DATE: Dual Attentive Tree-aware Embedding for Customs Fraud Detection

Sundong Kim*
Institute for Basic Science
sundong@ibs.re.kr

Yu-Che Tsai*
National Cheng Kung University
roytsai27@gmail.com

Karandeep Singh
Institute for Basic Science
ksingh@ibs.re.kr

Yeonsoo Choi
World Customs Organization
yeonsoo.choi@wcoomd.org

Etim Ibok
Nigeria Customs Service
joshuaibok@gmail.com

Cheng-Te Li
National Cheng Kung University
chengte@mail.ncku.edu.tw

Meeyoung Cha
Institute for Basic Science
mcha@ibs.re.kr

ABSTRACT

Intentional manipulation of invoices that lead to undervaluation of trade goods is the most common type of customs fraud to avoid ad valorem duties and taxes. To secure government revenue without interrupting legitimate trade flows, customs administrations around the world strive to develop ways to detect illicit trades. This paper proposes DATE, a model of Dual-task Attentive Tree-aware Embedding, to classify and rank illegal trade flows that contribute the most to the overall customs revenue when caught. The strength of DATE comes from combining a tree-based model for interpretability and transaction-level embeddings with dual attention mechanisms. To accurately identify illicit transactions and predict tax revenue, DATE learns simultaneously from illicitness and surtax of each transaction. With a five-year amount of customs import data with a test illicit ratio of 2.24%, DATE shows a remarkable precision of 92.7% on illegal cases and a recall of 49.3% on revenue after inspecting only 1% of all trade flows. We also discuss issues on deploying DATE in Nigeria Customs Services, in collaboration with the World Customs Organization.

CCS CONCEPTS

• **Applied computing → E-government**.

KEYWORDS

Customs frauds detection; Tree-based embedding model; Multi-task learning; E-government

ACM Reference Format:
Sundong Kim, Yu-Che Tsai, Karandeep Singh, Yeonsoo Choi, Etim Ibok, Cheng-Te Li, and Meeyoung Cha. 2020. DATE: Dual Attentive Tree-aware Embedding for Customs Fraud Detection. In *26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '20), August 23–27, 2020, Virtual Event, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/XXXXXX.XXXXXX

*Both authors contributed equally to this research.

KDD'20, August 23–27, 2020, Virtual Event, USA

ACM ISBN 978-1-4503-7998-4/20/08…$15.00
https://doi.org/10.1145/XXXXXX.XXXXXX

Table 1: Types of customs fraud and the scope of this research.

Scope	Fraud	Illicit motives
✓	*Undervaluation* of trade goods	To avoid ad-valorem customs duty, or conceal illicit financial flows from exporters
	Misclassification of HS code	To get a lower tariff rate applied or trade prohibited goods by avoiding restriction
	Manipulation of origin country	To get a preferential tariff rate under a free trade agreement
✗	*Smuggling* without declaration	To trade prohibited goods by avoiding restriction and custom duties
	Overvaluation of trade goods	To disguise illicit financial flows as legitimate trade payment from importers

1 INTRODUCTION

Customs are government authorities responsible for controlling the flows of goods and passengers across borders and collecting customs duties and taxes from traders. According to the World Customs Organization (WCO)[1], customs administrations cleared $19.7 trillion worth of imports, 1.4 billion passengers, and collected 30% of tax revenue globally in 2018 alone. Given the astronomical volume of cross-border flows, how to *control less but better* is the main task of customs administrations. On the other hand, traders may be tempted to manipulate and omit some declaration details to avoid customs duties, taxes, and regulations. Table 1 summarizes the types of customs fraud with the corresponding illicit motives.

Undervaluation is the most common type of customs fraud, where importers or exporters declare the value of trade goods at lower prices than actual ones, mainly to avoid ad valorem customs duties and taxes. However, in a broad definition, it encompasses misclassification in HS codes — a standardized international Harmonized System to classify globally traded products — and manipulation of countries of origin with a motive to avoid customs duties and taxes. An example would be to declare a television (HS 852859, 8% duty) as a PC monitor (HS 852852, 0% duty). This paper uses a broad definition of undervaluation. There are other kinds

[1] http://www.wcoomd.org/

of customs tax fraud, including smuggled goods that try to avoid invoicing altogether.

Fraud detection in trades, trade-related financial transactions, and cross-border passengers are the pillars of customs administration. Some customs administrations have successfully adopted machine learning models in their fraud detection systems, and many administrations are planning or requesting international support. Recently, the WCO has started looking into the potential of machine learning and data mining methods for fraud detection [8, 13, 31]. Nonetheless, the usage of machine learning in the field of customs has been limited, especially for developing economies, and there remain several challenges, such as interpretability, availability of historical data, ever-changing patterns of fraud, availability of labeled data, imbalanced data, and privacy concerns.

Building a customs fraud detection model needs to address the following considerations. First, interpretability is an essential requirement for customs administrations. In practice, inspecting an import involves reviewing tens of documents, finding clues of fraud out of hundreds of packages in a warehouse, and even pacifying angry traders who suffer from additional cost and delays due to the inspection. If our model fails in informing inspectors of any reasons for its targeting, they may resist the model's predictions. Second, the identifiers of traders and trade goods are essential features in detecting customs fraud. In our preliminary result with a random forest model, non-compliance records of importers and HS codes were the most critical variables. DATE was designed to fully utilize the two variables, overcoming their extremely high cardinality (165 thousands of importers and six thousands of HS codes). Lastly, we expect our model to deliver better performance than the current and traditional machine learning approaches for customs fraud detection.

To incorporate the points mentioned above, we propose Dual Attentive Tree-aware Embedding (DATE) for customs fraud detection. Given an import transaction, along with the corresponding trader and trade goods, our primary goal is to predict its illicitness, a binary label indicating whether it is illicit or not. Since the identification of trade frauds would lead to an increase in the customs tax revenue [13], we also aim at predicting the raised revenue as the secondary goal. The main idea of DATE is three-fold. First, we pre-train a tree-based model to identify the most significant combinations of original features, termed cross features. Cross features not only provide the effective representation capability to deal with structured data, but also allow our model to be equipped with interpretability. Second, by learning the embeddings of cross features, we devise a dual attentive mechanism to generate the representation of a transaction. We exploit multi-head self-attention to learn the interactions between cross features, and utilize an attention network to encode how the trader and trade goods (HS code) are correlated with cross features. Third, we devise a dual-task learning technique that predicts the illicit probability and jointly maximizes the amount of raised customs tax for customs authorities. The dual-task learning makes the classification task more effective and helps customs to identify the most *valuable* illicit transactions.

Results obtained with our dataset show that around 90% of the frauds can be detected, and 89% of the total additional tax revenue can be collected by inspection of only the top 10% suspicious transactions identified by the proposed DATE model. Automating the inspection process offers improved tax fraud detection, increased operational efficiency, and reduced human resources required to perform the inspections. Better tax fraud detection enhances revenue inflow as well as the country's trade competitiveness.

2 RELATED WORK

Fraud detection is a general task of interest that is relevant to many industries. With the advent of data mining and machine learning, numerous advances have been made, such as the decision-tree based approaches [1–3, 29]. When it comes to customs administrations (i.e., the target domain of this research), most tax authorities until today are using rule-based methods [17]. Rule-based systems are interpretable and straightforward, but are brittle against any new behaviors and changes, subjective to expert knowledge, and are cumbersome to maintain [16, 21]. Machine learning-based systems can overcome such limitations.

As far as customs fraud detection is concerned, the published literature is limited primarily due to the proprietary nature of the task. There, however, exist several efforts that can be summarized as supervised, unsupervised, and semi-supervised learning techniques. Sometimes an ensemble of these techniques is deployed for better performance. For example, the Belgian customs have tested an ensemble method of a support vector machine-based learner in a confidence-rated boosting algorithm [26]. The Columbia customs have demonstrated the use of unsupervised spectral clustering in detecting tax fraud with limited labels [10]. Most recently, an ensemble of tree-based approaches, support vector machine, the neural network has been tested on customs data warehouse in Indonesia [6].

Another research sheds light on customs fraud in international shipping records in the Netherlands [25]. In this work, a model was built based on the Bayesian network and neural networks that compare the presence of goods on the cargo list of shipments against the accompanying documentation of a shipment, to determine whether document fraud is perpetrated. There are other studies that utilize approaches like the Benford's Law to detect fraud in customs audits [22]. Another research employed a deep learning model to segregate high risk and low-risk consignment on randomly selected 200,000 data from Nepal Customs of the year 2017 [23].

Some countries have worked to develop their fraud detection systems. The Korea Customs Service developed an electronic customs clearance system Uni-pass[2] that recently evolved to deploy AI-based risk management module named IRM-pass. New Zealand established Joint Border Analytics (JBA) in 2016 to leverage data analysis and to mine to gain new insights into border and customs risks. While their methodology is not public, JBA uses data from different sources such as cargo, passenger, and mail streams as well as open-source data. JBA takes to looking at a range of customs risks and issues, including undervaluation, drug importation, and even the Darknet drugs markets.[3] Brazil's federal revenue authority developed a sophisticated selection system called SISAM [11], which reports the error rate for diverse fraud scenarios and outputs an assessment report to assist tax auditors. The algorithm for targeting system is proprietary and leverages recent advances from

[2]https://tinyurl.com/rc4vfh4
[3]https://tinyurl.com/t5tmq8n

computer vision and natural language processing. SISAM is known to be developed from hierarchical pattern Bayes [12].

Most countries that do not own fraud-detection systems use ASYCUDA, a computerized customs management system designed by the United Nations. Currently, more than 90 countries use this system worldwide.[4] ASYCUDA facilitates the inclusion of simple rule-based methodologies for fraud detection, which our algorithm aims to builds upon. Some fraud detection methodologies are applicable to ASYCUDA. For example, traditional fraud detection techniques, like mirror data analysis, can be performed to identify any discrepancy between the import and export sides of a trade [9]. However, it is rare to have transaction-level data for trading sides, and it is not straightforward to establish the exact type of goods and its retail price from import declarations.

3 PROBLEM SETTINGS

3.1 Dataset

This paper employed transaction-level import data of Nigeria, a partner country of WCO. A total of 1,932,151 import trade flows from 2013 to 2017 comprise the data. Table 2 lists the key data fields, where some fields such as trader ID had been anonymized. The trade goods are categorized by *tariff.code* that combines the six-digit HS code used worldwide and the four-digit code granted by the country. The bottom two rows are target variables to predict, which are generated after the inspection result.

Figure 1 depicts the daily transaction volume, daily illicit rate, and daily revenue of the data we have utilized. It is worth noting that due to the customs rules of Nigeria, every custom is subjected to detailed inspection (i.e., currently achieving a near 100% inspection rate). Therefore, illicit and legitimate transactions are accurately labeled in this complete log, except for the case of smuggling. Nonetheless we mention that our dataset includes only the logs of *single* goods declarations and does not include *multi* goods declarations. Hence, the transaction trend is not representative of the country's *entire* import volume. The overall illicit rate is 3.83%, but varies each year slightly. The daily tax revenue from detecting illicit transactions also varies throughout the year. According to the data, frequent importers commit fewer frauds than those importers with fewer transactions, as identified in the bottom right figure. Similarly, we observe that certain HS codes are used more frequently in illicit transactions. Based on these observations, we later treat importer ID and HS code as part of a key signal in fraud detection.

We quantify the risk indicators of importers, declarants, HS code, and countries of origin in the model. The risk indicators are calculated from their non-compliance records. For instance, importers are ranked by the specific number of fraudulent imports divided by their corresponding total transaction volume. The importers, whose ranks are above the 90th percentile, were regarded as high-risk importers, and their risk indicators were given a value of 1; otherwise, 0. Various nonlinear relationships such as *unit.value* (=cif.value/quantity), *value/kg* (=cif.value/gross.weight), *tax.ratio* (=total.taxes/cif.value), *unit.tax* (=total.taxes/quantity), as well as *face.ratio* (=fob.value/cif.value) were added. These features help identify fraudulent transactions. We also add three temporal features, Day of Year, Week of Year, and Month of Year of the imported goods.

[4] https://asycuda.org/en/user-countries/

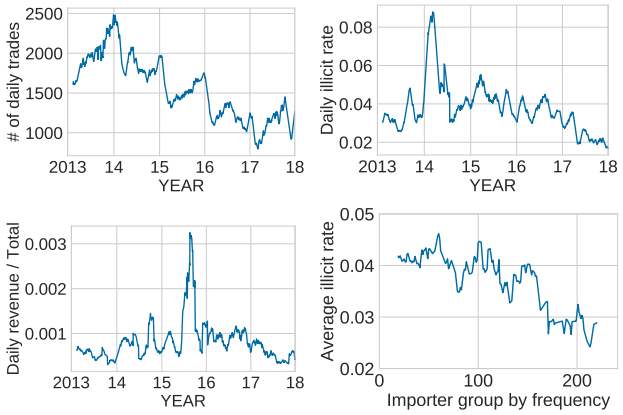

Figure 1: Major statistics of the dataset.

3.2 Customs Fraud Detection Problem

The customs administration aims to select which import trade flows inspectors should prioritize and manually verify. We formulate the customs fraud detection problem as follows.

Problem: *Given an import trade flow t, along with its importer u and HS code c of the goods, the goal is to predict both the fraud score y^{cls} and the raised revenue y^{rev} obtainable by inspecting transaction t.*

By using two predicted values, y^{cls} and y^{rev} from all trade flows[5] $T = \{t_1, \ldots, t_N\}$, customs administration can select fraudulent transactions $T_F \subset T$ according to their criteria. Anecdotal reports show that customs offices in developed economies are capable of no more than 5% inspection rate, due to astronomical trade volume. In light of this limitation, developing algorithms that detect fraud with minimal inspection is critical. In the remainder of this paper, we will describe the proposed DATE model that predicts y^{cls} and y^{rev} of each import traffic flow t and demonstrate its performance.

4 OUR MODEL: DATE

The Dual-task Attentive Tree-aware Embedding (DATE) model consists of three stages. The first stage pre-trains a tree-based classifier to generate cross features of each transaction. The second stage is a dual attentive mechanism that learns both the interactions among cross features and the interactions among importers, HS codes, and cross features. The third stage is the dual-task learning by jointly optimizing illicitness classification and revenue prediction. The overall architecture is depicted in Figure 2.

4.1 Tree-based Cross Feature Embeddings

The strength of tree-based methods lies at its effectiveness and interpretability [28]. A decision tree could be seen as an effective algorithm to express high-order cross features from the original features. Suppose we have a decision tree $T = \{V, E\}$, where V and E are sets of nodes and edges, respectively. The node-set V can be divided into three subsets, the root node $\{v_R\}$, internal nodes V_I, and leaf nodes V_L, $V = \{v_R\} \cup V_I \cup V_L$. Each internal node $v_I \in V_I$ splits a feature in the decisive space. A path that connects

[5] Terms in each of sets {"trade flow", "transaction"}, {"HS code", "item", "goods"}, and {"fraud", "illicitness"} are interchangeably used throughout this paper.

Table 2: Overview of the transaction-level import data, in which the description and example of each variable are provided.

Type	Variable	Description	Example
Features	*sgd.id*	An individual numeric identifier for Single Goods Declaration (SGD).	SGD347276
	sgd.date	The year, month and day on which the transaction occurred.	13-11-28
	importer.id	An individual identifier by importer based on the tax identifier number (TIN) system.	IMP364856
	declarant.id	An individual identification number issued by Customs to brokers.	DEC795367
	country	Three-digit country ISO code corresponding to transaction.	USA
	office.id	The customs office where the transaction was processed.	OFFICE91
	tariff.code	A 10-digit code indicating the applicable tariff of the item based on the harmonised system (HS).	8703232926
	quantity	The specified number of items.	1
	gross.weight	The physical weight of the goods.	150kg
	fob.value	The value of the transaction excluding, insurance and freight costs.	$350
	cif.value	The value of the transaction including the insurance and freight costs.	$400
	total.taxes	Tariffs calculated by initial declaration.	$50
Prediction Target	***illicit***	Binary target variable that indicates whether the object has fraud.	1
	revenue	Amount of tariff raised after the inspection, only available on some illicit cases.	$20

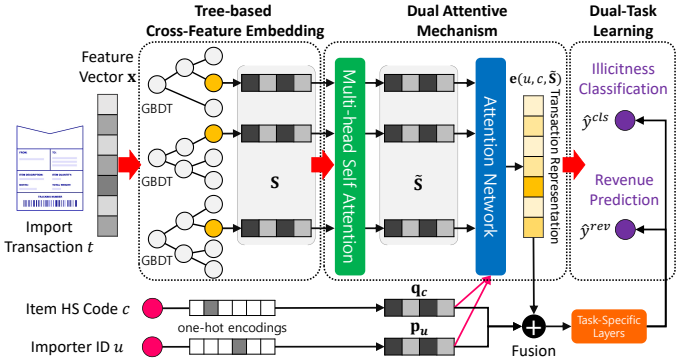

Figure 2: The architecture of our DATE model.

v_R and any node in $v_L \in V_L$ represents a *decision rule* or a *decision path*. A feature vector $\mathbf{x} \in \mathbb{R}^k$ (k is the dimension) is assigned to a leaf node v_L, which represents a decision rule. A rule can be the path from root to the leaf, e.g., "gross.weight > 100kg $\wedge$ quantity > 5" as it passes two internal nodes. We call each decision path instance as a *cross feature* that combines multiple feature ranges together. A single tree is not expressive to cover all of the complex patterns, and hence we consider to build a forest instead. We use Gradient Boosting Decision Tree (GBDT) that has been proved be effective in many classification tasks [4, 5]. GBDT is an ensemble learning method that boosts the prediction by growing decision trees sequentially. The prediction of GBDT is made by aggregating the results of trees:

$$\hat{y}_{GBDT}(\mathbf{x}) = \sum_{t=1}^{\tau} \eta \hat{y}_{DT}^{t}(\mathbf{x}), \tag{1}$$

where τ is the number of trees, and $\hat{y}_{DT}^{t}$ is the predicted value by t-th decision tree. The shrinkage parameter $0 < \eta < 1$ controls the learning rate of the procedure. Inspired by existing tree-based models [14, 28], we use the pre-trained GBDTs to obtain cross features from the feature vector of a transaction.

Assume that we are given τ trees, and let N_L be the total number of leaves in the forest. When an input vector $\mathbf{x}$ is given, the GBDT decides which leaf node should correspond to the input. Each activated leaf in a tree is treated as a cross feature, $\mathcal{F}_i$, as a one-hot encoding vector representing a leaf node in the decision tree. By concatenating them together, a multi-hot vector $\mathbf{p} \in \mathbb{R}^{N_L}$ can be produced, in which elements of value 1 indicate activated leaves and 0, non-activated ones in $\mathbf{p}$.

To further encode high-level semantic feature, we project each cross feature $\mathcal{F}_i$ into a learnable dense embedding vector $\mathbf{s}_i \in \mathbb{R}^d$, where d is the dimensionality. Given a multi-hot cross feature vector $\mathbf{p}$ obtained from GBDT, we collect those embedding vectors which are corresponding $p_i \neq 0$ ($p_i \in \mathbf{p}$), and construct an embedding matrix $\mathbf{S} \in \mathbb{R}^{\tau \times d}$ since we have τ trees, given by:

$$\mathbf{S} = \varphi\left(\left[p_1\mathbf{s}_1, p_2\mathbf{s}_2, ..., p_{N_L}\mathbf{s}_{N_L}\right]\right), \forall p_i \neq 0 \text{ and } p_i \in \mathbf{p}, \tag{2}$$

where $\varphi(\mathbf{M})$ is an operation that removes all zero row vectors from a matrix $\mathbf{M}$. The derived matrix $\mathbf{S}$ depicts all latent representations of non-zero cross features, and will be used for prediction.

The benefit of learning a dense embedding $\mathbf{s}_i$ of each cross feature $\mathcal{F}_i$ is two-fold. First, it can model the underlying correlation between different leaves and map similar cross features into near-by points in the embedding space. Second, since the embedding matrix $\mathbf{S}$ is learnable during training, it allows us to incorporate additional information, such as importer and item id. In other words, adopting a learnable vector, instead of a static vector provides our model some flexibility of customization because of different countries' customs record a variety of information.

4.2 Dual Attentive Mechanism

We present a dual attention mechanism, leaf-wise self-attention, and attention network. The former is to model the correlation between cross features from different views. The latter is to learn how each cross feature contributes to the importer and the HS code. Then we fuse embeddings to have the representation of each transaction.

4.2.1 Leaf-wise Self-attention. While some cross features concern about the price and quantity of items and some emphasize on

whether an importer is risky, their interactions can further reveal the potential illicit behaviors. Given the embedding matrix $\mathbf{S}$ of cross features (leaf nodes), we aim to learn how the interactions between cross features affect the prediction. A leaf-wise self-attention mechanism is developed to model the interactions between leaf embeddings. Let $\mathbf{F}^Q$, $\mathbf{F}^K$, and $\mathbf{F}^V$ denote matrices packed from vectors of queries, keys, and values in self-attention [27], respectively. We first define the *scaled dot product attention* (SDPA):

$$SDPA(\mathbf{F}^Q, \mathbf{F}^K, \mathbf{F}^V) = softmax\left(\frac{\mathbf{F}^K(\mathbf{F}^Q)^\top}{\sqrt{d}}\mathbf{F}^V\right), \tag{3}$$

in which dot product is used to capture the similarity between vectors. Instead of performing a single attention function with d_k-dimensional queries, keys, and values, we project queries, keys, and values n_h times with different learned linear projections to d_k, d_k, and d_v dimensions, respectively. On each of the projected queries, keys, and values, we can perform the attention mechanism in parallel, yielding to d_v-dimensional output values. To achieve the goal of modeling different aspects of interactions between leaf (i.e., cross feature) embeddings, we utilize multi-head (MH) self-attention:

$$MH(\mathbf{F}^Q, \mathbf{F}^K, \mathbf{F}^V) = \text{Concat}(head_1, head_2, ..., head_{n_h})\mathbf{W}^O, \tag{4}$$

where $head_i = SDPA(\mathbf{F}^Q\mathbf{W}_i^Q, \mathbf{F}^K\mathbf{W}_i^K, \mathbf{F}^V\mathbf{W}_i^V)$, $\mathbf{W}_i^Q, \mathbf{W}_i^K, \mathbf{W}_i^V \in \mathbb{R}^{d\times d_k}$ and $\mathbf{W}^O \in \mathbb{R}^{n_h d_v \times d}$ are learnable weights, and n_h is the number of heads. That said, dense layers (i.e., $\mathbf{W}_i^Q, \mathbf{W}_i^K, \mathbf{W}_i^V$) are used to project the queries, keys, and values into their vector spaces. Then we apply self-attention based on the leaf embedding matrix $\mathbf{S}$. Since the queries, keys, and values are all equal to the leaf embedding matrix, i.e., $\mathbf{F}^Q = \mathbf{F}^K = \mathbf{F}^V = \mathbf{S}$, we can produce the multi-head attention-aware cross feature embedding matrix $\tilde{\mathbf{S}} = MH(\mathbf{S}, \mathbf{S}, \mathbf{S})$, which is used to construct the attention network below.

4.2.2 Attention Network. We learn the contributing weights of cross features derived from different trees to exploit the interactions of cross features for intelligent prediction. Based on recent advances of attention mechanism [27, 28], we propose an attention network that considers the importer ID and the item identifier (i.e., HS code) to model the tri-interactions among cross features, importers, and items regarding a given transaction. Given a pair (u, c) of importer u and an HS code c, along with the cross feature embeddings $\tilde{\mathbf{S}}$, the aim is to generate the attention weight a_{uci}. The attention weights can reflect which cross feature $\mathcal{F}_i$ is more significant in determining illicit behaviors concerning a given importer u and item c:

$$\begin{aligned} a_{uci} &= \mathbf{h}^\top \phi(\mathbf{W}[\mathbf{p}_u \odot \mathbf{q}_c, \mathbf{s}_i] + \mathbf{b}), \\ \alpha_{uci} &= \frac{\exp(a_{uci})}{\sum \exp(a_{uci})}, \end{aligned} \tag{5}$$

where $\mathbf{W} \in \mathbb{R}^{d\times 2d}$ and $\mathbf{b} \in \mathbb{R}^d$ are the trainable weight matrix and the bias vector, vectors $\mathbf{p}_u \in \mathbb{R}^d$ and $\mathbf{q}_c \in \mathbb{R}^d$ represent the embeddings of importer ID u and HS code c, ϕ is ReLU activation function, and $\mathbf{s}_i \in \tilde{\mathbf{S}}$ is the embedding vector of cross feature $\mathcal{F}_i$, respectively. The hidden vector $\mathbf{h} \in \mathbb{R}^d$ projects the hidden vector into a scalar weight for output. We set $\mathbf{p}_u = \mathbf{0}$ and $\mathbf{q}_c = \mathbf{0}$ (i.e., zero vectors) upon encountering any unseen importers and HS codes. We use element-wise product $\mathbf{p}_u \odot \mathbf{q}_c$ to capture the co-occurrence of importer and item, and concatenate it with the cross feature embedding $\mathbf{s}_i$ to learn attentive weights. We use the attentive weights to aggregate cross features, and generate the final representation $\mathbf{e}(u, c, \tilde{\mathbf{S}})$ of a transaction declared by importer u on item c via: $\mathbf{e}(u, c, \tilde{\mathbf{S}}) = \sum_{i=1}^{N_L} \alpha_{uci}\mathbf{s}_i$, where $\mathbf{s_i} \in \tilde{\mathbf{S}}$.

4.2.3 Embedding Fusion. Since whether a transaction is illicit highly depends on who is the importer and what is the item inside, we combine the obtained transaction representation $\mathbf{e}(u, c, \tilde{\mathbf{S}})$ with the importer embedding $\mathbf{p}_u$ and the item embedding $\mathbf{q}_c$ for prediction. By concatenating such three vectors, along with a hidden layer, we can generate a fused vector $\mathbf{e}_f(u, c, \tilde{\mathbf{S}})$, given by

$$\mathbf{e}_f(u, c, \tilde{\mathbf{S}}) = \phi(\left[\mathbf{p}_u, \mathbf{q}_c, \mathbf{e}(u, c, \tilde{\mathbf{S}})\right]\mathbf{W}_f), \tag{6}$$

where $W_f \in \mathbb{R}^{3d\times d}$ is the learnable weight matrix, and we use ReLU to be the activation function ϕ.

In customs operations, it is common to find unseen importers and new items. As we initialize the embeddings of unseen ones to be zero vectors, we will be able to deal with such input. By projecting unseen ones into the same space shared with cross features, their embeddings are encoded with useful clues on illicitness.

4.3 Dual-Task Learning

Identifying illicit transactions naturally leads to an increase in customs tax revenue [13]. However, the amount of increased taxes can be determined only when the transaction is caught illicit. This indicates that if we could estimate the amount of raised tax precisely, it has the potential to benefit the prediction of illicit classification. Multi-task learning techniques have been used to train a model that optimizes multiple objectives simultaneously [20, 24].

We propose a dual-task learning method to use the transaction information (i.e., $\mathbf{e}_f$) for both tasks of binary illicit classification and increased revenue prediction. Given the transaction feature $\mathbf{e}_f$, we introduce the task-specific layer:

$$\begin{aligned} \hat{y}^{cls}(u, c, \tilde{\mathbf{S}}) &= \sigma\left(\mathbf{r}_1^\top \mathbf{e}_f(u, c, \tilde{\mathbf{S}}) + \mathbf{b}_1\right), \\ \hat{y}^{rev}(u, c, \tilde{\mathbf{S}}) &= \mathbf{r}_2^\top \mathbf{e}_f(u, c, \tilde{\mathbf{S}}) + \mathbf{b}_2, \end{aligned} \tag{7}$$

where $\mathbf{r}_1, \mathbf{r}_2 \in \mathbb{R}^d$ denotes the hidden vectors of task-specific layers that project $\mathbf{e}_f$ into the prediction tasks of binary illicitness and raised revenue, respectively. σ is the sigmoid function. $\hat{y}^{cls}(u, c, \tilde{\mathbf{S}})$ is the predicted probability of a transaction being illicit, and $\hat{y}^{rev}(u, c, \tilde{\mathbf{S}})$ is the predicted raised revenue value of a transaction. The final objective function $\mathcal{L}_{DATE}$ is given by:

$$\mathcal{L}_{DATE} = \mathcal{L}_{cls} + \alpha\mathcal{L}_{rev} + \lambda\|\Theta\|^2, \tag{8}$$

where Θ denotes all learnable model parameters, $\mathcal{L}_{cls}$ is the cross-entropy loss for binary illicitness classification, $\mathcal{L}_{rev}$ is the mean-square loss for raised revenue prediction, given by:

$$\begin{aligned} \mathcal{L}_{cls} &= -\sum_i y_i^{cls} \log(\hat{y}_i^{cls}(u_i, c_i, \tilde{\mathbf{S}}_i)) + (1 - y_i^{cls}) \log(1 - \hat{y}_i^{cls}(u_i, c_i, \tilde{\mathbf{S}}_i)), \\ \mathcal{L}_{rev} &= \frac{1}{n}\sum_i \left(y_i^{rev} - \hat{y}_i^{rev}(u_i, c_i, \tilde{\mathbf{S}}_i)\right)^2, \end{aligned} \tag{9}$$

where y_i^{cls} and y_i^{rev} are the ground-truth illicitness class and raised revenue of transactions t_i, respectively, λ is the regularization hyperparameter to prevent overfitting, and n is the number of training

samples. The hyperparameter α is used to balance the contributions between tasks. We use mini-batch gradient descent to optimize the objective function $\mathcal{L}_{DATE}$, along with *Ranger*, a synergistic optimizer combining Rectified Adam [19] and LookAhead [30] based on a dynamic rectifier to adjust the adaptive momentum of Adam [15].

5 EVALUATION

5.1 Evaluation Settings

5.1.1 Data Splitting. We split our data into training, validation and testing sets on temporal basis. The data from the last year (Y2017) is used as a testing set. The last month worth of data from the previous four years (Y2013–2016) is held out as a validation set. Accordingly, the size of training, validation, testing set becomes 1,635,157 (84.4%), 25,948 (1.3%), 276,440 (14.3%), and the corresponding illicit ratios are 3.94%, 2.51%, and 2.24%, respectively.

5.1.2 Evaluation Metrics. To evaluate model performance, we used five metrics. Given that customs administration inspect a limited quantity of goods, four inspection rate values are used — 1%, 2%, 5%, 10% — to measure precision, recall, and recall on revenue [11]. In practice, an adequate inspection rate varies according to customs policy and with the context. Besides, we also report AUC and F1-score to evaluate the overall model performance. In the following definitions, top n% refers to the top n% most suspicious transactions suggested by each algorithm.

- **Precision@n%**: This metric explains how many transactions are illicit, among the top n% of transactions.
- **Recall@n%**: This metric represents how many inspected transactions (i.e., the n% of flows chosen) have been successfully screened out of the total illicit volume.
- **Revenue@n%**: It is the total revenue in top n% transactions identified by a model divided by the total revenue among all transactions. This metric explains how much customs duties can be generated from top n% of transactions, as compared to the revenue generated by inspecting the entire transactions.
- **AUC**: This is a scale and the threshold invariant metric ranges from 0 to 1. It is used to evaluate the outcome of the predictive algorithm for the entire data.
- **F1-score**: This is a measure of a test's accuracy among six different inspection rates. In detail, we report the best F1-score calculated by adjusting the illicit threshold from 0.1 to 0.6 for every 0.1.

5.1.3 Parameter Settings. We use the default parameter settings for the XGBoost model[6]. The number and the maximum depth of trees are 100 and 4, respectively. If a transaction is assigned to n^{th} or higher leaves, we adjust its index to n. The embedding dimension of each leaf, importer ID, and HS code is set to 16, and we use the 16-dim attention layer and self-attention with 4 heads. The learning rate of 0.005. The mini-batch size is set to 128. We set the weight of final loss (Eq. 8) $\alpha = 10$, and the regularization parameter $\lambda = 0.01$. During the training process, we save the best model using validation set and accordingly predict test transactions after the termination of the fifth epoch.

[6]https://github.com/dmlc/xgboost/blob/master/doc/parameter.rst

5.2 Performance Comparison

We compared DATE with *six* baselines. For a fair comparison between tree-based approaches, we built a GBDT from the third baseline and shared the trees with GBDT+LR, TEM and DATE. Both outcomes of our model using $\hat{y}^{cls}$ and $\hat{y}^{rev}$ are reported as DATE_{CLS} and DATE_{REV}, averaged over 20 runs.

- **Price**: Widely used targeting method by selecting in the order of the most valuable transactions.
- **Importer**: Targeting transactions in the order of importers reported by the highest rate of fraud so far.
- **IForest** [18]: Tree-based anomaly detection algorithm trained on clean transactions to detect whether a new transaction is an outlier. That said, IForest treats outliers as illicit ones.
- **GBDT (XGBoost)** [7]: State-of-the-art tree-based model with cross features, trained on binary label y^{cls}.
- **GBDT + LR** [14]: Logistic regression based on cross features extracted from GBDT, trained on binary label y^{cls}.
- **TEM** [28]: State-of-the-art tree-enhanced embedding model with attention networks. We change its objective function from Bayesian Personalize Ranking (BPR) to cross entropy for classification.

Table 3 displays the performance with respect to Precision@n%, Recall@n%, Revenue@n%, AUC, and F1 score. Among baselines, tree-based models show fairly good results and perform much better than Price and Importer. GBDT+LR outperforms GBDT in all metrics, which demonstrates the model's strength in flexibility of giving weights to different cross features. TEM shows comparable performance against other baseline methods in terms of n equals to 2%, 5% and 10%. The performance of TEM indicates the effectiveness of using dense vector to embed cross feature into a low dimensional space and, incorporating attention mechanism to give dynamic weights for cross feature.

Although TEM achieves great performance, it fails to capture interactions among cross features. Besides, none of the baseline methods consider optimizing classification loss and revenue prediction simultaneously. In our DATE, we utilize self-attention to capture interactions among cross features and further obtain aspect-level embeddings by concatenating different heads in self-attention. Therefore, the performance of the DATE_{CLS} model is far ahead of other baselines in all measures except revenue. We can also confirm that DATE_{REV} is the most effective method to guarantee the most significant revenue for customs administration.

Due to a large imbalance in data, AUC of strong baselines are above 90%, so the results say that F1-score can better discriminate the model performance for overall prediction. For customs detection problem with n% inspection rate, revenue differences are substantial even between the state-of-the-art models, which prove the effectiveness of our proposed algorithm.

5.3 Ablation Analysis

To validate the contribution of each component of DATE, we conduct an ablation study by examining the performance after removing each component, listed and denoted as follows.

- **DATE_{CLS} (Full Model)**: Use all components.
- **w/o multi-head self-attention (MSA)**: Simple version without learning the relation between cross features.

Table 3: Performance comparison between baselines and the proposed DATE.

	n = 1% (Selecting top 1%)			n = 2%			n = 5%			n = 10%			Overall	
Model	Pre.	Rec.	Rev.	Pre.	Rec.	Rev.	Pre.	Rec.	Rev.	Pre.	Rec.	Rev.	AUC	F1
Price	2.75%	1.23%	15.17%	2.23%	1.99%	20.64%	2.06%	4.60%	34.95%	2.30%	10.28%	50.98%	67.57%	7.81%
Importer	11.43%	5.10%	4.36%	9.41%	8.39%	7.56%	6.47%	14.43%	13.18%	5.22%	23.31%	30.31%	59.20%	9.10%
IForest	5.61%	2.50%	14.30%	6.19%	5.52%	23.14%	5.66%	12.62%	40.62%	5.12%	22.85%	54.14%	66.89%	5.28%
GBDT	90.01%	40.15%	24.59%	66.16%	59.04%	38.89%	32.19%	71.80%	57.20%	17.58%	78.42%	66.86%	93.38%	63.69%
GBDT+LR	90.95%	40.40%	27.18%	72.94%	65.09%	44.22%	35.02%	78.11%	63.77%	18.72%	83.54%	73.77%	94.82%	68.76%
TEM	88.72%	39.59%	39.48%	74.70%	66.43%	58.48%	37.39%	83.41%	78.58%	19.91%	88.54%	85.02%	96.52%	70.55%
DATE$_{CLS}$	**92.66%**	**41.33%**	44.97%	**80.79%**	**72.05%**	67.14%	**38.77%**	**86.49%**	84.35%	**20.24%**	**90.29%**	89.03%	**96.79%**	**75.32%**
DATE$_{REV}$	82.25%	36.63%	**49.29%**	79.93%	71.22%	**68.48%**	38.74%	86.41%	**84.57%**	20.11%	89.74%	**89.2%**	95.66%	75.23%

- **w/o fusion with HS & Importer embeddings (FHI)**: Variant of the model in which HS and importer ID are not utilized as the inputs for the second stage.
- **w/o dual task leaning (DTL)**: After training the model only on the binary labels, use the predicted probability values for selecting transactions.
- **w/o attention network (AN)**: Treat each cross feature equally without considering the dynamic attention weights.

Table 4: Results on the ablation study of the proposed DATE model.

	n = 1%		n = 5%		n = 10%	
Model	Pre.	Rev.	Pre.	Rev.	Pre.	Rev.
Full Model	**92.66%**	**44.97%**	**38.77%**	**84.35%**	**20.24%**	**89.03%**
w/o MSA	91.83%	41.22%	38.47%	82.64%	20.17%	86.34%
w/o FHI	89.89%	27.91%	36.03%	80.39%	19.12%	78.79%
w/o DTL	90.72%	35.11%	37.57%	78.46%	19.74%	85.25%
w/o AN	91.58%	40.20%	38.11%	80.54%	19.02%	87.69%

Table 4 shows the effectiveness of each module. We can have the following findings. First, all measures drop substantially when removing the fusion operation in our model. This shows that using only cross features without importer and item information hurts the performance. The step of Fusion is absolutely necessary because it simultaneously learns the correlation among cross features, importers, and items so that various behaviors of importers and items can be distinguished (e.g., distinguishing new importers from existing ones). Second, Revenue@n% apparently drops without utilizing dual task learning. Such a result indicates that optimizing revenue prediction and classification task simultaneously mutually reinforces the effectiveness of both tasks. Third, removing the attention network worsens the performance in all metrics. This proves the usefulness of giving dynamic weights to cross features based on how they are attended by importers and goods. Fourth, although removing self-attention nearly maintain the performance on Precision@n%, it fails to achieve comparable performance *w.r.t.* Revenue@n%. It demonstrates learning aspect-level features by multi-head self-attention can benefit the revenue prediction. In short, every component of our DATE model truly takes effect in the performance of both tasks. The design of DATE is verified.

5.4 Effects on Training Length

We wonder how much the model's performance varies depending on the amount of training data. We conducted the experiments by varying the length of the training data of six months. The validation set and the test set were kept intact, and the most recent k months were used for training. Figure 3 shows that the performance rapidly increases until about two years of past data are secured, then the increment becomes smaller. As Figure 4 shows, it can be concluded that the richer the learning data, the less new information is acquired. Thus the performance improvements are gradually reduced.

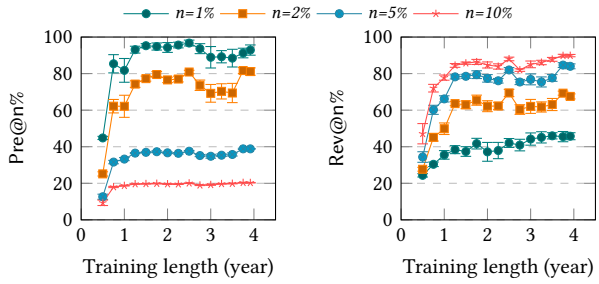

Figure 3: Performance by changing training length.

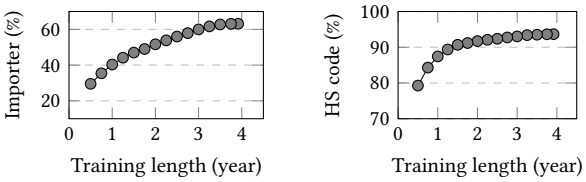

Figure 4: Coverage of two test attributes in training set.

5.5 Performance on Test Subgroups

Among transactions declared in customs, some importer IDs and HS codes are more frequent than others. To examine how the occurrence frequency affects the performance, we break down the test set into several subgroups, described as below.

- **Importer**: We divide importers into five subgroups based on the number of times they appear in the training set. For example, Imp$_{[0]}$ denotes the new importers, and Imp$_{(0,10]}$ denotes the

group of importers who appeared, but less than or equal to 10 times.

- **HS**: We divide HS codes into five subgroups in a similar way.

Table 5: Performance generated by DATE_{CLS} on different subgroups of importers and HS codes in terms of various frequencies.

	$n = 1\%$			$n = 5\%$			
Subgroup	Pre.	Rec.	Rev.	Pre.	Rec.	Rev.	Illicit rate
$\text{Imp}_{[0]}$	100.00%	39.27%	37.87%	45.72%	89.94%	87.41%	2.51%
$\text{Imp}_{(0,10]}$	98.84%	41.47%	35.60%	42.44%	89.18%	81.57%	2.37%
$\text{Imp}_{(10,50]}$	98.86%	37.25%	35.44%	46.43%	87.46%	80.43%	2.65%
$\text{Imp}_{(50,250]}$	91.72%	46.16%	38.64%	32.76%	82.31%	76.35%	1.99%
$\text{Imp}_{(250,\infty)}$	72.76%	45.24%	53.10%	23.80%	73.90%	79.39%	1.60%
$\text{HS}_{[0]}$	97.09%	27.52%	29.28%	51.21%	72.83%	70.66%	3.51%
$\text{HS}_{(0,312]}$	91.50%	41.00%	53.74%	35.40%	79.35%	88.15%	2.23%
$\text{HS}_{(312,1781]}$	96.20%	44.48%	40.15%	39.45%	91.22%	84.08%	2.16%
$\text{HS}_{(1781,8714]}$	98.19%	65.65%	54.79%	28.31%	94.67%	82.35%	1.49%
$\text{HS}_{(8714,\infty)}$	99.81%	55.47%	56.21%	33.16%	92.16%	96.00%	1.17%

Since our DATE learns the embeddings of importers and HS codes, we expect different frequencies affect the effectiveness. Table 5 shows each subgroup's prediction results and its corresponding illicit ratio based on DATE_{CLS}. Among importer subgroups, active traders (i.e., $\text{Imp}_{(250,\infty)}$) seem to commit the lowest fraud rates, comparing to the inactive traders (i.e., $\text{Imp}_{(0,10]}$ and $\text{Imp}_{[0]}$). This might reveal that importers probably tend to apply new IDs to avoid leaving illicit records and to deceive customs offices. The results on HS codes exhibit similar trends. Transactions with unpopular items tend to be predicted as illicit ones. Such a result can be linked to the manipulation of HS codes we introduce in Table 1. The results also show that DATE achieves a nearly 90% recall rate for unseen importers while the performance is lower for active importers due to their low illicit rate. Meanwhile, different from the results of importers, DATE leads to a remarkable recall rate in commonly-appeared HS codes. Since WCO creates HS codes for new goods once in a while, DATE performs poorly on $\text{HS}_{[0]}$.

5.6 Case Studies

Despite our DATE achieves remarkable performance on detecting frauds, it remains unknown what evidences are identified by DATE to predict the illicitness. To demonstrate the interpretability of DATE, Table 6 lists the comparison between illicit and licit cases with their corresponding cross features. We select the top-2 significant cross features based on the highest attention scores. Among the transactions in the data, used cars take the largest proportion, from which importers have higher possibility to report lower value so as to evade additional taxes. Hence, we select two used car transactions and analyze their differences.

For the illicit transaction, the cross features show that it has a high trade value (i.e. cif.value, fob.value) and low gross.weight while reporting a small portion of taxes obtainable (i.e., tax.ratio < 0.18%). In addition, the value per kilogram (i.e., value/kg) is relatively higher than the licit one. These variables present the high transaction value of used cars, but customs only receive a little amount of taxes. The result happened to be a convincing example of undervaluation of trade goods (mentioned in Table 1). The transaction obtainable from DATE are also examined by WCO customs

Table 6: Comparison of illicit and licit transaction w.r.t their corresponding cross features (CF) with highest attention score.

	Illicit case	Licit case
Item	Used TOYOTA VENZA, $16,863	Used TOYOTA CAMRY, $4,673
CF 1	risk.importer=0 & tax.ratio<43.7% & gross.weight<3327.43 & fob.value>$1,366	12.2%<tax.ratio<16.8% & face.ratio>62.5%
CF 2	value/kg>$2 & cif.value>$1,912 & risk.(office,importer)=0 & tax.ratio <0.18%	risk.HS.origin=0 & value/kg<$2 & cif.value>$1,640 & risk.(office,importer)=0
$\hat{y}^{cls}$	0.9849	0.0001

office, who pointed out that the transaction has a *low value per unit*. Such a statement from WCO domain experts draw an echo with cross features identified by DATE, which prove DATE has some potential to achieve human-level interpretability. Compared to the illicit case, the licit transaction exhibits normal trading information. Even though cif.value is larger than $1640, the tax ratio falls in an acceptable interval. Beside, value/kg is relatively small, indicating that there might not exists undervaluation problems. Regarding to the comparison listed in Table 6, we can conclude that DATE is able to effectively identify evidences that significantly determine whether a transaction is illicit. And the cross features with higher attention weights can be used to interpret the prediction results.

6 DISCUSSIONS

Deployment Plan and Expected Outcomes. The next phase of this research is to deploy and confirm the efficacy of DATE in a live system. This operation will be in close collaboration between the research teams, the World Customs Organization (WCO), and the Nigeria Customs Services (NCS)[7]. The test will be conducted in the two major ports of Nigeria — Tican port and Onne port — which are in charge of nearly 41% of all trade flows with a phased-in approach as follows:

- In the 1st phase, the model's predictions would be matched against the corresponding inspection results ex-post factor. It aims to verify our model's performance.
- In the 2nd phase, the officers from NCS would be informed with the model's predictions before they inspect the corresponding imports. It aims to examine whether informed officers perform better in detecting frauds.
- In the 3rd phase, NCS will reduce the number of inspections based on the model's predictions. It aims to measure additional metrics such as average clearance time, reduced cost for inspection, in comparison with reduced tax revenues.

Challenges. For the historical import transactions that were identified as illicit, the NCS e-clearance system has two values; one initially declared by importers and the other adjusted by NCS after inspections/audits. The two values may have a difference in *fob.value*, *cif.value* and *total.taxes*. While our algorithm should have been trained and tested with all the initial values, NCS had a technical glitch in extracting initial values from the system. Alternatively, for the undervalued imports which comprise 3.83% of our dataset, we used the adjusted values and conducted some experiments to check the robustness of our algorithm. In the deployment test, this

[7] https://customs.gov.ng/

Table 7: Results on revised data by DATE.

	$n = 1\%$		$n = 5\%$		$n = 10\%$	
Rescaling	Pre.	Rev.	Pre.	Rev.	Pre.	Rev.
None	94.18%	44.55%	38.77%	83.98%	20.27%	89.30%
Deterministic	92.77%	44.80%	37.13%	77.69%	19.64%	84.41%
Stochastic	95.05%	42.38%	36.96%	77.11%	19.58%	83.90%

problem will be solved as our algorithm will be tested with real-time import data before any adjustment by NCS.

Way to Leverage Existing Data. We note that the average *cif*, *fob*, and *total.taxes* values of illicit transactions are 66% higher than that of a typical transaction. Assume that there is no difference in the corresponding values for illicit and licit transactions. We can conjecture that the difference between the two groups is partially due to information updates. So, we conducted experiments by realistically rolling-back the *cif*, *fob*, and *total.taxes* values by two methods:

- **Deterministic**: multiply by a scalar value 0.6.
- **Stochastic**: multiply a Gaussian random variable $X \sim \mathcal{N}(0.6,\ 0.1^2)$.

The revised dataset on which the action was taken has similar value distribution between illicit and licit transactions, which is harder to predict. Table 7 shows that DATE performs comparably well on the revised datasets. Such results mean that DATE can learn from various patterns from import transactions, without dominated by biased distributions due to the errors. With this experiment, we expect that DATE will perform smoothly when the data pipeline is fixed, and we can receive cleaner data.

7 CONCLUSION

With the astronomically growing trade flows, customs administrations need effective and explainable methods to detect suspicious transactions. This paper presented DATE, a novel model that ranks trade flows in the order of fraud risk and to maximize customs revenue. Based on the test on five years' worth of import data, we confirm the superiority of DATE over state-of-the-art models. Predictions of DATE are interpretable, thanks to its decision rules from GBDT and the weights from the attention mechanism. Based on its outstanding performance and interpretable nature, we expect that DATE can be easily integrated into customs administrations and assist customs auditors inspect the fraud risk of individual transactions. DATE is robust against noise in input data and identify any manipulation of HS codes and countries of origin, which are also a popular type of customs fraud. In the first half of 2020, DATE will be deployed for testing under real-time import flows of Nigeria, and the trained model will be further tested on the import data of four other member countries of WCO.

ACKNOWLEDGMENTS

This work was supported by the Institute for Basic Science (IBS-R029-C2), Ministry of Science and Technology (MOST) of Taiwan (MOST Young Scholar Fellowship 109-2636-E-006-017 and grant 108-2218-E-006-036), Academia Sinica (AS-TP-107-M05), and WCO Customs Cooperation Fund of Korea (CCF Korea).

REFERENCES

[1] Aisha Abdallah, Mohd Aizaini Maarof, and Anazida Zainal. 2016. Fraud detection system: A survey. *Journal of Network and Computer Applications* 68 (2016).
[2] Aderemi O Adewumi and Andronicus A Akinyelu. 2017. A survey of machine-learning and nature-inspired based credit card fraud detection techniques. *International Journal of System Assurance Engineering and Management* 8, 2 (2017).
[3] Mohiuddin Ahmed, Abdun Naser Mahmood, and Md. Rafiqul Islam. 2016. A survey of anomaly detection techniques in financial domain. *Future Generation Computer Systems* 55 (2016), 278–288.
[4] Ismail Babajide Mustapha and Faisal Saeed. 2016. Bioactive molecule prediction using extreme gradient boosting. *Molecules* 21, 8 (2016), 983.
[5] Jonathan Burez and Dirk Van den Poel. 2009. Handling class imbalance in customer churn prediction. *Expert Systems with Applications* 36, 3 (2009).
[6] Canrakerta, Achmad Nizar Hidayanto, and Yova Ruldeviyani. 2020. Application of business intelligence for customs declaration: A case study in Indonesia. *Journal of Physics: Conference Series* 1444 (2020), 012028.
[7] Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A scalable tree boosting system. In *KDD*. 785–794.
[8] Bassem Chermiti. 2019. Establishing risk and targeting profiles using data mining: Decision trees. *World Customs Journal* 13 (2019).
[9] Yeonsoo Choi. 2019. Identifying trade mis-invoicing through customs data analysis. *World Customs Journal* 13, 2 (2019).
[10] Daniel de Roux, Boris Perez, Andrés Moreno, Maria del Pilar Villamil, and César Figueroa. 2018. Tax fraud detection for under-reporting declarations using an unsupervised machine learning approach. In *KDD*. 215–222.
[11] Jorge Jambeiro Filho. 2015. Artificial intelligence in the customs selection system through machine learning (SISAM). *Receita Federal do Brasil* (2015).
[12] Jorge Jambeiro Filho and Jacques Wainer. 2008. HPB: A model for handling BN nodes with high cardinality parents. *JMLR* 9 (2008), 2141–2170.
[13] Christopher Grigoriou. 2019. Revenue maximisation versus trade facilitation: the contribution of automated risk management. *World Customs Journal* 13 (2019).
[14] Xinran He, Junfeng Pan, Ou Jin, Tianbing Xu, Bo Liu, Tao Xu, Yanxin Shi, Antoine Atallah, Ralf Herbrich, Stuart Bowers, et al. 2014. Practical lessons from predicting clicks on ads at facebook. In *ADKDD*. 1–9.
[15] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).
[16] Maria Krivko. 2010. A hybrid model for plastic card fraud detection systems. *Expert Systems with Applications* 37, 8 (2010), 6070–6076.
[17] Yiğit Kültür and Mehmet Ufuk Çağlayan. 2017. Hybrid approaches for detecting credit card fraud. *Expert Systems* 34, 2 (2017), e12191.
[18] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. 2008. Isolation forest. In *ICDM*. 413–422.
[19] Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. 2019. On the variance of the adaptive learning rate and beyond. *arXiv preprint arXiv:1908.03265* (2019).
[20] Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2017. Adversarial multi-task learning for text classification. *arXiv preprint arXiv:1704.05742* (2017).
[21] Andrea Dal Pozzolo, Giacomo Boracchi, Olivier Caelen, Cesare Alippi, and Gianluca Bontempi. 2018. Credit Card Fraud Detection: A Realistic Modeling and a Novel Learning Strategy. *IEEE Transactions on Neural Networks and Learning Systems* 29, 8 (2018), 3784–3797.
[22] Muhamad Ridwan Tri Prabowo, Siti Nuryanah, and Sardar M.N. Islam. 2019. The implementation of Benford's law testing method to detect fraud in customs audit. In *Proceedings of the 33rd International Business Information Management Association Conference*. 9329–9339.
[23] Ram Hari Regmi and Arun K. Timalsina. 2018. Risk Management in customs using Deep Neural Network. In *IEEE International Conference on Computing, Communication and Security*. 133–137.
[24] Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098* (2017).
[25] Ron Triepels, Hennie Daniels, and Ad Feelders. 2018. Data-driven fraud detection in international shipping. *Expert Systems with Applications* 99 (2018), 193–202.
[26] Jellis Vanhoeyveld, David Martens, and Bruno Peeters. 2019. Customs fraud detection: Assessing the value of behavioural and high-cardinality data under the imbalanced learning issue. *Pattern Analysis and Applications* (2019).
[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*. 5998–6008.
[28] Xiang Wang, Xiangnan He, Fuli Feng, Liqiang Nie, and Tat-Seng Chua. 2018. TEM: Tree-enhanced embedding model for explainable recommendation. In *WWW*. 1543–1552.
[29] Jarrod West and Maumita Bhattacharya. 2016. Intelligent financial fraud detection: A comprehensive review. *Computers & Security* 57 (2016), 47–66.
[30] Michael Zhang, James Lucas, Jimmy Ba, and Geoffrey E Hinton. 2019. Lookahead Optimizer: k steps forward, 1 step back. In *NeurIPS*. 9593–9604.
[31] Xin Zhou. 2019. Data mining in customs risk detection with cost-sensitive classification. *World Customs Journal* 13 (2019).

SUPPLEMENTARY MATERIAL

7.1 Code and Data Availability

We have uploaded source code for DATE at http://bit.ly/DATE-Fraud-Detection. Unfortunately, the import transaction data used in the paper cannot be made public due to non-disclosure agreements. However, in line with this study, the World Customs Organization launched a collaborative research project (BACUDA; Band of Customs Data Analysts) to encourage member countries to use big data analytics, and we are working on developing virtual customs data for research purposes. When it is prepared and approved, we plan to release the data through the repository and launch a competition through platforms. We expect the community to conduct visible research through realistic data and publish models beneficial to the public sector.

7.2 Hyperparameter Analysis

We analyze the performance of the model by varying hyper-parameter values of each module, such as the number of trees τ (Eq. 1) and the *depth* of the first-stage gradient boosting decision trees (GBDT), embedding dimension d of cross features $\mathcal{F}_i$, leaves importer ID p_u and goods identifier q_h (Eq. 5) from the second-stage attention network. Due to lack of space, we demonstrate the performance patterns by changing representative parameters and fixing other parameters with default settings mentioned in Section 5.1.3. Among various metrics, we report Precision@n% and Rev@n%, since the results from other metrics show similar patterns. The results are averaged over *ten* repeated experiments.

7.2.1 Analysis on Tree Number. Figure 5 shows the effect of changing the number of trees τ. Note that we report the performance of DATE, not the performance of GBDT. From the result, We can confirm that the default value $\tau = 100$ is well set for training DATE. In all cases, the smaller number of trees $\tau \in \{25, 50\}$ in the first stage showed lower performance than the default setting $\tau = 100$. For some cases, $\tau = 200$ shows better performance. But, the performance drops by having more trees, $\tau = 400$. We can conclude that DATE needs a sufficient number of trees to provide adequate information for subsequent attention mechanisms.

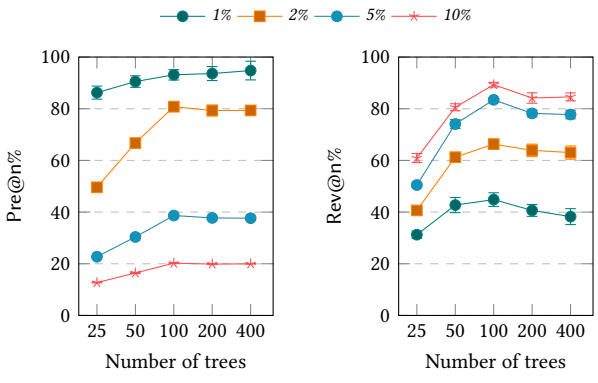

Figure 5: Performance difference by tree number of GBDT.

7.2.2 Analysis on Tree Depth. Figure 6 shows the effect of changing the tree depth. Unlike previous experiment, the default parameter *depth* = 4 was not the best parameter. From the result, we can see that DATE with simpler tree with *depth* $\in \{2, 3\}$ performs better than the DATE with *depth* = 4. Tree with additional complexity, *depth* $\in \{5, 6\}$ does not always guarantee additional performance, except for Precision@2%. One of the reasons that DATE does not get benefit from tree depth is the way in which cross features obtained from the tree are utilized. Even if the decision rule obtained from the leaf is complex, the embedded value would not be complex in an explicit manner since the dimension of dense representation is fixed. We can interpret that embeddings from concise rules make a synergy to train the following attention mechanism.

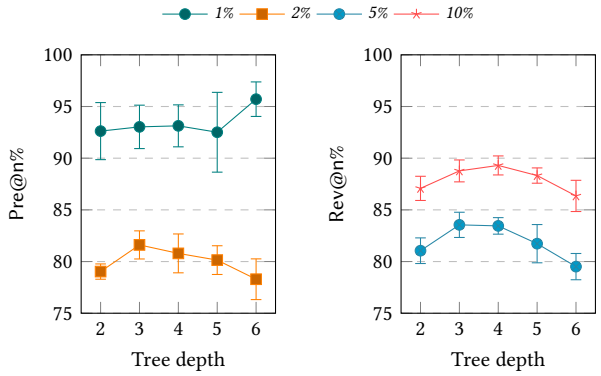

Figure 6: Performance difference by tree depth of GBDT.

7.2.3 Analysis on Embedding Size. Figure 7 shows the effect of changing dimensions d of cross features, importer ID, and HS codes. Due to high cardinality of importer id (= 165,305) and HS codes (= 4,704) variables, the embedding size is shown to be at least 16 to perform well. However, the performance improvement is marginal for $d > 16$. Since the computation cost of attention mechanisms increases polynomially by increasing the embedding size, we can confirm that $d = 16$ is a good setting to ensure both effectiveness and efficiency.

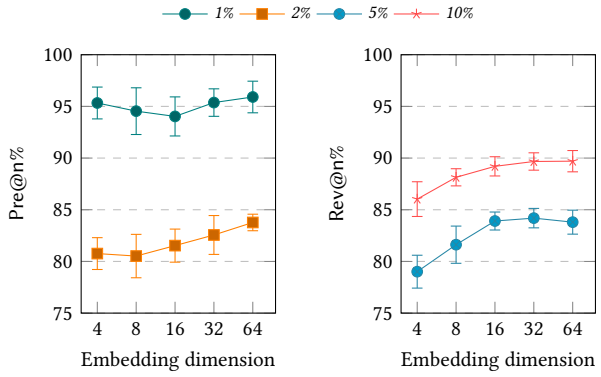

Figure 7: Performance difference by embedding size.

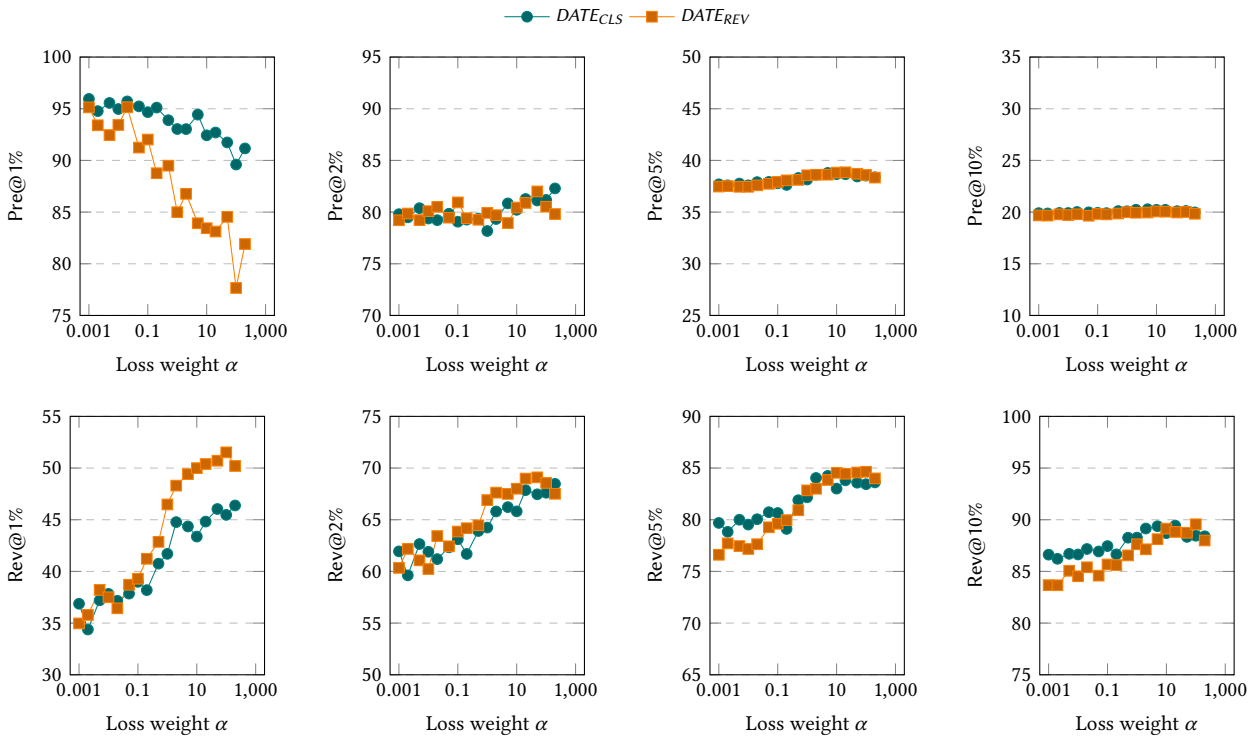

Figure 8: Performance difference between DATE_{CLS} and DATE_{REV} by controlling loss weights α.

7.3 DATE_{CLS} and DATE_{REV} by controlling α

We measure the performance difference between DATE_{CLS} and DATE_{REV} by controlling the weight α between two sub-losses $\mathcal{L}_{cls}$ and $\mathcal{L}_{rev}$ (Eq. 8). As described in Section. 5.2, DATE learns from dual-task optimization and the predicted values $\hat{y_i}^{cls}$ and $\hat{y_i}^{rev}$ can be used to select top-n% of transactions, we named these inspection approaches as DATE_{CLS} and DATE_{REV}, respectively. Since the final objective function $\mathcal{L}_{DATE}$ can be controlled by the weight α,

$$\mathcal{L}_{DATE} = \mathcal{L}_{cls} + \alpha \mathcal{L}_{rev} + \lambda \|\Theta\|^2, \tag{10}$$

one can raise a question about how the DATE_{CLS} and DATE_{REV} results change according to the parameter α. For instance, DATE_{REV} should have high Rev@n% with large α, or small α should guarantee higher Pre@n% for DATE_{CLS}. For this experiment, we use the parameter α in the range of $\alpha \in \{0.001, 0.002, 0.005, \ldots, 200\}$, where the default parameter value was 10.

Figure 8 shows the effect of varying changing α. Notably, Precision@1% and Revenue@1% are affected the most among $n \in \{1, 2, 5, 10\}$. As in the leftmost figures, the ranges of Precision@1% and Revenue@1% are over 15%. For all inspection rates, the larger the α, the higher the Revenue@n% as expected. But, the Precision@n% does not always follow the estimated behavior except Precision@1%. Note that Revenue@1% of DATE_{REV} shows 51.5% with $\alpha = 100$, which is 2% higher value than the result with $\alpha = 10$, which was reported in Table 3. The results also show that the performance of DATE_{REV} changes more drastically than DATE_{CLS}. In accordance with the customs policy and the evaluation criteria, practitioners should control α carefully and use two outcomes $\hat{y_i}^{cls}$ harmoniously for customs inspection.