee	↔ HS Code	↔ Shipper	
2	Shipper	↔ HS Code	↔ Consignee	
3	Consignee	↔ Carrier	↔ Shipper	
4	Carrier	↔ Port of Lading	↔ Shipment Origin	
5	Carrier	↔ Port of Unlading	↔ Shipment Destination	
6	Shipment Destination	↔ HS Code	↔ Shipment Origin	
7	Shipment Origin	↔ HS Code	↔ Shipment Destination	
8	Port of Lading	↔ Shipment Origin	↔ HSCode	
9	Port of Unlading	↔ Shipment Destination	↔ HSCode	
10	HS Code	↔ Carrier	↔ Port of Unlading	↔ Consignee
11	HS Code	↔ Carrier	↔ Port of Lading	↔ Shipper

Algorithm 1: Record Scoring Model

```
// Scoring Model Initialization
```
input: entity embedding function f_e; feature interaction function f_{ij}; set of all m domains D_m; set of entities: $e \in domain(D_i)$; set of relevant domains $D_s \in D_m$ (where entities should be flagged); Samples : X,Y$\in \{+1, -1\}$

$\mathfrak{M}.f_e \leftarrow f_e$; $\mathfrak{M}.g_{ij} \leftarrow g_{ij}$;

for $d \in D_s$ **do**

  $\mathfrak{M}.binaryFeatVect[d] \leftarrow [0]^{|d|}$

end

Initialize interaction feature weights $\mathfrak{M}.W_{ij} \sim N(0,1)$

Minimize $\frac{1}{2}\left(y_r - \Sigma_{ij}\mathfrak{M}.W_{ij}^T g_{ij}(f_e^i(x_p), f_e^j(x_q))\right)^2$

```
// Scoring Model Update using
   Feedback
```
input: $\mathfrak{M}$; gradient clip value:γ=0.1; $\beta = 0.5$

for *each iteration of feedback with labelled records* (R_b) **do**

  for *each record* $r \in R_b$ *labelled True* **do**

    $\hat{y}_r \leftarrow 2$

    $binaryFeatVect[D_i][e_i^j] \leftarrow 1$ if $D_i \in D_s$

    $\nu \leftarrow$ Count of entity pair (e_i, e_j) marked as cause of anomaly in r, with domains i,j

    for *each entity pair* (e_i, e_j) **do**

      if (e_i, e_j) *flagged* **then**

        $p_{ij} \leftarrow \beta * \nu / |W_{ij}|$

      else

        $p_{ij} \leftarrow 0$

      end

    end

  end

  for *each record* $r \in R_b$ *labelled False* **do**

    $\hat{y}_r \leftarrow 0$; $p_{ij} \leftarrow 1/|W_{ij}|$

  end

end

Calculate average gradient ∇J, Clip-Gradients(∇J,γ);

$\mathfrak{M}.W_{ij} \leftarrow \mathfrak{M}.W_{ij} - p_{ij}\eta\nabla J_{ij}$

return $\mathfrak{M}$

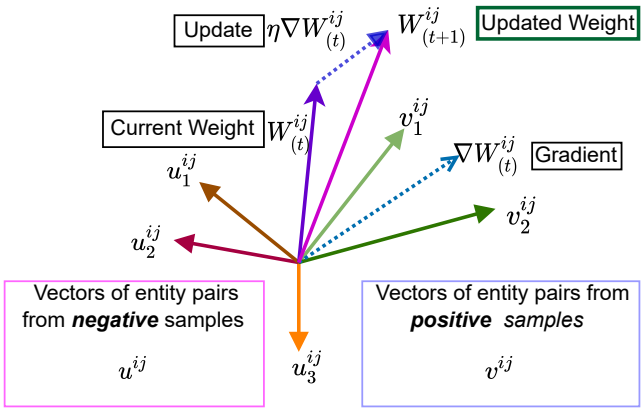

Figure 5. Update mechanism. Second-order feature interaction weight W_{ij} update mechanism in the **Record Scoring Model**. Here u^{ij} and v^{ij} are outputs of g_{ij}, where the entities belong to domains i,j. Note that v^{ij} and u^{ij} are annotated as positive and negative entity pairs towards cause of an anomaly.

anomaly. The simplest way to provide feedback would be to flag the Shipper and/or Consignee, i.e. based on white-listing and black-listing companies. But this may overlook the underlying cause of why the record was judged a relevant anomaly and does not take into account the contextual information. If another transaction T2 contains a similar company (e.g. sister company) Shipper:S2 and HSCode:H1, it would thus evade detection. The most atomic unit of capturing contextual information is through observing *binary relationships* or entity pairs, since even higher order interactions can be decomposed in terms of binary relationships.

Thus we have two requirements when designing the scoring model to utilize feedback: *a*) incorporate feedback on specific entities and entity pairs in updating results; and *b*) score records which have entities and/or entity pairs similar to ones that have been flagged.

The user thus provides the following inputs, based on the understanding of the record details that are enabled by the visualizations.

(*i*) Entity of interest (e.g. Shipper S); and

(*ii*) Entity pairs that possibly cause the record to be anomalous (e.g Port A, Carrier C).

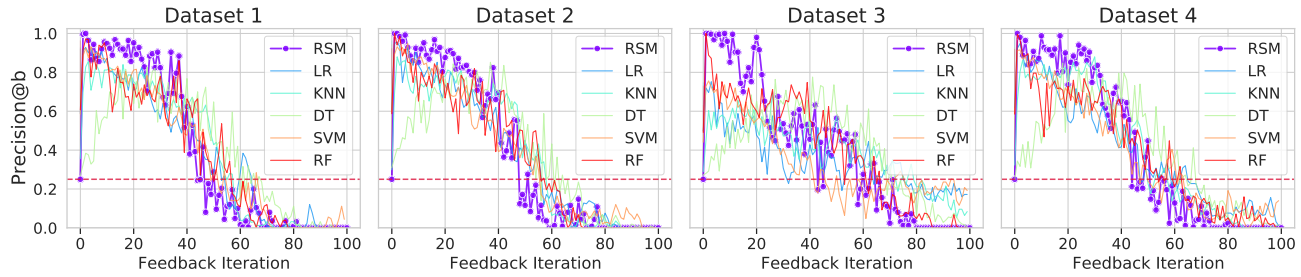

Figure 6. Performance comparison. Baseline and scoring model for the four synthetic datasets. The metric is precision@b (b=25), i.e. the precision at the top. The performance curve for RSM(●) shows a clear advantage over competing methods for all datasets. The dotted horizontal line at 0.25 shows E[precision] without any human input, which is the ratio of relevant labels. Abbreviations used: RSM(●): Record Scoring model, LR: Logistic Regression, DT: Decision Tree, RF: Random Forest classifier, KNN: k-Nearest Neighbor classifier, SVM: linear SVM.

Record Scoring Model Details

Given the limited amount of feedback available to train the scoring model, we propose a data efficient additive model. Generalized linear and additive models such as factorization machines[52] and GA^2M[53] have been proposed for classification and ranking tasks on large scale data. Such models are expressive, yet are inherently interpretable in terms of feature importance or contribution. We refer to our model as the *Record Scoring Model* (RSM).

While the anomaly detection model in the prior stage (MEAD) is unsupervised, RSM is semi-supervised following an active learning paradigm. RSM is trained with a regression objective, where sign(+/−) of the predicted value indicates label. RSM uses the set of labelled records at each feedback iteration. To capture the effect of **individual entities** in the relevant domains (*Consignee* and *Shipper* in our example), a binary feature vector is used. A scaling hyperparameter (α) controls the importance of this component. We use $\alpha = 0.1$ for our experiments.

The second component is **cross-entity feature interaction**. Representing entities as one-hot encoded vectors for cross entity interaction feature leads to a very high dimensional sparse feature space, and simpler approaches for dimensionality reduction such as feature hashing do not preserve inter-entity similarity.

Thus the entities are represented as L_2 normalized embedding vectors which are obtained from the graph schema by applying *Metapath2Vec*[54] for capturing cross entity interaction features. Specifically, let the L_2 normalized entity embedding function be denoted as $f_e^i(e_p) \rightarrow R^d$ where $e_p \in domain_i$. The feature interaction function $g_{ij}(f_e^i(e_p), f_e^j(e_q)) \rightarrow R^z; e_p \in domain_i, e_q \in domain_j$ can be chosen as concatenation, mean or Hadamard product. It is empirically found that concatenation works well in our case. The model is described in Equation 1. Here $\mathbb{1}(x_m)$ is an indicator function that is set to 1 if the entity has been previously flagged.

One important thing to note here is that W_{ij}—the weight for each domain pair capturing entity interaction—is updated separately, with the annotated samples and the associated explanations. This enables Record Scoring Model to explicitly high scores to anomalous records that are similar to records that have been annotated as relevant in previous iterations.

$$y_r = \alpha \Sigma_m \mathbb{1}(x_m) + \Sigma_{i,j} W_{ij}^T g_{ij}(f_e^i(x_p), f_e^j(x_q)) \quad (1)$$

The scoring model is updated at each iteration in an online manner. This online learning problem is a modification of an online convex optimization approach. It is important to note that prior work exists where models based on online learning have been proposed for active anomaly detection scenarios[35]. However the model and nature of data are different from our case. Specifically previous models are not suited to tabular data with high dimensional categorical attributes.

Iterative Retraining of Record Scoring Model

Since RSM is a semi-supervised model, labels are required to train the model. Initially there are no labelled samples. Here a record is labelled *True* if is relevant, i.e. if it is actually illegal and/or suspicious. A record is labelled *False* if it is not relevant to the applicant scenario. Note that these labels are assigned by annotators or users during the feedback process.

Initialization of RSM. The weights of the RSM model are initialized prior to the first iteration of feedback. To provide a good starting point, these weights are initialized by training the model as follows. An initial set of records which are ranked most anomalous by the anomaly detection model (MEAD) are taken as a proxy set of positively labelled samples, assigned a score of +1. Correspondingly, a set of records ranked lowest by the anomaly detection model i.e. nominal are sampled and assigned a score of −1. These are used as an proxy set of negatively labelled samples, which are not relevant. Using a regression objective with mean squared error loss, we train the RSM to predict the record score y_r. Weight decay[55] is used for regularization to ensure of the L_2 norm of the cross entity feature weights are low.

Updating RSM With Feedback. During the first and subsequent iterations of feedback, annotated samples are obtained. These are utilized in retraining and updating the RSM. In this phase, the target score (y_r) is updated to +2 for samples with label *True* and for the records with label=$False$ the target scorey_r is set to 0. The absolute magnitude of the target scores are not important, but only the relative scores among records are used for ranking them.

An important point to note is that for a record labelled relevant in user feedback, only the weights of entity pair

interaction(s) which have been flagged as a *causal factor* or explanation should be increased while others should remain unchanged. This causes similar entity pairs to be assigned a higher score in subsequent inference steps by the RSM. For records which are marked not relevant (*False*), all contribution of the entity interaction features should be reduced albeit to a lower degree. Confidence Weighted Learning [56] explores a similar idea, however it is not directly applicable here.

The weights of the RSM are updated using *batch gradient descent*, and *gradient clipping* [57] is used during weight updates to prevent instability in training process. Like any gradient based approach, the learning rate is an important hyperparameter. It is initially set to 0.9 and linearly decayed. The mechanism of update is demonstrated in Figure 5.

It is also important to note that in RSM W_{ij} is a vector, and its absolute value or magnitude does not directly help in the interpretation as contribution of coefficients, as in other generalized linear models. However, the fact that the feature component is obtained using g_{ij} for the candidate set of all possible entity pairs from the domains i and j can provide intuitions towards which entity pairs from domains i and j are present in relevant anomalies.

At the end of each feedback iteration, the scoring model is re-trained and is used to obtain the updated scores of the remaining unlabelled records. The procedural steps of the update are outlined in Algorithm 1.

Record Scoring Model Evaluation

We evaluate the proposed RSM model as part of the overall architecture, whose inputs are from the anomaly detection model (MEAD). The objective is to determine the effectiveness of RSM such that the relevant records are ranked higher (percolate to the top) with successive iterations of feedback so as to minimize human effort.

Datasets. Four sets of U.S. import data are extracted from the larger data corpus to perform experimental evaluation. These are hereon referred to as Dataset-{1,2,3,4}. For each dataset, the first four months of data are used as training set and the next two months are used for testing. The training set is needed for training the anomaly detection model MEAD. We assume, following prior works, that the training data is approximately clean and does not contain anomalies.

Experimental Setup. Since we use real world data, we do not have ground truth anomalies or known data instances that are relevant anomalies. To overcome this limitation synthetic anomalies are generated following prior work presented in Chen et al. [58] and Datta et al. [48], by randomly perturbing two or three of the entities in a record. It is important to note that we are not evaluating MEAD but only the RSM. Since RSM is semi-supervised, we require labels for these anomalies as to whether they are relevant (*True*) or not (*False*) in order to perform evaluation. Since relevant instances are generally fewer, we design an imbalanced testing set of approximately 4000 samples for each of the datasets, containing a ratio of 1 : 3 for positive and negative samples. Generated anomalies labelled $True$ contain underlying patterns or similarities, which can be utilized in a sequential model update scenario, similar to the work presented in Lamba and Akoglu. [37]

The synthetic anomaly generation process utilizes the graph view of data, and groups or micro-clusters of records that share instances of unexpected co-occurrences. The anomalous records which are labelled *True* contain partial similarities, in terms of context or entities that belong to micro-clusters of entities. We ensure however these similarities are non-trivial to ensure a fair evaluation of RSM. We include checks to exclude records with rarely occurring key entities (*Shippers* and Consignees), since they are not informative of expected patterns which is standard practice in evaluating anomaly detection approaches. We obtain the combined set of anomalies from the anomaly detection model using a threshold or considering the top-k records.

Evaluation Steps. The online update model works on the output of the anomaly detection system. For evaluation purposes this is the set of generated synthetic anomalies. The weights of RSM are initialized following Algorithm 1. With each subsequent iteration of feedback more records are labelled (annotated by the use), and they are used to retrain (update) the RSM. The *precision@b* is calculated after each update and b records are labelled at each step. b is chosen to be small (25), since human labelling bandwidth is limited. For each dataset we perform multiple (10) runs, where the relative order of records are chosen randomly at start to emulate multiple output scenarios from the underlying anomaly detection model.

Competing Baselines. There are no readily available baselines that are applicable to our scenario. Thus, following the work presented in [30], we compare our model against a set of classifiers as baselines. This is an appropriate comparison since we utilize the same paradigm as iterative classification. The set of classifiers we use as competing baselines in an iterative classification scenario includes a Random Forest classifier, Logistic Regression, a K-Nearest Neighbor classifier, and a linear Support Vector Classifier. For a fair comparison of RSM against baseline methods, we represent the records as the concatenation of the embedding vectors that are used in RSM, rather than using one-hot encoded vectors to represent entities for feature inputs to baseline classifiers. The embeddings provide a more informative input feature and are less sparse than one-hot encoded vectors. We observe that RSM outperforms baselines significantly and it quickly identifies the set of relevant ($True$) anomalies using the feedback. This is shown in Figure 6. RSM is able to learn the patterns of relevant anomalies quickly. The precision (at the top) decreases as more of the relevant anomalies are discovered—that is, the recall increases and drops to a low value when no further positive samples remain. The number of iterations is limited by the count of positively labelled samples in the testing sets.

Effect of Feedback Size on Record Scoring Model. As an iterative update algorithm, the number of items labelled at each step is an important factor in the performance of the scoring model. Different labelling budgets can be present in different application scenarios, which can lead to variance in our parameter of interest: precision at the top ($precision@b$). Thus we perform experimental evaluation to observe the model performance with respect to this parameter. We perform multiple runs (10) with random

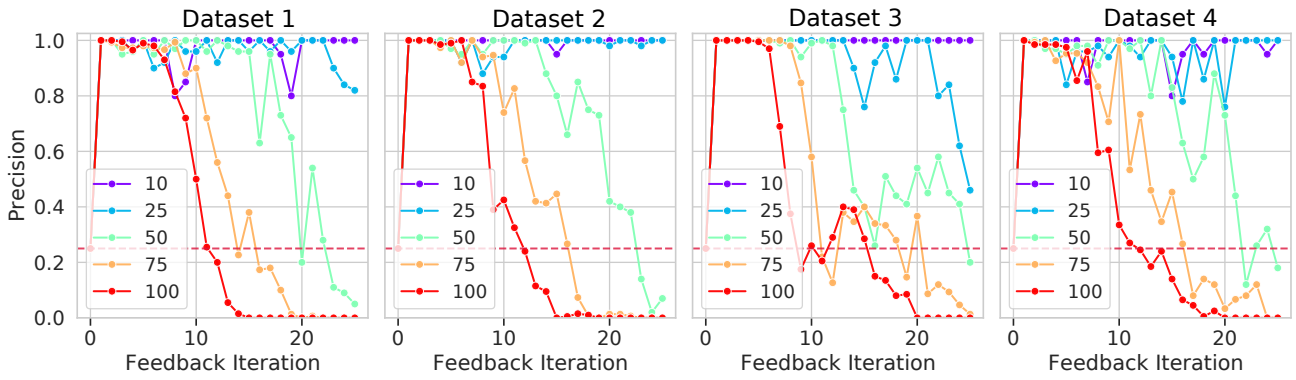

Figure 7. The effect of feedback batch size on model performance. We consider the first 25 batches, due to limitation on number of samples. The metric used is $precision@b$. Here b is chosen as the feedback size. We observe that the model performs well for varying feedback sizes.

Iteration 1

PanjivaRecordID	Carrier	ConsigneePanjivaID	HSCode	PortOfLading	PortOfUnlading	ShipmentDestination	ShipmentOrigin	ShipperPanjivaID	Label (Relevant)
3155365412	C-ChaRev	CPID-AspWra	440793	POL-CarCle	POU-FucVar	SD-JetGod	SO-AspEch	SPID-NavUni	False
7732863662	C-JetYan	CPID-SilDer	940161	POL-ClaBur	POU-AvoMcb	SD-SieBer	SO-AshMin	SPID-CamCab	False
5774331259	C-BeiUnt	CPID-ClaLia	441820	POL-PinFus	POU-KhaWoo	SD-RusSpr	SO-AspEch	SPID-HeaBeh	False
7240517823	C-SalSan	CPID-GolGee	940161	POL-MarWhi	POU-CriAba	SD-MinDep	SO-AspEch	SPID-ChaVan	False
1295329695	C-RedSun	CPID-RusMud	441900	POL-GreCat	POU-AspZei	SD-BonMan	SO-AspEch	SPID-AshCac	True
2918176595	C-PinClo	CPID-BonUne	930200	POL-GreCut	POU-JetPar	SD-EboVel	SO-BlaEar	SPID-AmbJoc	False
3277466340	C-BroWir	CPID-AshCyt	940161	POL-CanVic	POU-CoaSno	SD-CorMis	SO-AspEch	SPID-ChaYap	False
9443138766	C-VioSiv	CPID-BraKar	940360	POL-IvoMan	POU-CorDea	SD-AmbBel	SO-CarMar	SPID-BroSca	False
6318664768	C-AubDev	CPID-CorLiq	940360	POL-CorLoo	POU-NavLud	SD-VioBox	SO-AspEch	SPID-LimCen	False
4038752068	C-BeiUnt	CPID-CorSco	920992	POL-CrePat	POU-FucVar	SD-YelBol	SO-CorOct	SPID-IvoAng	False

Iteration 2 (Post Feedback)

	PanjivaRecordID	Carrier	ConsigneePanjivaID	HSCode	PortOfLading	PortOfUnlading	ShipmentDestination	ShipmentOrigin	ShipperPanjivaID	Label (Relevant)
	1857765465	C-RedSun	CPID-RusMud	821192	POL-CyaBob	POU-AspZei	SD-BonMan	SO-VioGov	SPID-AshCac	True
1	4983549840	C-RedSun	CPID-RusMud	821110	POL-RedPor	POU-AspZei	SD-MagInt	SO-AspEch	SPID-AshCac	True
	4019591314	C-RedSun	CPID-BelLus	940360	POL-ClaBur	POU-AspZei	SD-BroMet	SO-ChaRol	SPID-TanShr	False
2	4622729728	C-RusAly	CPID-RusMud	441900	POL-CyaBob	POU-AspZei	SD-ChaSin	SO-AspEch	SPID-AshCac	True
	1323461951	C-BlaCop	CPID-GraGal	441875	POL-ClaBur	POU-SilDem	SD-ChoSax	SO-AspEch	SPID-AshCac	True
	1378999876	C-CanCer	CPID-RusMud	821193	POL-AmbWal	POU-AspZei	SD-DenUnt	SO-AspEch	SPID-AshCac	True
3	7510368444	C-AzuTri	CPID-FucPla	940179	POL-BluDra	POU-MagFoo	SD-BonMan	SO-CitCou	SPID-TomCab	True
	5549153683	C-KhaJoh	CPID-CreDul	441820	POL-CyaBob	POU-CorBer	SD-GolCum	SO-AspEch	SPID-AshCac	True
	8786412467	C-IvoNeg	CPID-RusMud	940171	POL-WhiWoo	POU-ChaGer	SD-CamMin	SO-AspEch	SPID-AshCac	True
	1686130451	C-CoaYar	CPID-WhiRes	440290	POL-CyaBob	POU-CitExe	SD-VioBox	SO-AspEch	SPID-AshCac	True

Figure 8. Anomaly detection feedback demonstration. The left image shows the first iteration of the system, while the one on the right shows the records in the next update. The entities part of unexpected co-occurrence pairs are highlighted. This demonstrates how a few records along with their underlying *explanations* can help improve the precision in the next step through feedback.

initialization(record ordering) for each feedback size, and report the median value of $precision@b$, where b is the the feedback size. The values of b are chosen from $\{10,25,50,75,100\}$. The results for the first 25 batches are shown, since precision in the earlier batches is of greater interest and also we have a limited number of batches with higher values of b given the fixed size of test set. It is observed that the model performs as expected in different scenarios. Even for small feedback size, the model is able to identify similar records quickly. As the recall increases, the precision value drops gradually till all the positive records are discovered—which is the expected behavior for such a model. Thus the Record Scoring Model is shown to perform well in different settings for feedback size.

Use Case Demonstration

We consider a case where a single record (denoted as $\mathcal{R}_1$) is relevant and thus labelled *True* in the first iteration, highlighted in Figure 8. The user observes that the set of entity pairs {Shipper: SPID-AshCac, HSCode: 441900} and {Shipper: SPID-AshCac, ShipmentDestination: SD-BonMan} are interesting, and wants to have similar records ranked higher in next iterations. Once the feedback is submitted, model weights of the RSM are updated and the updated scores of the remaining records are calculated. Of the yet unlabelled records, the highest scored top 10 records sorted by the updated scores are presented to the user, denoted as *Iteration 2* in Figure 8.

In these updated results, three instances (marked as *1*, *2* and *3*) have easily interpretable association with the prior input. Looking at *1*, we see the records are similar since they share the same set of consignee and shipper. For the record marked as *2*, the shipper, SPID-AshCac, is present in the record marked *True* in the previous iteration—although the consignee is different. For the record marked as *3*, the records share same shipment destination. Further, the top-10 of the updated result contains records that share the consignee (C-RedSun) as well as other entities such as the Port of Unlading marked in the positive input.

This demonstrates how RSM is able to capture similarities between records that have been marked as *True* (relevant) in previous feedback iterations by the end user, following a *more like this* strategy as discussed in Ghani et al.[30] Thus RSM is able to identify records that have similar patterns based on user input and improving precision at the top.

Interactive Visual Dashboard

The interactive interface binds together the different components of the TimberSleuth (Figure 1). This dashboard is designed to fulfill requirements **R2** and **R3** by allowing the user to interact with the data and backend output. Specifically the dashboard is designed with the following criteria:

(i) Efficiently display tabular data with a large number of attributes and entities;

(ii) Support exploratory analysis through visual encoding of entities, aggregated values, and relationships to enable visual comparison; and

(iii) Responsive, scalable and low-latency user interface.

Design Rationale

We designed the TimberSleuth dashboard to consist of multiple pages—with a tabular data view page and a detailed visualization page. Our users indicated a preference for this design rather than a more traditional single-page visualization dashboard. This also meant that we did not link the views in different pages; in other words, our dashboard is **not** based on the *coordinated multiple views* (CMV)[59] paradigm. However the user can navigate between them quickly using a persistent navigation bar at the top of the page.

The objective of the record detail page is to display the details for a selected record, explain the anomaly, provide visualization components for analysis, and finally obtain feedback. In determining whether a record is suspicious and relevant, the end user needs to understand the underlying relationships among entities, which would allow them to connect the dots. The process entails the following steps: (i) The user marshals their implicit domain knowledge; (ii) The user interprets the information presented in the interface based on their domain knowledge; and (iii) The user bridges the gap between information presented and assimilated, and arrives at a conclusion.

The components are designed with inputs from the domain experts who are aware of the challenges encountered in deciphering shipment records, with automated aggregation and directed visualizations suitable for the task.

Specifically some of the requirements for users, as explained by our collaborating domain experts, to analyze a given record are as follows: (i) Decomposing the relationships between the entities of supply chain; (ii) Need to compare entities with respect to other entities, of the same or different type (domain); (iii) Simplifying comparison given the high dimensionality (cardinality) of the attributes; and (iv) Understanding how entities such as companies are interconnected to learn relationships that are conspicuous

The visual components are thus designed to aid the end-user in understanding whether a trade instance is suspicious and then provide feedback to improve underlying model. We utilize the three basic design paradigms as discussed in Javed and Elmqvist[60] and Gleicher[61] for visual design—*superposition*, *juxtaposition*, and *explicit visual encoding* of relationships. In our case, while we allow the user to perform queries in terms of the entities of interest, the exploration is not query driven such as in the work of Vartak el al.[62]

Visualization Layout

The dashboard consists of two main views. The main result (landing) page displays the ranked list of records in a table as shown in Figure 10. The records are sorted in descending order of anomaly scores, so that the user sees the most anomalous records first. The user can choose the time period for which the shipment records are of interest. Given that we have over fifty attributes in the raw data and limited horizontal space, only the key attributes are displayed in this compact tabular view. A clickable button allows the user to expand each row, where further (but not all) attributes are revealed. This expansion view also allows the user to navigate to the *record detail page*. The table is paginated to enable a large number of records to be displayed, and also support sorting (by score) and implements search functionality. The user looks at each record in this table, and proceeds to the record detail page to investigate a record further.

Explaining Anomalies

It is imperative that the end users can trust the output of the anomaly detection algorithm as discussed in Lipton et al.[63] and bridge the cognitive gap to their implicit domain knowledge. Here visual analytics can be a powerful tool.[64,65] Most other anomaly detection models including MEAD are opaque (*blackbox* models)[63] since they do not provide the end user any understanding why a record is judged anomalous. Post-hoc model interpretability has been explored in approaches such as *LIME*[66], *SHAP*[67], and *InterpretML*[68] which have also been applied to anomaly detection, similar to the work presented by Das et al.[69]. However such approaches are not directly applicable to our scenario due to the strictly categorical attributes with high cardinality of our data. Explaining a prediction can involve the following interrogative questions: *Why?*, *Who?*, *What?*, *How?*, *When?* and *Where?*[65]

Method. For potentially suspicious timber trades the elements of interest are *Who* and *Why*. Anomalies in multivariate categorical data are unexpected patterns of entity co-occurrences[48], and our goal is to present the user with possible unexpected co-occurrences which can explain why a record is deemed anomalous, and help in answering the question *Who*. It is also important to present a user with relative importance among the candidates, to aid in the decision process to answer the question *Why*. It thus allows the user to construct the correct explanation from the set of candidates.

We accomplish this through presenting the user with a ranked list of entity pairs, where the ranking is based on decreasing order of *discordance* among the entities in the pair. So the visual representation of the explanation is a sorted list of entity pairs in a tabular format, that is part of the record detail view.

While there is no direct measure of such a quantity—*discordance*, using vector representations of entities with a suitable distance metric such as cosine distance allows us to define how dissimilar, or far apart, a pair of entities are in the latent embedding space. This can be a good measure of their *discordance* and provides a quantifiable answer to the question: *Why?*

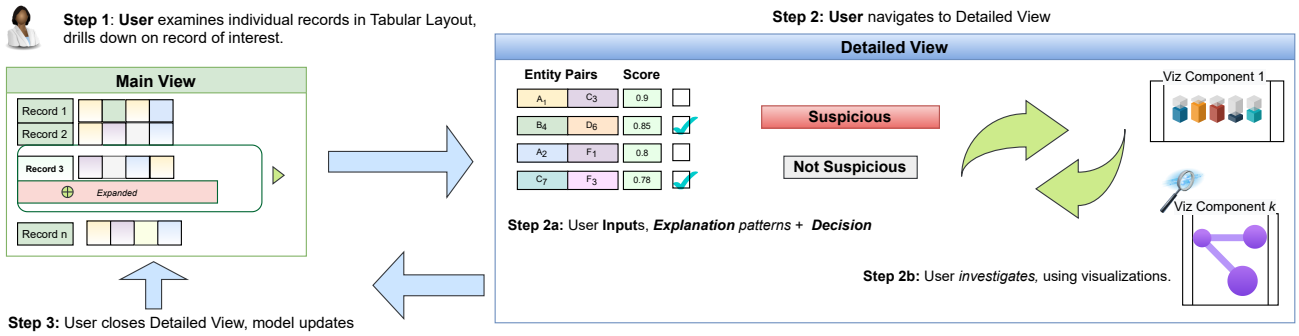

Figure 9. Operational steps in TIMBERSLEUTH. The user starts with the main page, with its tabular view of records. Once they decide to investigate a particular record, they navigate to the detailed view page. In the detailed view page, they utilize the visual analytics components to gain insights and provide input and decide whether the record is relevant. They close the window and the model updates.

Table 2. Explainability. Evaluation of explainability using pairwise entity distance for the synthetic anomaly sets.

PRECISION/RECALL	DATASET 1	DATASET 2	DATASET 3	DATASET 4
@top-3	0.74 / 0.65	0.74 / 0.74	0.66 / 0.76	0.68 / 0.61
@top-4	0.70 / 0.74	0.72 / 0.81	0.64 / 0.80	0.71 / 0.76
@top-5	0.67 / 0.80	0.70 / 0.84	0.62 / 0.82	0.72 / 0.86
@top-6	0.67 / 0.86	0.70 / 0.87	0.62 / 0.84	0.72 / 0.94
@top-7	0.68/0.92	0.70/0.91	0.63/0.87	0.71 / 0.98
@top-8	0.69 / 0.96	0.71 / 0.94	0.64 / 0.92	0.71 / 0.99
@top-9	0.69/0.98	0.71/0.96	0.65/0.97	0.70 / 0.99
@top-10	0.70 / 0.99	0.71 / 0.98	0.65 / 0.99	0.69 / 0.99

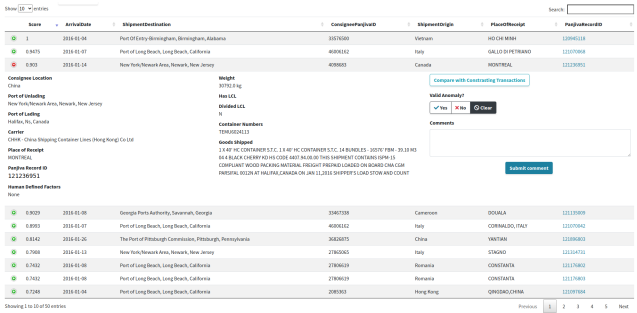

Figure 10. Landing Page with tabular view of records sorted by anomaly score. The clickable button allows the user to access details and navigation to the detailed view page.

For each pair of entities (belonging to different domains), their *discordance* is calculated as the cosine distance of their vector representations, i.e., the outputs of the underlying anomaly detection model (MEAD). Since the anomaly score from multiple models are combined using an ensemble of model instances of MEAD as previously discussed, we calculate the mean discordance score across models for any pair of entities.

The rationale behind presenting ranked pairs of entities is that binary relationship between entities is the most atomic form of interaction among the entities of the record. This is an intuitive approach to explaining anomalies, and ranking those possible explanations based on how *plausible* that entity pair is cause of the record being anomalous and also if that entity co-occurrence is of interest to the end user. There are no clear alternative design choices that would allow both explainability at this granular level and simultaneously enable user feedback.

It is also important to note that that the end users currently operate without any automated tools in a mostly manually driven process. Our automated system leverages their implicit knowledge and their expertise. It does not indeed force the end user to commit to memory all facts about the different entities and how they are related, on the contrary aids exploration, investigation and recall through the visualization components.

Evaluation. To evaluate how effective our approach is at capturing the cause for anomalies, we use the synthetic anomaly datasets where multiple entities are perturbed in a nominal (normal) record. The metrics used are adapted from information retrieval— *precision* and *recall*, since we have a ranked list. An explanation item in the list is considered to be a correct if it contains at least one of the perturbed entities, which is used to calculate precision at top-k. For recall, we check how perturbed entities are discovered at k^{th} position in the list. The evaluation results shown in Table 2 demonstrate that this approach effectively detects the entity-pair due to which a record was judged anomalous.

In considering subsets of the entire record, we reveal multiple sub-spaces in which the potential cause of the anomaly potentially exists. Embeddings can be obtained from using the underlying network structure of the entities, or embeddings calculated as output of the anomaly detection system. We choose the latter noting this does not violate the blackbox assumption of the anomaly detection model since we use the output rather than modify the model. The entity pairs that are deemed a relevant explanation for why a

record is judged suspicious can be highlighted by the end user through the interface, using a simple checkbox. The visualization components described in the next section are intended to help the user in providing this input. The RSM incorporates this feedback to provide better output.

Visualization for Investigating Records

Visualization components in TimberSleuth are designed with the objective of aiding in analysis of the records that are highlighted as relevant anomalies by the RSM. The analysis is intended to aid the user to analyze relationships between entities within the record and also explore entities in context of other entities in the supply chain.

The views presented here are all integrated within the main TimberSleuth visual dashboard and are accessed by selecting different views in the top navigation bar (Figure 1).

Comparison of Similar Records

The process of visual anomaly detection requires effectively conveying the contextual information for a trade record. The design rationalize of this visualization component is based on the fact that users need to compare a potentially suspicious record with similar records, and doing so through a simple tabular view of a set of records is often difficult. Since records are sets of entities, visualizing multiple such records simultaneously to enable comparison is not a trivial task. This task is conceptualized as a process of superposing multiple "images"—which are entities of the same domain across multiple records—stacked vertically to reveal clusters and deviations. We adopt a simple approach which tries to mimic a process of finding the odd one out of set of pictures by superposing them—as if using a translucent projection.

We refer to this as *stacked comparison* shown in Figure 11. Entities are represented as text or numerical identifier, they need to be visually represented in the same space simultaneously for superposition. This can be done by using vector representation or embedding of entities in low dimensional (2-D) space, allowing for simultaneous representation of entities from a domain across multiple records to be compared.

The Heterogeneous Information Network schema of the data is used to obtain the embeddings of the entities. We consider the HIN schema along with the metapaths described in Table 1 and apply *Metapath2vec* to obtain embeddings for the nodes (entities). Dong et al.[54] demonstrated *Metapath2vec* as an effective approach to obtain node embeddings that capture semantic similarities between nodes of a HIN, using metapaths and the structure of the network. The embedding vector obtained is transformed to a 2-dimensional vector using t-SNE which preserves the relative proximities between the points representing the entities. It should be noted that alternative proximity preserving dimensionality reduction approaches such as UMAP[70] are also applicable.

While glyphs have been used in pictorial representation they are not suitable for this scenario. Specifically for the record being examined, similar records (up to a specified count) are considered for comparison. Records are deemed similar if they share same set of Consignee and/or Shipper, along with other key attributes determined by domain experts. Kernel Density Estimation is performed on the points representing entities, which highlights the higher density regions in this space. This can reveal clusters of points and whether the current record's entity is dissimilar to entities that should usually occur in this context.

Visualizing Record Entities

The comparative analysis of the entities in a particular record is important in the task of determining the possible cause of a record being anomalous. Providing a visual summary[61] of a record is non-trivial since the entities belong to different domains. For instance how does one how compare a Port such as *Los Angeles* to a Carrier such as *Mersk*? While pairwise proximities help examine each pair of entities with all possible combinations, it does not provide a comprehensive representation with juxtaposition.

We use the HIN (graph) schema as shown in Figure 4, and the relationships between entities described by metapaths from Table 1. Applying *Metapath2Vec*[54], we obtain the node embeddings which represents entities in the same low-dimensional space. We transform the obtained embeddings to 2-dimensional space using t-SNE.

Scatterplots are a simple and efficient tool for visualizing data[71], and use of t-SNE[72] provides a two-dimensional representation of the embeddings that can be visualized using a scatterplot. The entities of a record are represented in the same space using an interactive scatter plot that allows for effective visual comparison, as shown in Figure 12. It is important to note that the scatter plot captures the approximate proximity among entities, which is difficult to measure through manual unassisted investigation. The axes, while they do not have a physical interpretation, they show the relative distances in the latent space which can be interpreted as relative co-occurrence likelihoods. The buttons on the top of the plot enables the user to toggle entity selection, thus allowing for comparing relative proximities or co-occurrence likelihood among different entity subsets. Embeddings have been used in other domains such as NLP to visualize relative proximity among items, and has been shown to be a useful visualization tool.

Visualizing Flows

Determining whether a trade record pertains to illegal activity requires understanding the supply chain of commodities and entities involved in a record, which can be done by examining relationships centred on type of goods transported (HS Code). It is difficult to navigate the data space if multiple dimensions are visualized simultaneously without proper structure due to large number of entities. Thus efficiently managing the dimensions through ordering, spacing and organization is important.

Sankey Diagrams[73], which are related to parallel coordinate sets[74], allow users to interactively explore complex flow scenarios involving entities pertaining to multiple attributes simultaneously. The flows quantify the degree of interaction (in terms of aggregated counts) between entities of the supply chain. This helps in conveying overall context and shows relationships between multiple entities. It also allows for exploration in a scenario where multiple many-to-many relationships exist among

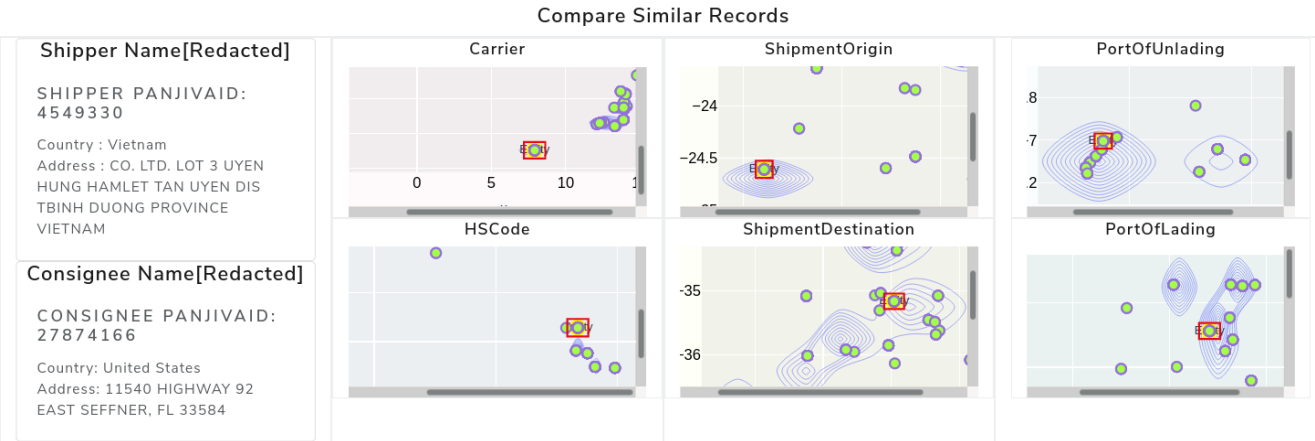

Figure 11. Stacked comparison of trade records. The shipper and consignee have been chosen as anchor points. The entities for the current record are highlighted using a red box. Clusters show if an entity belongs to low density region; for instance, the *Carrier* of this record stands out with respect to similar records.

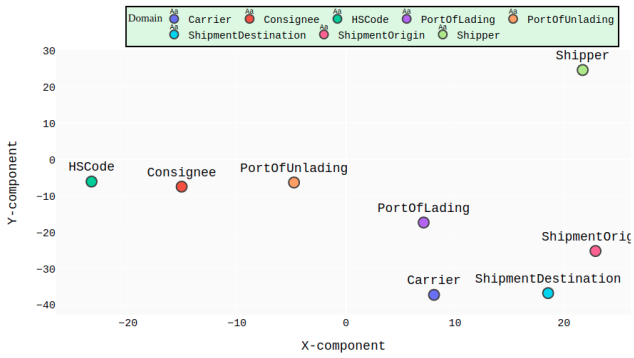

Figure 12. Entities in an anomalous trade record. Buttons toggle entity selection, allowing comparison among user selected entities only. Here the *Shipper* is dissimilar to the *Shipment Origin* and *Destination* entities, which can indicate that the *Shipper* is a possible cause of the anomaly.

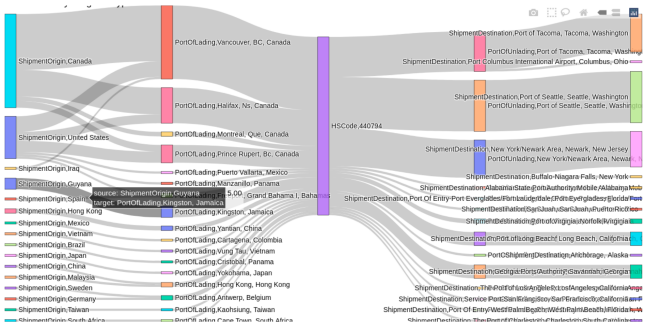

Figure 13. Entity flows as a Sankey diagram. The particular type (SF2) highlights the ports and the companies that transports items with the given HS Code for the record being investigated.

attributes, facilitating comparison through explicit encoding and juxtaposition. Thus we design two types of Sankey diagrams, denoted as *SF1* and *SF2* which shows relationships with respect to the HS Code found in a particular record.

SF1. In this case the set of domains considered are *Consignee*, *HS Code* and *Shipper*. This tells the user what are the major consignees and shippers trading in the goods described by the HS Code. This can reveal patterns such as a particular set of companies previously flagged are engaged in trading commodities which has been known to contain illegal timber which can be further investigated. Broadly it presents an aggregated representation of consignee and shippers in terms of their commodity characteristic.

SF2. This Sankey diagram visualizes relationships between *Shipment Origin*, *Port Of Lading*, *HS Code*, *Port Of Unlading* and *Shipment Destination* respectively. The geographical entities such as ports and origin or destination can potentially reveal crucial information regarding the commodity supply chains, which is essential in the decision making process. With respect to the HS Code, the user is presented with an aggregated view of trade flows through such entities that allows for assimilation of the context in which the item is traded. We show an instance of this in Figure 13.

Visualizing Shipper-Consignee Interaction

While visualizing the flows presents an aggregation of activity of the entities with respect to the HS Code (product type), it does not reveal which companies have trading relationships. Examining these relationships between shippers and the consignees is crucial to understanding if a trade instance is suspicious since the companies are the key actors in trading process and supply chains. The need for such analysis arises due to trading practices such as mislabelling and shell companies that are prevalent in illegal timber trade.[75]

Knowledge of which companies have trading relationships can reveal anomalous or interesting clusters, along with insights pertaining to overall trading patterns in the context of the shipper and consignee of the shipment record being investigated. This is important in terms of exploration as some inter-company relationships might not be readily recognized or known to the user. Moreover, there can be latent or indirect relationships which could potentially reveal links between entities of interest.

Data Modelling and Proximity Estimation. The possibility of relationships existing can be quantified using relative

probability of a link existing or relative proximity between two nodes, which can be calculated using an appropriate distance metric on vector representation of nodes. To discover such possible links, we build a bipartite network $\mathcal{B}_G$ with the *Shippers* and *Consignees* as the nodes types. In $\mathcal{B}_G$ edges signify that the training set contains interaction between respective *Shipper* and *Consignee*.

Graph Neural Networks such as GraphSage[76] have been demonstrated to be effective in capturing and aggregating both structural and attribute information in graphs. Since we do not have implicit node features for this bipartite network, we initialize them with node embeddings obtained as follows. We consider the Heterogeneous Information Network (HIN) view of data, consisting of all node types i.e all domains and not only the company domains (shippers and consignee) and the metapaths in Table 1 that captures semantic relationships between all the different types of entities. Metapath2vec[54] is used to compute the initial node embeddings from this HIN. This initialization helps bring in latent information from other node types which are not part of the bipartite graph $\mathcal{B}_G$.

Next, we design a two layered Graph Neural Network for the bipartite graph $\mathcal{B}_G$, and compute the embeddings of the nodes in $\mathcal{B}_G$—from the initial node features and the structural features. These embeddings serve in the task of capturing relative proximity of these actors (Consignee and Shippers). The triplet ranking loss function used is shown in Equation 2, which is optimized to reduce distance between neighboring nodes. Negative samples are selected randomly from the set of non-neighboring nodes. Here $\mathcal{D}$ is a suitable distance metric, such as cosine distance, and m is the margin value, a hyperparameter set by user.

$$\begin{cases} h_v^1 = Mean(h_u^0) \forall u \in Nbr(v) \\ h_v^2 = W^T((Mean(h_u^1)) \oplus h_v^1) \forall u \in Nbr(v) \\ Loss(X, X_p, X_n) = max(\mathcal{D}(X, X_p) - \mathcal{D}(X, X_n) + m, 0) \end{cases} \tag{2}$$

Visualization Details. The network (subgraph) with the actual and projected neighbors of the consignee and shipper of the record being investigated is visualized, as shown in Figure 14. The actual and projected links are differentiated through use of different colors and labels that are displayed on hover. Also the shipper and consignee are displayed with different colors. Further, node sizes reflect how prevalent a company is in terms of total trades, using a logarithmic scale. The interactive visualization allows the user to zoom in, click on nodes to reveal details such as company names and reposition the nodes in space to customize the representation. Juxtaposing companies in the same space facilitates a more intuitive understanding in the investigative process.

Implementation Notes

The overall platform is implemented in Python 3.0 using PyTorch, Deep Graph Library, and scikit-learn for machine learning, and Django, Plotly, and PyVis for the frontend. TimberSleuth's main data store uses SQLite for ease of use. We choose mature libraries for optimized performance and maintainability. To improve latency and make the experience responsive, we incorporate multiple caches as part of the design.

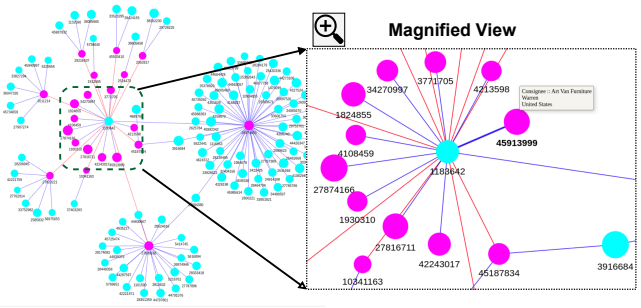

Figure 14. Network of trading companies. Actual links are represented in cyan, whereas projected links are magenta. Company information is displayed when clicking a node, while the label shown is their ID to allow for quick referencing.

We utilize *Redis*, an in-memory cache, so that intermediate computations are faster by preemptively populating frequently used data at system start up. We further use FAISS[77] to index and fetch nearest neighbors for vector based operations. We also perform extensive pre-computation for intermediate results, and these results are cached for efficiency.

Application Example

Finally, we validate the TimberSleuth system by showcasing how a user can utilize it to analyze shipping data. The TimberSleuth user interface has multiple components that work together to provide the desired functionality. The system begins with the main landing page, as shown in Figure 10. The results are shown in a tabular format, and pertain to a fixed period of data—which is chosen as a month of data and the preceding (six) months as the training data for the anomaly detection model. A certain number of prior months, usually six months of data is taken as the *training* set—from which patterns are extracted and upon which the underlying machine learning models are trained. Note that this entails both the anomaly detection model as well as the visualizations that rely on representation learning.

The main landing page with tabular view presents a comprehensive view of top-most anomalous records, and it allows the user the starting point for navigating and investigating the records. The results in this view are updated when the underlying learning model—Record Scoring Model—is updated.

Starting at the Main Page. At the start, before any user feedback, the main page displays the records sorted by their anomaly scores—where the most anomalous records are at the top. The user can view the basic details of the record in this overview. They can expand upon the details further but clicking on the *expand* button as shown in Figure 10. This expanded view, contained as part of the tabular allows them to see more of the record attributes.

While it is understood that the end-user may have domain expertise to discern the relevance of a record, it is not without having detailed insights and associations. It was understood through communication with collaborating domain experts that a user can unsure of the correct choice and requires more insights, given users have different areas of expertise in terms of geographical locations and trade patterns pertaining certain types of products (timber). This

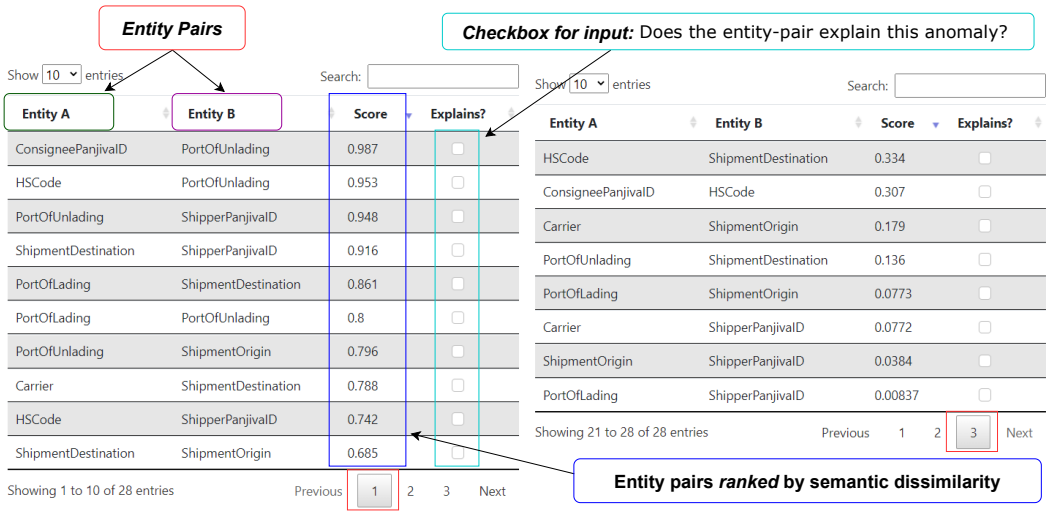

Figure 15. Visual component for user feedback. The user can select entity pairs that are deemed to be the cause of anomaly. The ranking of entity pairs provides the user with the semantic dissimilarity or discordance that is learnt from historical training data.

is where the visualization components are useful. The user is provided with a *hyperlink* that allows them to navigate to the *record detail view* and delve into the record.

Feedback Through Analysis. The record detail page has multiple parts to it. The first part is the set of attributes that are not shown in the main page's tabular view to avoid clutter. This helps the user get a more complete picture, and informs of the exact details of the record regarding the supply chain entities as well as company specific details.

The user provides feedback as follows (Figure 15): (i) Individual entities of interest—a Shipper or Consignee, that is of interest based on some previously acquired or implicit user knowledge. (ii) Entity pairs that appear to the user as out of place. For instance a particular port might not be expected to ship a particular product type, based on historical patterns or knowledge.

The system presents tabular view of entity pairs with pagination, ranked by their semantic dissimilarities or *discordance*. Higher ranked entities are expected to be more probable cause of the record being anomalous. For each entity pair, the user can provide a boolean input—marking the entity pair as relevant cause of the anomaly. Not all the entity pairs need to selected, only the one those are relevant. The remaining unmarked one are considered as not valid causes by default.

It is important to note that the Record Scoring Model operates in a way such it scores unannotated anomalous records higher that contain entity pairs that been marked by the user—as well as semantically similar entity pairs. So it learns the patterns from the user feedback, that goes beyond exact matches. The user can also indicate if the *Shipper* or *Consignee* is of interest, so that the Record Scoring Model would rank records containing them higher in future iterations.

The process to decide whether the record is indeed anomalous and relevant, it requires the user to have a detailed understanding of the relationships between the entities constituting the record. The visualization components described earlier are crucial towards this. The user uses these multiple views of data, to ascribe the reason to why the record is of interest and why one or more relationships are relevant towards causing the anomaly. These visualization components play a key part in the decision making process since domain specific knowledge cannot be explicitly encoded into the machine intelligence.

After the user performs the analysis, they arrive at a decision. They provide feedback by selecting the entity pairs—in case the anomaly is relevant. Then, they mark the record as *Suspicious* i.e. relevant. If the anomaly is not relevant, the user marks it *Not Suspicious*. The responses are recorded and integrated into the system, and the user can exit this page by closing the browser window.

Getting Updated Results. After the user has provided feedback, the next step is updating the model. After closing the record detail page, the user navigates to the main page. The number of feedback instances can be varied depending on the annotation budget, and is scenario dependant. The user selects the *Update Model* button in the main page with tabular view. This prompts the Record Scoring Model to be re-trained based on the received feedback, and it presents a new ranked list of records based on the patterns learnt from feedback. The user can again perform the same set of iterative steps as described above, and provide more feedback to aid in further improvement in relevance of detected anomalies.

Conclusion and Future Work

Enforcement agencies have been making a concerted effort towards algorithmically targeting potentially illegal timber shipments that violate trade, tax, and ecological regulations. However, an integrated targeted system for timber shipments is absent, which TimberSleuth attempts to solve. There are multiple challenges in designing such a system towards which we propose well-defined solutions.

Due to data confidentiality regulations from enforcement agencies in the U.S. government as well as proprietary data, we were unable to obtain a formal user study and perform a detailed validation of the visualization components of the system. However, the next planned steps include knowledge

transfer to target users and we have incorporated elements in our design from informal discussions. Another important research question relates to the usability characteristics for in-the-wild operation of our system with respect to end users. This requires the direct involvement of enforcement agencies in conducting user studies, which has its own set of challenges related to the sensitivity of data and on-going investigations.

Acknowledgements

We acknowledge non-profit organizations and U.S. government agencies who worked with us on this project.

References

1. Channing Mavrellis. Transnational crime and the developing world. https://gfintegrity.org/report/transnational-crime-and-the-developing-world/, 2017.
2. Maya Forstater. Illicit financial flows, trade misinvoicing, and multinational tax avoidance: the same or different. *CGD policy paper*, 123(29), 2018.
3. P. Pacheco, Mo, N. K., Dudley, A. Shapiro, N. Aguilar-Amuchastegui, P.Y. Ling, C. Anderson, and A. Marx. Deforestation fronts: Drivers and responses in a changing world. https://bit.ly/3DlTfzZ, 2021.
4. Sam Lawson and Larry MacFaul. Illegal logging and related trade, 2010. URL https://bit.ly/logIllegalLSLM.
5. REA Almond, M Grooten, and T Peterson. *Living Planet Report 2020-Bending the curve of biodiversity loss*. World Wildlife Fund, 2020. URL https://bit.ly/wwfReportLPR.
6. Janine E Robinson and Pablo Sinovas. Challenges of analyzing the global trade in CITES-listed wildlife. *Conservation Biology*, 32(5):1203–1206, 2018. doi: 10.1111/cobi.13095.
7. Christopher Nelson. Machine learning for detection of trade in strategic goods: An approach to support future customs enforcement and outreach. https://worldcustomsjournal.org/archive/volume-14-number-2-september-2020/, 09 2020.
8. Alex Endert, William Ribarsky, Cagatay Turkay, BL William Wong, Ian Nabney, I Díaz Blanco, and Fabrice Rossi. The state of the art in integrating machine learning into visual analytics. In *Computer Graphics Forum*, volume 36, pages 458–486. Wiley Online Library, 2017. doi: https://dl.acm.org/doi/abs/10.1111/cgf.13210.
9. Liu Jiang, Shixia Liu, and Changjian Chen. Recent research advances on interactive machine learning. *Journal of Visualization*, 22(2):401–417, 2019.
10. D. Thom, H. Bosch, S. Koch, M. Wörner, and T. Ertl. Spatiotemporal anomaly detection through visual analysis of geolocated Twitter messages. In *Proceedings of the IEEE Pacific Visualization Symposium*, pages 41–48, 2012. doi: 10.1109/PacificVis.2012.6183572.
11. N. Cao, C. Shi, S. Lin, J. Lu, Y. Lin, and C. Lin. Targetvue: Visual analysis of anomalous user behaviors in online communication systems. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):280–289, 2016. doi: 10.1109/TVCG.2015.2467196.
12. N. Cao, C. Lin, Q. Zhu, Y. Lin, X. Teng, and X. Wen. Voila: Visual anomaly detection and monitoring with streaming spatiotemporal data. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):23–33, 2018. doi: 10.1109/TVCG.2017.2744419.
13. Nan Cao, Yu-Ru Lin, David Gotz, and Fan Du. Z-glyph: Visualizing outliers in multivariate data. *Information Visualization*, 17(1):22–40, 2018.
14. John R Goodall, Eric D Ragan, Chad A Steed, Joel W Reed, G David Richardson, Kelly MT Huffer, Robert A Bridges, and Jason A Laska. Situ: Identifying and explaining suspicious behavior in networks. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):204–214, 2018. doi: 10.1109/TVCG.2018.2865029.
15. C. Xie, W. Xu, and K. Mueller. A visual analytics framework for the detection of anomalous call stack trees in high performance computing applications. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):215–224, 2019. doi: 10.1109/TVCG.2018.2865026.
16. L. Wilkinson. Visualizing big data outliers through distributed aggregation. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):256–266, 2018. doi: 10.1109/TVCG.2017.2744685.
17. Sungahnn Ko, Shehzad Afzal, Simon Walton, Yang Yang, Junghoon Chae, Abish Malik, Yun Jang, Min Chen, and David Ebert. Analyzing high-dimensional multivariate network links with integrated anomaly detection, highlighting and exploration. In *Proceedings of the IEEE Conference on Visual Analytics Science and Technology*, pages 83–92. IEEE, 2014. doi: 10.1109/VAST.2014.7042484.
18. C. Chen, J. Yuan, Y. Lu, Y. Liu, H. Su, S. Yuan, and S. Liu. Oodanalyzer: Interactive analysis of out-of-distribution samples. *IEEE Transactions on Visualization and Computer Graphics*, pages 1–1, 2020. doi: 10.1109/TVCG.2020.2973258.
19. Charles Perin, Pierre Dragicevic, and Jean-Daniel Fekete. Revisiting bertin matrices: New interactions for crafting tabular visualizations. *IEEE Trans. Vis. Comput. Graph.*, 20(12):2082–2091, 2014. doi: 10.1109/TVCG.2014.2346279. URL https://doi.org/10.1109/TVCG.2014.2346279.
20. Alexander Lex, Nils Gehlenborg, Hendrik Strobelt, Romain Vuillemot, and Hanspeter Pfister. Upset: visualization of intersecting sets. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1983–1992, 2014. doi: 10.1109/TVCG.2014.2346248.
21. Alexander Lex, Hans-Jorg Schulz, Marc Streit, Christian Partl, and Dieter Schmalstieg. Visbricks: multiform visualization of large, inhomogeneous data. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2291–2300, 2011. doi: 10.1109/TVCG.2011.250.
22. Mehmet Adil Yalçın, Niklas Elmqvist, and Benjamin B Bederson. Keshif: Rapid and expressive tabular data exploration for novices. *IEEE Transactions on Visualization and Computer Graphics*, 24(8):2339–2352, 2018. doi: 10.1109/TVCG.2017.2723393.
23. M. Blumenschein, M. Behrisch, S. Schmid, S. Butscher, D. R. Wahl, K. Villinger, B. Renner, H. Reiterer, and D. A. Keim. SMARTexplore: simplifying high-dimensional data analysis through a table-based visual analytics approach. In *Proceedings of the IEEE Conference on Visual Analytics Science and Technology*, pages 36–47, 2018. doi: 10.1109/

VAST.2018.8802486.

24. Katarina Furmanova, Samuel Gratzl, Holger Stitz, Thomas Zichner, Miroslava Jaresova, Alexander Lex, and Marc Streit. Taggle: Combining overview and details in tabular data visualizations. *Information Visualization*, 19(2):114–136, 2020. doi: 10.1177/1473871619878085.
25. Chris North and Ben Shneiderman. Snap-together visualization: A user interface for coordinating visualizations via relational schemata. In *Proceedings of the ACM Conference on Advanced Visual Interfaces*, page 128–135, New York, NY, USA, 2000. ACM. doi: 10.1145/345513.345282.
26. C. Weaver. Building highly-coordinated visualizations in Improvise. In *IEEE Symposium on Information Visualization*, pages 159–166, 2004. doi: 10.1109/INFVIS.2004.12.
27. Dan Pelleg and Andrew Moore. Active learning for anomaly and rare-category detection. In *Proceedings of the International Conference on Neural Information Processing Systems*, pages 1073–1080, Cambridge, MA, USA, 2004. MIT Press.
28. Naoki Abe, Bianca Zadrozny, and John Langford. Outlier detection by active learning. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, New York, NY, USA, 2006. ACM. doi: 10.1145/1150402.1150459.
29. Jingrui He and Jaime Carbonell. Nearest-neighbor-based active learning for rare category detection. In *Proceedings of the International Conference on Neural Information Processing Systems*, pages 633–640, 2007.
30. Rayid Ghani and Mohit Kumar. Interactive learning for efficiently detecting errors in insurance claims. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, page 325–333, New York, NY, USA, 2011. ACM. ISBN 9781450308137. doi: 10.1145/2020408.2020463.
31. Kalyan Veeramachaneni, Ignacio Arnaldo, Vamsi Korrapati, Constantinos Bassias, and Ke Li. AI^2: Training a big data machine to defend. In *Proceedings of the IEEE Conference on Big Data Security on Cloud*, pages 49–54. IEEE, 2016. doi: 10.1109/BigDataSecurity-HPSC-IDS.2016.79.
32. Shubhomoy Das, Weng-Keen Wong, Thomas Dietterich, Alan Fern, and Andrew Emmott. Incorporating expert feedback into active anomaly discovery. In *Proceedings of the IEEE International Conference on Data Mining*, pages 853–858. IEEE, 2016. doi: 10.1109/ICDM.2016.0102.
33. T. Pevný. Loda: Lightweight on-line detector of anomalies. *Machine Learning*, 102:275–304, 2015. doi: 10.1007/s10994-015-5521-0.
34. Shubhomoy Das, Weng-Keen Wong, Alan Fern, Thomas G Dietterich, and Md Amran Siddiqui. Incorporating feedback into tree-based anomaly detection. *arXiv preprint arXiv:1708.09441*, 2017.
35. Md Amran Siddiqui, Alan Fern, Thomas G Dietterich, Ryan Wright, Alec Theriault, and David W Archer. Feedback-guided anomaly discovery via online optimization. In *Proceedings of the ACM Conference on Knowledge Discovery & Data Mining*, pages 2200–2209, 2018. doi: 10.1145/3219819.3220083.
36. Shubhomoy Das and Janardhan Rao Doppa. Glad: Glocalized anomaly detection via active feature space suppression. *arXiv preprint arXiv:1810.01403*, 2018.
37. Hemank Lamba and Leman Akoglu. Learning on-the-job to re-rank anomalies from top-1 feedback. In *Proceedings of the International Conference on Data Mining*, pages 612–620. SIAM, 2019.
38. Luyang Kong, Lifan Chen, Ming Chen, Parminder Bhatia, and Laurent Callot. Improve black-box sequential anomaly detector relevancy with limited user feedback. *ICML Workshop on Human in the Loop Learning*, 2020.
39. Panjiva. Panjiva trade data. https://panjiva.com, 2020.
40. Kaustav Das and Jeff Schneider. Detecting anomalous records in categorical datasets. In *Proceedings of the ACM SIGKD DConference on Knowledge Discovery and Data Mining*, pages 220–229. ACM, 2007. doi: 10.1145/1281192.1281219.
41. A. C. Wiedenhoeft, J. Simeone, A. Smith, et al. Fraud and misrepresentation in retail forest products exceeds US forensic wood science capacity. *PloS one*, 14(7), 2019. doi: 10.1371/journal.pone.0219917.
42. Ben Shneiderman. *Human-Centered AI*. Oxford University Press, 2022.
43. IUCN. The IUCN Red List of Threatened Species. https://www.iucnredlist.org, 2019.
44. Purushottam Kar, Harikrishna Narasimhan, and Prateek Jain. Surrogate functions for maximizing precision at the top. In *International Conference on Machine Learning*, pages 189–198. PMLR, 2015.
45. Yizhou Sun, Jiawei Han, Xifeng Yan, Philip S Yu, and Tianyi Wu. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. *Proceedings of the VLDB Endowment*, 4(11):992–1003, 2011.
46. Bokai Cao, Mia Mao, Siim Viidu, and Philip Yu. Collective fraud detection capturing inter-transaction dependency. In *Proceedings of the KDD Workshop on Anomaly Detection in Finance*, pages 66–75, 2018.
47. B. Cao, M. Mao, S. Viidu, and P. S. Yu. Hitfraud: A broad learning approach for collective fraud detection in heterogeneous information networks. In *Proceedings of the IEEE International Conference on Data Mining*, pages 769–774. IEEE, 2017. doi: 10.1109/ICDM.2017.90.
48. Debanjan Datta, M Raihanul Islam, Nathan Self, Amelia Meadows, John Simeone, Willow Outhwaite, Chen Hin Keong, Amy Smith, Linda Walker, and Naren Ramakrishnan. Detecting suspicious timber trades. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13248–13254, 2020. doi: 10.1609/aaai.v34i08.7032.
49. Leman Akoglu, Hanghang Tong, Jilles Vreeken, and Christos Faloutsos. Fast and reliable anomaly detection in categorical data. In *Proceedings of the ACM Conference on Information and Knowledge Management*, pages 415–424, 2012. doi: 10.1145/2396761.2396816.
50. Thomas G Dietterich. Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer, 2000.
51. Javed A. Aslam and Mark Montague. Models for metasearch. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 276–284, New York, NY, USA, 2001. ACM. doi: 10.1145/383952.384007.
52. Steffen Rendle. Factorization machines. In *Proceedings of the IEEE International Conference on Data Mining*, pages 995–1000. IEEE, 2010. doi: 10.1109/ICDM.2010.127.
53. Yin Lou, Rich Caruana, Johannes Gehrke, and Giles Hooker. Accurate intelligible models with pairwise interactions. In *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining*, pages 623–631, 2013. doi: 10.1145/2487575.2487579.

54. Yuxiao Dong, Nitesh V. Chawla, and Ananthram Swami. Metapath2vec: Scalable representation learning for Heterogeneous Networks. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, page 135–144, New York, NY, USA, 2017. ACM. ISBN 9781450348874. doi: 10.1145/3097983.3098036.
55. Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. The MIT Press, 2016.
56. Mark Dredze, Koby Crammer, and Fernando Pereira. Confidence-weighted linear classification. In *Proceedings of the International Conference on Machine Learning*, pages 264–271, 2008. doi: 10.1145/1390156.1390190.
57. Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In *Proceedings of the International Conference on Machine Learning*, pages 1310–1318. PMLR, 2013.
58. Ting Chen, Lu-An Tang, Yizhou Sun, Zhengzhang Chen, and Kai Zhang. Entity embedding-based anomaly detection for heterogeneous categorical events. In *Proceedings of the International Joint Conference on Artificial Intelligence*, page 1396–1403. AAAI Press, 2016.
59. Jonathan C. Roberts. State of the art: Coordinated & multiple views in exploratory visualization. In *Proceedings of the International Conference on Coordinated and Multiple Views in Exploratory Visualization*, pages 61–71. IEEE, 2007. doi: 10.1109/CMV.2007.20.
60. W. Javed and N. Elmqvist. Exploring the design space of composite visualization. In *Proceedings of the IEEE Pacific Symposium on Visualization*, pages 1–8, 2012. doi: 10.1109/PacificVis.2012.6183556.
61. M. Gleicher. Considerations for visualizing comparison. *IEEE Transactions on Visualization and Computer Graphics*, 24(1): 413–423, 2018. doi: 10.1109/TVCG.2017.2744199.
62. Manasi Vartak, Aditya Parameswaran, Neoklis Polyzotis, and Samuel R Madden. SeeDB: automatically generating query visualizations. *Proceedings of the VLDB Endowment*, 2014. doi: 10.14778/2733004.2733035.
63. Zachary C. Lipton. The mythos of model interpretability. *Commun. ACM*, 61(10):36–43, September 2018. ISSN 0001-0782. doi: 10.1145/3233231.
64. Jaegul Choo and Shixia Liu. Visual analytics for explainable deep learning. *IEEE Computer Graphics & Applications*, 38 (4):84–92, 2018.
65. Fred Hohman, Minsuk Kahng, Robert Pienta, and Duen Horng Chau. Visual analytics in deep learning: An interrogative survey for the next frontiers. *IEEE Transactions on Visualization and Computer Graphics*, 25(8):2674–2693, 2018. doi: 10.1109/TVCG.2018.2843369.
66. Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why should I trust you?" Explaining the predictions of any classifier. In *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining*, pages 1135–1144, New York, NY, USA, 2016. ACM. doi: 10.1145/2939672.2939778.
67. Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Proceedings of the International Conference on Neural Information Processing Systems*, page 4768–4777, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.
68. Harsha Nori, Samuel Jenkins, Paul Koch, and Rich Caruana. InterpretML: A unified framework for machine learning interpretability. *arXiv preprint arXiv:1909.09223*, 2019.
69. Shubhomoy Das, Md Rakibul Islam, Nitthilan Kannappan Jayakodi, and Janardhan Rao Doppa. Active anomaly detection via ensembles: Insights, algorithms, and interpretability. *arXiv preprint arXiv:1901.08930*, 2019.
70. Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.
71. Andrada Tatu, Georgia Albuquerque, Martin Eisemann, Jorn Schneidewind, Holger Theisel, Marcus Magnork, and Daniel Keim. Combining automated analysis and visualization techniques for effective exploration of high-dimensional data. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*, pages 59–66. IEEE, 2009. doi: https://doi.org/10.1109/VAST.2009.5332628.
72. Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(86): 2579–2605, 2008.
73. Patrick Riehmann, Manfred Hanfler, and Bernd Froehlich. Interactive sankey diagrams. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 233–240. IEEE, 2005. doi: 10.1109/INFVIS.2005.1532152.
74. Jing Yang, Wei Peng, M. O. Ward, and E. A. Rundensteiner. Interactive hierarchical dimension ordering, spacing and filtering for exploration of high dimensional datasets. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 105–112, 2003. doi: 10.1109/INFVIS.2003.1249015.
75. Bambang Setiono and Yunus Husein. Fighting forest crime and promoting prudent banking for sustainable forest management: The anti money laundering approach. *CIFOR*, 2005. doi: 10.17528/cifor/001881.
76. William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Proceedings of the International Conference on Neural Information Processing Systems*, page 1025–1035, Red Hook, NY, USA, 2017. Curran Associates Inc. doi: 10.5555/3294771.3294869.
77. J. Johnson, M. Douze, and H. Jégou. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, pages 1–1, 2019. doi: 10.1109/TBDATA.2019.2921572.