TOWARDS UNDERSTANDING THE ROBUSTNESS AGAINST EVASION ATTACK ON CATEGORICAL INPUTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Characterizing and assessing the adversarial risk of a classifier with categorical inputs has been a practically important yet rarely explored research problem, especially in safety-critical applications. Conventional wisdom attributes the difficulty of solving this problem for a general classifier to its combinatorial nature and adopts domain-specific heuristics to circumvent the challenge in a few applications. Our study shows that the adversarial robustness assessment can be obtained with provably optimality guarantees for Lipschitz-continuous classifiers. We further unveil theoretically the impact factors of adversarial vulnerability of a targeted classifier with categorical inputs via an information-theoretic adversarial risk analysis. Corroborating these theoretical findings with a substantial experimental study over various real-world categorical data sources, we assess and confirm empirically the impact of the key adversarial risk factors over a targeted learning system with categorical inputs.

1 INTRODUCTION

Categorical data pervasively exist in real-world safety-critical Machine-Learning-as-a-Service (MLaaS) applications, such as ML-driven intrusion detection and digital healthcare. The vulnerability to attacks by intentionally crafting categorical signatures raises concerns on trust and utility of the ML-based analytic services. Characterizing and assessing adversarial robustness on categorical data can thus help evaluate the reliability of the core ML models and flag potential evading efforts.

For a classifier f with categorical inputs, we formulate the adversarial risk of f under evasion attack:

$$\mathcal{R}_{\varepsilon}^{adv} = \sup_{|diff(x,\hat{x})| \leq \varepsilon} P[f(x) \neq f(\hat{x})]. \tag{1}$$

where x and $\hat{x}$ are an unperturbed data instance and its adversarially modified opponent respectively. $|diff(x,\hat{x})|$ denotes the number of modified categorical features and ε is the attack budget. From the defender/learner's perspective, a classifier f is more robust if it can tolerate a larger number of feature modifications without inducing miss-classification. Unlike continuous measurements, a categorical attribute only takes one particular category value among others. Individual or a combination of multiple categorical attributes can therefore carry strong physical significance and present highly structural and semantic information. Evasion attack manipulating categorical inputs is in nature **an NP-hard knapsack problem**. The discontinuity of categorical input thus raises fundamental yet rarely addressed questions to evaluate the adversarial vulnerability of models with categorical inputs:

- **Q1** What are the key factors characterizing f's adversarial risk $\mathcal{R}_{\varepsilon}^{adv}$ on categorical data ?
- **Q2** Regardless whether it is a binary or multi-class scenario, for a general classifier f, can we assess the adversarial risk of f with categorical inputs with provably accuracy guarantee ?

Both questions rest open despite recent efforts of adversarial vulnerability exploration with discrete data. First, the discontinuity of categorical space prevents the direct use of the previous progress on adversarial risk analysis with continuous data. The local subspace assumption of l_p-bounded adversarial attacks does not apply to the categorical features (Hein & Andriushchenko, 2017; Wang et al., 2018; Fawzi et al., 2016; Gilmer et al., 2018; Yin et al., 2019; Khim & Loh, 2018; Tu et al., 2019). Second, current model-specific attack analysis strategies have low generalizability. Graph Neural Networks were analyzed by building differentiable surrogates to the knapsack objective function (Bojchevski & Günnemann, 2019; Bojchevski & Günnemann, 2019; Zugner & Gunnemann, 2019). Image classifiers were attacked

by narrowing down the range of greedy search over pixels with highly domain-specific heuristics and empirical knowledge (Narodytska & Kasiviswanathan, 2017; Croce & Hein, 2019), e.g., limiting the search range within the local image areas that are empirically sensitive. Restricted Deep neural networks (DNN) with non-negative weights were attacked by greedy search in the original categorical space (Qi et al., 2019; Wang et al., 2020). Although the greedy search strategy can be employed regardless of the architecture of the classifier, a provably guarantee to the success of the attack is yet to establish, as applying the greedy search to a general knapsack problem can produce arbitrarily bad solutions.

We target to address the above-mentioned two questions. We demonstrate qualitatively an information-theoretic characterization of adversarial vulnerability of a classifier. Our analysis unveils that *the informativeness of the input categorical instance*, *the sensitivity of the perturbed categorical features* and *the information geometry property of the targeted classifier* are the three driving forces jointly determining adversarial vulnerability of the classifier.

Our study also shows that evasion attack against *any measurable binary / multi-class classifier with categorical inputs* can be cast to a weakly submodular maximization problem, with a mild smoothness condition. It can thus be conducted using a simple yet efficient greedy exploration strategy with provable approximation guarantees. *The theoretical findings not only explain the empirical success of using the greedy search strategy to generate adversarial textual and image samples (Gong et al., 2018; Yang et al., 2018; Narodytska & Kasiviswanathan, 2017), but also pave the way to a domain-independent adversarial robustness assessment on categorical data with provably precision*.

We instantiate the problem of adversarial risk characterization and assessment with a widely used DNN model, i.e. *Long Short-Term Memory* (LSTM) and three real-world multi-class categorical datasets in Section.4. With the proposed assessment method, we measure the variation of adversarial robustness of a targeted classifier associated with each of the identified risk factors. The results confirm the impact of these factors in encouraging/suppressing adversarial vulnerability. In addition, we organize an empirical arms race against a variety of defense methods, namely, *adversarial training*, *model regularization* and *random smoothing*, which are popularly used on continuous data. We show that not of all them can produce as effective defense in the categorical space as in the continuous space. It implies that tailoring defense methods for the learning systems with categorical inputs is not unnecessary effort and needs to be addressed.

2 Related Work

Tremendous efforts have been made to *vulnerability measurement of a classifier under evasion attack* (Hein & Andriushchenko, 2017; Wang et al., 2018; Fawzi et al., 2016; Gilmer et al., 2018; Weng et al., 2019; Sinha et al., 2018; Cohen et al., 2019; Shi et al., 2020; Yin et al., 2019; Khim & Loh, 2018; Tu et al., 2019). They formulate the adversarial risk bound and evaluate the stability of the targeted classifier within an l_p-ball centered at a given input. All these works focus on continuous input with an essential assumption: *the local neighborhood around the input instance form a smooth manifold*. Nevertheless, this assumption does not hold in a categorical feature space.

Pioneering works of evasion attacks with categorical inputs depend on domain-specific knowledge to facilitate the attack exploration. Kuleshov et al. (2018); Papernot et al. (2016); Miyato et al. (2016); Samanta & Mehta (2017); Gao et al. (2018); Yang et al. (2018); Gong et al. (2018); Ebrahimi et al. (2018) focus on replacing individual words or phrases to cheat text classifiers. These methods use heuristic semantic rules, e.g., replacing words with manually defined candidate synonyms and constraining the word change to preserve readability and semantic integrity. Narodytska & Kasiviswanathan (2017); Croce & Hein (2019) treat image pixel intensities as categorical features and use heuristic-search based methods to conduct L_0 attacks. They initiate the candidate pixels to perturb by measuring pixel-wise sensitivity in the classification task. In each iteration, they constraint the search range within the local neighbors of the pixels modified in the previous rounds. Despite of the sounding empirical results, there is no guarantee on a successful attack within the attack budget.

Bojchevski & Günnemann (2019); Bojchevski & Günnemann (2019); D.Zugner et al. (2019); D. Zügner & Günnemann (2018); Akbarnejad & Günnemann (2019) adopt edge-flipping and node attribute perturbation to poison graph data mining pipelines, e.g., graph neural networks and graph embedding models. The key idea is to introduce relaxed surrogate functions to the combinatorial attack objective and then solve the relaxed optimization problem instead. Notably, D.Zugner et al. (2019); D. Zügner

& Günnemann (2018); Akbarnejad & Günnemann (2019) define the attack objective as a sum of smallest eigenvalues of the adjacency matrix of a given graph. Though it is not explicitly claimed, it is intrinsically a submodular maximization problem.

The greedy search strategy can be employed regardless of the architecture of the classifier, which makes it a proper candidate for a generally applicable adversarial risk assessment. Notably, Qi et al. (2019); Wang et al. (2020) reveal that simple greedy search can be used to address the attack problem against deep neural networks (DNN) without the help of domain-specific heuristics, if the attack objective is close to be strictly submodular. Nevertheless, the parameters of the targeted DNN model are required to be non-negative to guarantee the submodularity, which restricts flexibility of the classifier and cause significant loss of classification accuracy. Besides, though the attack method proposed in Qi et al. (2019) also uses gradients to guide the iterative greedy search over categorical inputs, the search is restricted within the candidate features with the strongest gradient magnitude, while ignoring the combinatorial patterns composed by the candidate features and the features modified in previous iterations. It inevitably causes suboptimality of the solution.

3 Robustness Characterization and Assessment

We assume an input instance $x = \{x_1, x_2, x_3, \ldots, x_n\}$ of n categorical attributes. Each x_i takes any of m ($m \geq 1$) categorical values. The classifier f outputs decision probabilities f_{y_k} ($k = 1,2,3,\ldots,K$) with respect to different class labels. In practices, each categorical value of x_i^j is cast to a D-dimensional pre-trained embedding vector, e.g., $e_i^j \in \mathbb{R}^D, j = 1,2,\ldots,m$. To represent an instance x with the embedding vectors of its category values, we define binary variables $b = \{b_i^j\}$, i=1,2,...,n, j=1,2,...,m, where $b_i^j = 1$ when the j-th attribute value is present for x_i and $b_i^j = 0$ otherwise. An instance x can then be represented by an $\mathbb{R}^{n*m*D}$ tensor with $x_{\{i,j,:\}} = b_i^j e_i^j$. Let $\hat{b} = \{\hat{b}_i^j\}$ indicate the adversarial modifications introduced into b. For a perturbed $\hat{x}$, its $\hat{b} \neq b$. Depending on the type of attacks to implement, e.g., *insertion*, *deletion* or *substitution*, $\hat{b}$ differs from b in different ways. Without loss of generality, let y_K denote the true class label of x and all the other y_k ($k = \{1,\ldots,K-1\}$) are the potential targets of an evasion attack. The goal of attack is to make $f_{y_K}(x,\hat{b})$ as low as possible and increase $f_{y_k}(x,\hat{b})$ of a fixed k (*targeted attack*) or any k of $\{1,\ldots,K-1\}$ (*non-targeted attack*) as high as possible simultaneously. We focus on *non-targeted evasion attack*, and leave the targeted scenario for future study.

3.1 Information-theoretic Characterization of Adversarial Vulnerability

Theorem 1 *Let S be the training set for building classifier f, and $I(x;y,S)$ denotes the mutual information between the feature x and the pair of label y and training set S. By adsorbing the constant terms in the attack process into const, the lower bound of $\mathcal{R}_\varepsilon^{adv}$ gives:*

$$\mathcal{R}_\varepsilon^{adv} \geq 1 - \frac{2I(x;y,S) - 2\eta_x - I(f_y;S)/2 + const}{\log K} \tag{2}$$

$$\eta_x = \sup_{|diff(b,\hat{b})| \leq \varepsilon} I(x;y,S) - I(\hat{x};y,S) \tag{3}$$

where $diff(b,\hat{b})$ indicates the set of the categorical attributes modified in the attack, and η_x is the supremum of the difference between the mutual information before and after adversarial perturbation, noted by $I(x;y,S)$ and $I(\hat{x};y,S)$ respectively. $I(f_y;S)$ denotes the mutual information between S and f.

Theorem 1 unveils the three impact factors jointly boosting adversarial vulnerability of the targeted classifier f for answering Q1 in Section 1. *In Eq.2, a lower mutual information $I(x;y,S)$ indicates higher adversarial risk*, i.e., $\mathcal{R}_\varepsilon^{adv}$ has a higher lower bound. A lower $I(x;y,S)$ denotes weaker consistency between the input instance (x,y) and the training set S. The classifier trained by S thus produces weaker decision confidence for such (x,y). With (x,y) dropped to the ambiguous zone near the classification boundary, the classifier is more prone to adversarial perturbation.

A higher $I(f_y;S)$ leads to a higher adversarial risk according to Eq.2. The mutual information $I(f_y;S)$ reflects the dependence between the classifier f and the training set S, which can be considered as a lower bound of the VC-dimension for countable hypothesis space $f \in \mathcal{H}$ (Xu & Raginsky, 2017; Zhu et al., 2020). A higher $I(f_y;S)$ thus denotes that f is more likely to suffer from being overfitted to S.

Model overfiting is one of the causes of adversarial vulnerability (Tu et al., 2019). Resonating with the association between $I(f_y;S)$ and adversarial risk, three robustness enhancement strategies popularly used with continuous data can potentially harden the targeted classifier with categorical inputs. **1) Adversarial training** (Miyato et al., 2016; Sinha et al., 2018; Florian et al., 2018; Wang et al., 2019; Shafahi et al., 2019). The adversarially retrained classifier $\hat{f}$ is less correlated with the original training set S compared to f, which reduces $I(f_y;S)$ and may help mitigate the adversarial threat. **2)** The addition of **nuclear norm regularization** over the classifier's parameters in the training process (Ravi et al., 2019; Tu et al., 2019). The resultant regularized classifier has a controlled model complexity, which can potentially reduce the adversarial risk. **3) Random smoothing** (Lee et al., 2019; Levine & Feizi, 2020; Dvijotham et al., 2020; Boj. et al., 2020). Following Cohen et al. (2019), this defense method randomly selects and flips the input categorical features of the targeted classifier. The randomly perturbed classification output is less correlated with the training data S's distribution, which may reduce adversarial risk.

A higher η_x in Eq.3 indicates a higher loss of predicative information by changing x to $\hat{x}$, and thus a higher adversarial risk. In this sense, the vulnerability of the classifier depends on the sensitivity of the perturbed categorical features to adversarial changes.

3.2 Scoring Robustness via Set Function Optimization

We formulate the assessment of adversarial robustness on categorical data as *a set function maximization problem*: we aim at finding a minimal set of categorical feature perturbations $l = diff(b,\hat{b})$, with which the maximum gap between the classification confidence on any wrong label k and the correct label K is larger than a predefined threshold Γ. The size of the minimal perturbation set l is used to assessment adversarial robustness of f over the input instance. Within the perturbation budget ε, a smaller/larger $|l|$ provoking the alert indicates that the classifier's output on the input x is more/less vulnerable w.r.t. adversarial perturbation in the categorical feature space. We consider both the pessimistic (in Eq. 4) and optimistic scenario (in Eq. 5) from the defender's perspective for robustness scoring.

$$\psi^{pess}(l) = \max_{\hat{b}_{i=1\ldots n},\, l=diff(b,\hat{b})} \; \max_{\hat{b}_i^{j=1\ldots m}} (m_f) \qquad s.t. \quad m_f \geq \Gamma,\, |l| \leq \varepsilon \tag{4}$$

$$\psi^{optim}(l) = \min_{\hat{b}_{i=1\ldots n},\, l=diff(b,\hat{b})} \; \max_{\hat{b}_i^{j=1\ldots m}} (m_f) \qquad s.t. \quad m_f \geq \Gamma,\, |l| \leq \varepsilon \tag{5}$$

where $m_f = \max_{k\in\{1,\ldots,K-1\}} \{f_{y_k}(x,\hat{b})\} - f_{y_K}(x,\hat{b})$ is the gap. $m_f \in [-1,0)$ and $m_f \geq 0$ correspond to correct and wrong classification respectively. Γ is a threshold reflecting different levels of tolerance to the perturbation of the classifier's output. $m_f \geq \Gamma$ then triggers an alert of potential adversarial attack.Safety-critical domains, such as cyber security, prefer a lower Γ to provoke alerts even with small yet suspicious input perturbations.

In the pessimistic scenario, Eq. 4 is a bi-level optimization problem. The inner level seeks the optimal change of the category value of a given categorical feature, in order to produce maximum possible perturbation on the classifier's output. The outer level finds the optimal combination of different categorical features for maximizing m_f within the attack budget. $|l|$ is thus a **pessimistic** robustness assessment with respect to **a knowledgeable adversary**. The adversary can locate the most effective modification for both individual categorical features and the combination set of the categorical features. In contrast, $|l|$ to Eq. 5 reflects an **optimistic** robustness assessment with respect to **an oblivious adversary**. In practices, an adversary can be limited by the knowledge about the targeted classifier and/or the query budget. Such an oblivious adversary can only locate the effective category values to attack for a given categorical feature, while can't find the optimal feature combinations to solve the outer-level optimization.

3.3 Provably Accurate Robustness Assessment via Weakly Submodular Maximization

We give in Definition.1 the smoothness condition for a general classifier to meet to approach computationally tractable robustness assessment. Theorem 2 and Theorem 3 then elaborate the association between the smoothness condition met by the classifier and the solvability of the robustness assessment for the classifier, answering the Q2 in Section 1.

Definition 1 ***Smoothness Condition of*** f**.** *Let $\Omega = (p,q)$, $p,q \in \mathbb{R}^n$ and f: $\mathbb{R}^n \rightarrow \mathbb{R}$ be a Lipschitz-continuous and differentiable function. A function f is (m_Ω, M_Ω)-smooth on Ω, if for any $(p,q) \in \Omega$,*

$m_\Omega \in \mathbb{R}$ and $M_\Omega \in \mathbb{R}^+$, $\epsilon = f(q) - f(p) - \langle \nabla f(p), q-p \rangle$ satisfies:

$$\frac{m_\Omega}{2}\|q-p\|_2^2 \le |\epsilon| \le \frac{M_\Omega}{2}\|q-p\|_2^2. \tag{6}$$

Different from (Elenberg et al., 2018), Eq.6 controls only the fluctuation of the function value by varying the input from p to q and vice versa, regardless of convexity or concavity of f. It allows broader choices of the classifier's architecture. With γ-weak submodularity (Elenberg et al., 2018; Santiago & Yoshida, 2020) defined in Appendix B, we establish Theorem 2.

Theorem 2 *Let $\Omega_\zeta = \{(\hat{b},\hat{b}') : |diff(b,\hat{b})| \le \zeta, |diff(b,\hat{b}')| \le \zeta, |diff(\hat{b},\hat{b}')| \le \zeta, \zeta \ge 1\}$, where $\hat{b}$ and $\hat{b}'$ denote two sets of attribute changes. If the classifier $f_{y_k}(x)$ (k=1,...,K) follows the regularity condition given by $(m_{k,\Omega_\zeta}, M_{k,\Omega_\zeta})$-smoothness constraint on Ω_ζ,* ***the pessimistic and optimistic robustness assessment*** *(with Eq.4 and Eq.5) can be formulated respectively as* ***monotone and non-monotone γ_ζ-weakly submodular maximization****.*

Let $\epsilon_k = f_{y_k}(x,\hat{b}') - f_{y_k}(x,\hat{b}) - \langle \nabla f_{y_k}(x,\hat{b}), \hat{b}' - \hat{b} \rangle$, and $\nabla f_y(x,b)_\nu$ denote the elements of $\nabla f_y(x,b)$ corresponding to the difference between $\hat{b}$ and $\hat{b}'$, where $\nu = diff(\hat{b},\hat{b}')$. The submodularity ratio γ_ζ^{pess} for the pessimistic robustness assessment is bounded as:

$$\gamma_\zeta^{pess} = \min_{k=1,\dots,K} \{\gamma_{k,\zeta}^{pess}\} \tag{7}$$

where $\gamma_{k,\zeta}^{pess} \ge \frac{\|\nabla f_{y_k}(x,b)_\nu\|_2 + m_{k,\Omega_1}|\zeta|/2}{\|\nabla f_{y_k}(x,b)_\nu\|_2 + M_{k,\Omega_\zeta}|\zeta|/2}$ for $\epsilon_k \ge 0, k < K$, $\gamma_{k,\zeta}^{pess} \ge \frac{2m_{k,\Omega_\zeta}}{\|\nabla f_{y_k}(x,b)_\nu\|_2^2}(\|\nabla f_{y_k}(x,b)_\nu\|_2 - M_{k,\Omega_1}|\zeta|/2)$ for $\epsilon_k < 0, k < K$, $\gamma_{K,\zeta}^{pess} \ge \frac{2m_{K,\Omega_\zeta}}{\|\nabla f_{y_K}(x,b)_\nu\|_2^2}(\|\nabla f_{y=K}(x,b)_\nu\|_2 - M_{K,\Omega_1}|\zeta|/2)$ for $\epsilon_K \ge 0$, and $\gamma_{K,\zeta}^{pess} \ge \frac{\|\nabla f_{y_K}(x,b)_\nu\|_2 + m_{K,\Omega_1}|\zeta|/2}{\|\nabla f_{y_K}(x,b)_\nu\|_2 + M_{K,\Omega_\zeta}|\zeta|/2}$ for $\epsilon_K < 0$.

Similarly, the submodularity ratio γ_ζ^{optim} of the optimistic assessment is given as:

$$\gamma_\zeta^{optim} = \min_{k=1,,\dots,K} \{\gamma_{k,\zeta}^{optim}\} \tag{8}$$

where $\gamma_{k,\zeta}^{optim} \le \frac{\|\nabla f_{y_k}(x,b)_\nu\|_2^2}{2M_{k,\Omega_\zeta}(\|\nabla f_{y_k}(x,b)_\nu\|_2 - m_{k,\Omega_1}|\zeta|/2)}$ for $\epsilon_k \ge 0$, $k < K$, $\gamma_{k,\zeta}^{optim} \le \frac{\|\nabla f_{y_k}(x,b)_\nu\|_2 + m_{k,\Omega_\zeta}|\zeta|/2}{\|\nabla f_{y_k}(x,b)_\nu\|_2 + M_{k,\Omega_1}|\zeta|/2}$ for $\epsilon_k < 0, k < K$, $\gamma_{K,\zeta}^{optim} \le \frac{\|\nabla f_{y_K}(x,b)_\nu\|_2 + m_{K,\Omega_\zeta}|\zeta|/2}{\|\nabla f_{y_K}(x,b)_\nu\|_2 + M_{K,\Omega_1}|\zeta|/2}$ for $\epsilon_K \ge 0$, and $\gamma_{K,\zeta}^{optim} \le \frac{\|\nabla f_{y_K}(x,b)_\nu\|_2^2}{2M_{K,\Omega_\zeta}(\|\nabla f_{y_K}(x,b)_\nu\|_2 - m_{K,\Omega_1}|\zeta|/2)}$ for $\epsilon_K < 0$.

Enjoying the weak submodularity, the pessimistic and optimistic assessment can be derived by greedy search with provable accuracy. We present a *Forward Stepwise Greedy Search* (FSGS) algorithm for deriving the pessimistic assessment (Eq.4) in Algorithm 1. Due to the non-monotone objective function of the optimistic assessment (Eq.5), a stochastic variant of the standard *FSGS*, namely *Randomized FSGS* (RandGS, in Algorithm 2) is adopted, following the suggestion in (Buchbinder et al., 2014). Compared to *FSGS*, the difference in *RandGS* is to **randomly select one of the top-ranked candidate features** contributing with the largest marginal gain m_f to the objective function. Theorem 3 gives the approximation guarantee of the pessimistic and optimistic robustness assessment using *FSGS* and *RandGS* respectively.

Theorem 3 *For a $(m_{\Omega_\zeta}, M_{\Omega_\zeta})$-smooth classifier f, the approximation quality of the solution l to Eq.4 and Eq.5 is bounded respectively as*

$$\begin{aligned} &|m_f(l)| + e^{-\gamma_\zeta^{pess}} \le (1 - e^{-\gamma_\zeta^{pess}})|m_f(l^*_{pess})| \quad (\textit{Pessimistic Assessment}) \\ &|m_f(l)| \ge \frac{1}{\gamma_\zeta^{optim}} e^{-\gamma_\zeta^{optim}} |m_f(l^*_{optim})| \quad (\textit{Optimistic Assessment}) \end{aligned} \tag{9}$$

where γ_ζ^{pess} and γ_ζ^{optim} are the submodularity ratio in Eq. 7 and Eq. 8, respectively. l^*_{pess} and l^*_{optim} are the true optimal solutions to the two problems.

With Theorem 3, the Q2 in Section.1 is fully addressed. Notably, any classifier with a finite Lipshitz constant meets the smoothness condition. Recent work (Fazlyab et al., 2019) has shown that a large

Algorithm 1 FSGS for pessimistic robustness assessment

Input: The candidate set $H=\{a_i\}_{i=1...n}$ of all modifiable attributes
Output: The maximum support set l_t pushing $m_f(l_t)$ to violate the tolerance constraint in Eq.4

$l_0 \leftarrow \emptyset$
for $t=0,1,2,...,\rho$ **do**
  $T \leftarrow$ *zero-valued vector* $\in \mathbb{R}^n$
  for each $a_i \in H/l_{t-1}$ **do**
    $T(a_i) = \underset{s \subset l_{t-1}}{\text{argmax}} m_f(s \cup a_i)$ (**Inner level Optimization**)
  end for
  $\bar{a} = \underset{a_i}{\text{argmax}} T(a_i)$ (**Outer level Optimization**)
  $l_t \leftarrow l_{t-1} \cup \{\bar{a}\}$
  $m_f(l_t) \leftarrow m_f(l_{t-1} \cup \bar{a})$
  if $m_f(l_t) \geq \Gamma$ **then break**
end for

Algorithm 2 RandGS for the optimistic robustness assessment

Input: The candidate set $H=\{a_i\}_{i=1...n}$ of all modifiable features
Output: The maximum support set l_t pushing $m_f(l_t)$ to violate the tolerance constraint in Eq.5

$l_0 \leftarrow \emptyset$
for $t=0,1,2,...,\rho$ **do**
  $T \leftarrow zero-valuedvector \in \mathbb{R}^n$
  for each $a_i \in H/l_{t-1}$ **do**
    $T(a_i) = \underset{s \subset l_{t-1}}{\text{argmax}} m_f(s \cup a_i)$ (**Inner level Optimization**)
  end for
  $M \subset H/l_{t-1}$ be a subset of size τ
  minimizing $\sum_{a_i \in M} T(a_i)$ (**Outer level Optimization**)
  Let $\bar{a}$ be a uniformly random element from M
  Let $\bar{s}(\bar{a})$ be the subset corresponding to $\bar{a}$
  $l_t \leftarrow l_{t-1} \cup \{\bar{a}\}$
  $m_f(l_t) \leftarrow min\{m_f(\bar{a} \cup \bar{s}(\bar{a})), m_f(l_{t-1})\}$
  if $m_f(l_t) \geq \Gamma$ **then break**
end for

number of neural networks are Lipschitz continuous, such as the fully-connected neural networks with *ReLU*, *sigmoid*, and *tanh* activation functions. The *FSGS* based robustness assessment method can thus be applied to most DNN-based classifiers that are popularly adopted in real-world applications.

Theorem 3 also reveals that the role that *the smoothness of f_y plays in characterizing the solvability of the robustness assessment problem.* As indicated by Eq. 7 and Eq. 8, a smoother f_k with smaller m_{k,Ω_ζ} and M_{k,Ω_ζ} has a higher submodularity ratio ($\gamma^{pess}_{k,\zeta}$ and $\gamma^{optim}_{k,\zeta}$). With a submodularity ratio closer to 1, both *FSGS* and *RandGS* can obtain better approximation quality to the underlying optimal result of the robustness assessment, according to Eq. 9. Comparing Eq. 7 to 8, the submodularity ratio of the pessimistic assessment is higher than that of the optimistic opponent. The greedy search has lower approximation quality under the optimistic setting. The optimistic assessment is thus generally higher than the pessimistic assessment. It echos with the intuition that a classifier is more vulnerable to a knowledgeable adversary than an oblivious one.

3.4 Orthogonal Matching Pursuit Greedy Search for Robustness Assessment

In the inner optimization of Algorithm 1 and 2, both *FSGS* and *RandGS* need to traverse all the combinations of candidate attributes and the subsets of the selected set of attributes. In the worst case, they require $O(m(\sum_{i=1}^{\kappa-1} C_\kappa^i)(|H|-\kappa))$ objective function evaluations in each iteration, where $|H|$ denotes the size of all the candidate attributes to change, $\kappa = |l_t|$, and m is the number of category values of each attribute. Large $|H|$ and m are usually witnessed in the real-world data. Both methods thus are costly in querying the oracle classifier in such real-world cases.

In Wang et al. (2020), an orthogonal matching pursuit guided greedy searching strategy (*OMPGS*) is introduced to address the query bottleneck of primary greedy search in evasion attack on discrete data. In the context of attack, this method computes $\nabla f_y(x)_{\hat{b}}$ (the gradient of f_y w.r.t. the binary indicators $\hat{b}$ of unmodified categorical features). The magnitude of the gradient measures the capability of each candidate feature in reducing the classification confidence of the correct label. In this sense, *OMPGS* narrows down the candidate attributes to those that best correlate with the orthogonal complement of the subset of the attributes already selected in greedy search. The query complexity of *OMPGS* is thus independent of $|H|$ and m. Due to the space limit, the algorithm details about *OMPGS* are explained in Appendix C.

We apply *OMPGS* to speed up the solutions to both the pessimistic and optimistic assessment problems. To derive the optimistic assessment score, we use a randomized variant of *OMPGS*,namely *RandOMPGS*. It first selects a set of candidate categorical attributes. After that, one of the candidate attributes is chosen *randomly* to attack, following the same randomized process in Algorithm 2. The approximation guarantee of *OMPGS* given in (Wang et al., 2020) only explains the provably accuracy of using *OMPGS* in *the pessimistic assessment*. In this work, we further bound the approximation accuracy of *RandOMPGS* to derive *the optimistic assessment* in Theorem 4. Comparing to Theorem 3, *RandOMPGS* achieves a lower but similar approximation accuracy to the underlying optimal solution to the optimistic assessment (Eq. 5).

Theorem 4 *Following the notations in Theorem 2 and Theorem.3, the approximation quality of the optimistic assessment score $|l|$ obtained by RandOMPGS is bounded as in the followings:*

$$
\begin{aligned}
&|m_f(l)| \geq \frac{1}{\psi} e^{-\psi} |m_f(l^*_{optim})|, \psi = \max\{\varphi_k\} (k=1,2,3,...,K) \\
&\varphi_k = \max\{\frac{\|\nabla f_{y_k}(x,b)_\nu\|_2^2}{2M_{\mathbf{k},\Omega_\zeta}(\|\nabla f_{y_k}(x,b)_\nu\|_2 - m_{k,\Omega_1}|\zeta|/2)}, \frac{\|\nabla f_{y_k}(x,b)_\nu\|_2 + m_{k,\Omega_\zeta}|\zeta|/2}{\|\nabla f_{y_k}(x,b)_\nu\|_2 + M_{k,\Omega_1}|\zeta|/2}\}
\end{aligned}
\tag{10}
$$

Like *OMPGS*, the accuracy of *RandOMPGS* also depends on the submodularity ratio of the optimistic robustness scoring problems, which resonates with the association between the smoothness of the

classifier and the approximation accuracy of the derived robustness scores. In both of the pessimistic and optimistic assessment, the gradient magnitude used by *OMPGS* and *RandOMPGS* can be interpreted as an upper bound of the increase of m_f value by adding adversarial perturbation to a previously unmodified feature, as shown in Appendix C.

4 Experimental Study

We instantiate the study with standard LSTM based classifiers trained on three multi-class datasets collected from real-world applications of *Text analysis*, *Cyber security* and *Biomedicine*.

Yelp-5 (**Yelp**)(Asghar, 2016) The Yelp-5 reviews dataset was obtained from the Yelp Dataset Challenge in 2015. We use the reviews containing 650K training and 50K testing samples with the classes from 1 star to 5 stars for learning classifier. Each word is a categorical feature with 300-dim embedding.

Intrusion Prevention System Dataset (IPS) (Wang et al., 2020). The IPS dataset has 242,467 instances, each of which is a 20-step attack sequence and represented by $x \in \mathbb{R}^{20*1103*70}$, because each step can be a categorical value from 1,103 different malicious actions embedded in 70-dim space by word embedding. An x is classified with 3 labels: *attack type 1, attack type 2*, and *others*.

Splice-junction Gene Sequences (Splice) (Noordewier et al., 1991). This dataset has 3,190 samples. Each one is a gene fragment of 60 categorical features with values in {'A', 'G', 'C', 'T', 'N'}. Learning an embedding for each value with 30-dim, one instance is given as $x \in \mathbb{R}^{60*5*30}$. The task is to predict each splice instance as an intro-exon boundary (*IE*), an exon-intro boundary (*EI*), or neither of them.

The LSTM-based classifiers with *ReLu* activation function and dropout achieve **accuracy** scores of *0.61*, *0.92* and *0.95* respectively for *Yelp*, *IPS* and *Splice*. More details about the datasets and the experimental setting are given in Appendix D.1. To strengthen the universality of the robustness assessment framework, we also include additional results on binary-classification datasets using Convolution Neural Networks (CNN) in the appendix.

The experimental study is to address questions below:

1) What's the robustness assessment result of a targeted classifier? (Empirically evaluate the answer in Theorem 2,3 and 4 to Q2 in Section 4.1). For the classifier trained on each dataset, we first find the robustness assessment $|l|$ for each data instance. We report the median and mean value of all the $|l|$ derived on the dataset. A lower/higher median value indicates a weaker/stronger robustness of the classifier to the adversarial attack over the given data set. The tolerance threshold Γ is tested on -0.4 and 0 to assess our proposed assessment method with varied tolerance to adversarial threats in safety-critical applications. Different greedy search methods are compared on their assessment result. Besides *FSGS* and *OMPGS* (for solving Eq.4) and *RandGS* and *RandOMPGS* (for solving Eq.5) , we also include *Gradient-based Attack (GradAttack)* from (Qi et al., 2019) in the evaluation. *GradAttack* tackles only the pessimistic assessment problem. The comparison with *GradAttack* is to verify the merit of *FSGS* and its variants in solving the robustness assessment problem of a general classifier. For *RandGS* and *OMPGS*, we empirically choose 10 candidate features in each iteration of greedy search[1].

2) Can the information-theoretic characterization of adversarial vulnerability in Theorem 1 be verified using the proposed robustness assessment? (Empirically evaluate the answer in Theorem 1 to Q1 in Section 4.2 and 4.3). We verify an increase of risk on $x_{l_{mi}}$ with lower mutual information, i.e., $I(x_{l_{mi}};y,S) < I(x;y,S)$, and verify an increase of robustness by reducing $I(f_y;S)$.

4.1 Robustness Assessment With Greedy Search

Besides showing the median and mean of the assessment score $|l|$ (*Med(Avg)*) of the testing instances, we also report attack success rate (noted as **SR** in the table), i.e., the fraction of the testing instances on which the greedy methods can push m_f greater than the tolerance threshold within the one-hour time limit. With similar median/mean values of $|l|$, a lower **SR** also indicates stronger robustness of the classifier. If the greedy search fails to push m_f to surpass the threshold Γ for all the testing instances within the time limit, we note this situation as *OVF* (an abbreviation for overflow) and **SR** = 0 in the tables. P/O denotes the *pessimistic* or *optimistic* assessment respectively.

[1] Implementations are available at `https://github.com/XYZ211923Y/-RobustXXXXX`.

Table 1: Pessimistic and Optimistic Robustness Assessment $|l|$. A lower/higher assessment score indicates weaker/stronger robustness of the classifier. SR is the success rate of attack. OVF indicates SR=0.

GreedySeearch	P/O	Γ	Yelp		IPS		Splice	
			SR	Med (Avg)	SR	Med(Avg)	SR	Med(Avg)
FSGS/ RandGS	P	-0.4	0.98	1 (1.5)	0.80	1 (1.2)	0.91	1 (1.7)
		0	0.98	1 (1.9)	0.80	1 (1.2)	0.91	1 (1.7)
	O	-0.4	0.09	9 (8.1)	0.33	1 (1.2)	0	OVF
		0	0.09	12 (11.8)	0.33	1 (1.3)	0	OVF
GradAttack	P	-0.4	0.84	1 (1.5)	0.59	1 (1.8)	0.72	3 (3.7)
		0	0.76	2 (1.7)	0.59	1 (1.7)	0.72	3 (3.7)
OMPGS/ RandOMPGS	P	-0.4	0.98	1 (2.0)	0.91	1 (1.6)	0.74	2 (2.1)
		0	0.97	2 (2.5)	0.91	1 (1.8)	0.74	2 (2.3)
	O	-0.4	0.63	7 (6.9)	0.35	3 (3.4)	0	OVF
		0	0.52	8 (7.6)	0.35	3 (3.6)	0	OVF

Table 1 shows several interesting findings. First, for both the pessimistic and optimistic robustness assessment, **a lower tolerance threshold Γ results in lower scores over all the datasets**. This observation is consistent with the intuition of setting the tolerance threshold. A lower tolerance threshold helps choose more worst-case resilient classifiers. Our study allows for the analysis with different vigilance levels. Second, as expected, **the pessimistic robustness assessment scores are always significantly lower than those of the optimistic assessment, and the *SR* value is much lower on the optimistic assessment scenario.** Due to lack of query capability and knowledge about f, it is difficult for the oblivious adversary to locate the sensitive features.

OMPGS and *RandOMPGS* produce similar robustness assessment scores, compared to *FSGS* and *RandGS*. Compared to evaluate each candidate features as in *FSGS*/*RandGS*, *OMPGS* improves the search efficiency using the gradient based heuristic guidance. On *IPS* and *Splice*, *FSGS*/*RandGS* needs 217.65 and 1123.51 seconds on average on each data sample for the pessimistic and optimistic assessment. In contrast, *OMPGS*/*RandOMPGS* costs only 0.24 and 1.59 seconds for the pessimistic and optimistic assessment, more than 700 times faster than *FSGS*/*RandGS*. *GradAttack*, in contrast, produces much lower *SR* than *FSGS* and *OMPGS* in the pessimistic assessment setting. As discussed in Section.2, the greedy search in *GradAttack* ignores the combinatorial effect of the candidate features and the previously modified features, which leads to worse approximation quality and much less efficient attacks compared to *FSGS* and *OMPGS*. The results echo the theoretical analysis. With significantly lower *SR*, *GradAttack* can not provide an accurate answer to the robustness assessment problem.

We conduct one-factor-at-a-time sensitivity analysis on each dataset (Campbell et al., 2008) to measure the sensitivity of features in the classification task. Given a data instance, we change each feature while keeping all the others fixed. The averaged change of the classifier's output over all the testing instances is used as the feature-wise sensitivity measurement. A larger value indicates that the classifier's output is more sensitive to the change over the corresponding feature. We find that the top-sensitive features also appear as the most frequently selected features by *FSGS*,*GradAttack* and *OMPGS*. Details about the sensitivity analysis can be found in Appendix D.3. The interesting overlapping between the attributes which are useful for attack and the top-ranked sensitive attributes confirms our intuition about the association between feature sensitivity and adversarial vulnerability of the classifier.

4.2 Robustness vs Feature Informativeness

According to the discussion in Theorem 1, a lower $I(x;y,S)$ indicates a higher adversarial risk on the input instance (x,y), implying a smaller robustness assessment score. To verify such impact of feature informativeness, we compare the robustness on x and on $x_{l_{mi}}$ that has a lower mutual information, i.e., $I(x_{l_{mi}};y,S) < I(x;y,S)$. The sample $x_{l_{mi}}$ is constructed as those on the half-way of pushing the m_f of the original sample x to surpass Γ. For each dataset, we select the instances requiring at the least two modified features in order to deliver a successful attack. These instances are treated as *original*. There are 125, 414 and 324 *original* instances in *Yelp*, *IPS*, and *Splice* respectively. The results in Table 2 confirm that the classifier becomes less robust to adversarial modifications over $x_{l_{mi}}$ samples in the ambiguous zone, i.e., higher SR and smaller $|l|$ on the $x_{l_{mi}}$ samples than the original samples x.

4.3 Arms Race with Robustness Enhancement Techniques

As discussed in Theorem.1, reducing $I(f_y;S)$, the dependence of a classifier over the training set S, improves the classifier's adversarial robustness. We tune a classifier via *adversarial training* (**AdvC**), *the nuclear-norm based regularization* (**NuR**) following (Ravi et al., 2019) and *random smoothing* by (**RS**) as suggested in (Boj. et al., 2020) to reduce $I(f_y;S)$, and then evaluate if the pessimistic robustness assessment becomes higher, e..g, descrese of SR. To adopt adversarial training, we solve Eq. 4 using

Table 2: Robustness Assessment vs Feature Informativeness, reported at $\Gamma = 0$. $I(x_{l_{mi}}; y, S) < I(x; y, S)$ results in higher adversarial risk (higher SR, smaller $|l|$) on $x_{l_{mi}}$ than original x.

GreedySeearch	P/O	Sample	Yelp		IPS		Splice	
			SR	Med(Avg)	SR	Med(Avg)	SR	Med(Avg)
FSGS/ RandGS	P	original x	0.98	1 (1.9)	0.80	1 (1.2)	0.91	1 (1.7)
		$x_{l_{mi}}$	1.00	1 (1.7)	0.95	1 (1.1)	0.99	1 (1.2)
	O	original x	0.09	12 (11.8)	0.33	1 (1.3)	0	OVF
		$x_{l_{mi}}$	0.94	6 (6.0)	0.65	1 (1.2)	0	OVF
OMPGS/ RandOMPGS	P	original x	0.97	2 (2.5)	0.91	1 (1.8)	0.74	2 (2.3)
		$x_{l_{mi}}$	0.99	2 (2.4)	0.93	1 (1.3)	0.94	1 (1.5)
	O	original x	0.97	4 (4.6)	0.35	3 (3.6)	0	OVF
		$x_{l_{mi}}$	1.00	4 (4.5)	0.57	3 (3.3)	0	OVF

Table 3: Pessimistic Robustness Assessment ($\Gamma = 0$) after Robustness Enhancement by reducing $I(f_y; S)$. $\downarrow^{\mathbf{SR}}$ is the SR decreased amount comparing to the corresponding SR value with $\Gamma = 0$ in Table 1.

Model	Algo.	Yelp			IPS			Splice		
		SR	$\downarrow^{\mathbf{SR}}$	Med(Avg)	SR	$\downarrow^{\mathbf{SR}}$	Med(Avg)	SR	$\downarrow^{\mathbf{SR}}$	Med(Avg)
AdvC	FSGS	0.84	0.13	1 (1.6)	0.74	0.06	1 (1.4)	0.85	0.06	2 (2.1)
	OMPGS	0.83	0.15	2 (2.5)	0.60	0.31	2 (2.2)	0.65	0.09	2 (2.8)
NuR	FSGS	0.72	0.25	1 (1.9)	0.79	0.01	1 (1.1)	1	-0.09	1 (1.4)
	OMPGS	0.68	0.30	2 (2.4)	0.63	0.28	1 (1.0)	0.72	0.02	1 (2.5)
RS	FSGS	0.76	0.21	1 (1.8)	1.00	-0.20	1 (1.4)	0.91	0	1 (1.9)
	OMPGS	0.74	0.24	2 (2.2)	0.95	-0.04	1 (1.7)	0.78	-0.04	2 (2.2)
SRS (Boj. et al., 2020)		1.0	-	2	1.0	-	2	1.0	-	1

OMPGS until the value of m_f drops into the interval $[-0.2, 0.0]$. The resultant modified instances are used for adversarial training. The regularization parameter of the nuclear-norm regularized training method is chosen following (Ravi et al., 2019). The random smoothing is conducted by randomly flipping the values of input categorical features (**RS**) with $p_- = 0.6$ and $p_+ = 0.01$ as suggested in (Boj. et al., 2020). We also implement the original sparsity-aware random smoothing based certifiable robustness (*SRS*) (Boj. et al., 2020) as a reference method of random smoothing based defense techniques. It performs first the same randomly flipping of categorical features as in **RS**. The least number of the features changed to trigger a wrong classification, a.k.a. *certifiable robustness*, is then estimated via a dimension-independent evaluation based on the randomly flipped features. **SRS** (Boj. et al., 2020) is reported to be significantly more efficient and produce tighter robustness assessments than other random smoothing-based methods (Lee et al., 2019; Levine & Feizi, 2020; Dvijotham et al., 2020).

Table 3 illustrates the pessimistic robustness assessment scores of the tuned classifiers with the tolerance threshold $\Gamma = 0$. We skip the results of the optimistic robustness assessment as they present a similar variation tendency. Both adversarial training and nuclear-norm regularization help improve the classifier's robustness, leading to lower *SR* and higher $|l|$ over all the three datasets in most of the cases. However, on *Splice*, the adversarial vulnerability reduction effect is much less visible than those observed on the other two datasets. We ascribe this phenomenon to the fact that the feature-wise sensitivity on *Splice* has a more skewed distribution than the other datasets. A few categorical features in *Splice* are significantly more sensitive than the others in the classification task, corresponding to important proteins dominating the molecules' properties (see feature sensitivity analysis in Section 4.1 and Appendix D.3). The existence of such highly sensitive features makes it difficult to enhance the targeted classifier's robustness without losing the classification utility. *RS* and *SRS* have higher SR than the other robustness enhanced models, showing that the random smoothing technique helps the least improve robustness or even make it more vulnerable to the attack, in the case where a fraction of categorical features are highly sensitive. As seen in Table.1, highly sensitive features can be witnessed in many diversified applications. Firstly, flipping these sensitive features can lead to miss-classification rather than providing robustness. Secondly, if random flipping misses the sensitive features, the adversary can still target at these features to deliver a successful attack.

5 Conclusion

We unveil the key factors that jointly determine the adversarial vulnerability of a classifier with categorical inputs via information-theoretic analysis. The characteristics of the categorical data and the functional property of the targeted classifier act as the external and internal cause of the adversarial threat. We further set a theoretical association between the smoothness of the targeted classifier and the solvability of both pessimistic and optimistic robustness assessment with categorical inputs. Despite of the NP-hard nature of the problem, we show that a provably accurate assessment is approachable for any classifier using simple yet efficient greedy search. Our study is useful for assessing the model robustness before deploying it to real-world safety-critical applications. Our future work will pursue to provide tighter quality bound for assessing adversarial robustness assessment on categorical data.

REFERENCES

A. Akbarnejad and S. Günnemann. Adversarial attacks on node embedding via graph poisoning. In *ICML*, 2019.

Nabiha Asghar. Yelp dataset challenge: Review rating prediction. *arXiv preprint arXiv:1605.05362*, 2016.

A. Boj., J. Klicpera, and S. Günnemann. Efficient robustness certificates for discrete data: Sparsity-aware randomized smoothing for graphs, images and more. In *ICML*, pp. 1003–1013, 2020.

A. Bojchevski and S. Günnemann. Certifiable robustness to graph perturbations. In *NeurIPS*, pp. 8319–8330, 2019.

Aleksandar Bojchevski and Stephan Günnemann. Adversarial attacks on node embeddings via graph poisoning. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 695–704. PMLR, 09–15 Jun 2019. URL `https://proceedings.mlr.press/v97/bojchevski19a.html`.

N. Buchbinder, M. Feldman, J. Naor, and R. Schwartz. Submodular maximization with cardinality constraints. In *SODA*, 2014.

J.E. Campbell, G.R. Carmichael, T Chai, M Mena-Carrasco, Y Tang, DR Blake, N.J. Blake, S.A. Vay, G.J. Collatz, I Baker, et al. Photosynthetic control of atmospheric carbonyl sulfide during the growing season. *Science*, 322(5904):1085–1088, 2008.

J. Cohen, E. Rosenfeld, and Z. Kolter. Certified adversarial robustness via randomized smoothing. In *ICML*, pp. 1310–1320, 2019.

Francesco Croce and Matthias Hein. Sparse and imperceivable adversarial attacks. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4723–4731, Los Alamitos, CA, USA, nov 2019. IEEE Computer Society. doi: 10.1109/ICCV.2019.00482. URL `https://doi.ieeecomputersociety.org/10.1109/ICCV.2019.00482`.

A. Akbarnejad D. Zügner and S. Günnemann. Adversarial attacks on neural networks for graph data. In *KDD*, 2018.

K. D. Dvijotham, J. Hayes, B. Balle, Z. Kolter, C. Qin, A. Gyorgy, K. Xiao, and P. Gowal, S.and Kohli. A framework for robustness certification of smoothed classifiers using f-divergences. In *ICLR*, pp. 656–672, 2020.

D.Zugner, A.Akbarnejad, and S.Gunnemann. Adversarial attacks on neural networks for graph data. In *IJCAI*, 2019.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. HotFlip: White-box adversarial examples for text classification. In *ACL*, 2018.

E.R. Elenberg, R Khanna, A. G. Dimakis, and S. Negahban. Restricted strong convexity implies weak submodularity. *The Annals of Statistics*, 46(6B), 2018.

A. Fawzi, S.M. Moosavi-Dezfooli, and P. Frossard. Robustness of classifiers: From adversarial to random noise. In *NIPS*, pp. 1632–1640, 2016.

M. Fazlyab, A. Robey, H. Hassani, M. Morari, and G. Pappas. Efficient and accurate estimation of lipschitz constants for deep neural networks. In *NeurIPS*, pp. 11423–11434, 2019.

T. Florian, A. Kurakin, N. Papernot, D. Boneh, and P. McDaniel. Ensemble adversarial training: Attacks and defenses. In *ICLR*, 2018.

J. Gao, J. Lanchantin, M. L. Soffa, and Y. Qi. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *SPW*, pp. 50–56, 2018.

J. Gilmer, L. Metz, F. Faghri, S. Schoenholz, M. Raghu, M. Wattenberg, and I.J. Goodfellow. Adversarial spheres. *arXiv preprint arXiv:1801.02774*, 2018. URL http://arxiv.org/abs/1801.02774.

Z. Gong, W. Wang, B. Li, D. Song, and W. Ku. Adversarial texts with gradient methods. *ArXiv*, abs/1801.07175, 2018.

M. Hein and M. Andriushchenko. Formal guarantees on the robustness of a classifier against adversarial manipulation. In *NIPS*, pp. 2266–2276, 2017.

Z.Z. Htike and S. L. Win. Classification of eukaryotic splice-junction genetic sequences using averaged one-dependence estimators with subsumption resolution. *Procedia Computer Science*, 23:36–43, 2013.

J. Khim and P.L. Loh. Adversarial risk bounds for binary classification via function transformation. *arXiv preprint arXiv:1810.09519*, 2018.

V. Kuleshov, S. Thakoor, T. Lau, and S. Ermon. Adversarial examples for natural language classification problems. In *ICLR*, 2018.

G. Lee, Y. Yuan, S. Chang, and T. Jaakkola. Tight certificates of adversarial robustness for randomly smoothed classifiers. In *NeurIPS*, pp. 4910–4921, 2019.

Alexander Levine and Soheil Feizi. Robustness certificates for sparse adversarial attacks by randomized ablation. In *AAAI*, pp. 4585–4593, 2020.

F. Ma, J. Gao, Q. Suo, Q. You, J. Zhou, and A. Zhang. Risk prediction on electronic health records with prior medical knowledge. In *KDD*, 2018.

Chris A Mack. *Elements of Information Theory, 2nd Edition*. Wiley, 2006.

T. Miyato, A. M. Dai, and I.J. Goodfellow. Adversarial training methods for semi-supervised text classification. In *ICLR*, 2016.

Nina Narodytska and Shiva Kasiviswanathan. Simple black-box adversarial attacks on deep neural networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1310–1318, 2017. doi: 10.1109/CVPRW.2017.172.

M.O. Noordewier, G.G. Towell, and J.W. Shavlik. Training knowledge-based neural networks to recognize genes in dna sequences. In *NIPS*, pp. 530–536, 1991.

N. Papernot, P. D. McDaniel, A. Swami, and R. E. Harang. Crafting adversarial input sequences for recurrent neural networks. In *MILCOM*, 2016.

L. Qi, L. Wu, P. Chen, A. Dimakis, I. Dhillon, and M. Witbrock. Discrete attacks and submodular optimization with applications to text classification. In *SysML*, 2019.

S.N. Ravi, T.Dinh, V.S. Lokhande, and V. Singh. Explicitly imposing constraints in deep networks via conditional gradients gives improved generalization and faster convergence. In *AAAI*, 2019.

S. Samanta and S. Mehta. Towards crafting text adversarial samples. *CoRR*, abs/1707.02812, 2017.

R. Santiago and Y. Yoshida. Weakly Submodular Function Maximization Using Local Submodularity Ratio. In *ISAAC*, volume 181, pp. 1–17, 2020.

A. Shafahi, M. Najibi, M. A. Ghiasi, Z. Xu, J. Dickerson, C. Studer, L. S. Davis, G. Taylor, and T. Goldstein. Adversarial training for free! In *NeurIPS*, pp. 3358–3369, 2019.

Z.X. Shi, H. Zhang, K.W. Chang, M. Huang, and C.J. Hsieh. Robustness verification for transformers. *arXiv preprint arXiv:2002.06622*, 2020.

A. Sinha, H. Namkoong, and J. Duchi. Certifiable distributional robustness with principled adversarial training. In *ICLR*, pp. 4910–4921, 2018.

Z. Tu, J. Zhang, and D. Tao. Theoretical analysis of adversarial learning: A minimax approach. In *NeurIPS*, pp. 12259–12269, 2019.

Y. Wang, S. Jha, and K. Chaudhuri. Analyzing the robustness of nearest neighbors to adversarial examples. In *ICML*, pp. 5133–5142, 2018.

Y. Wang, X. Ma, J. Bailey, J. Yi, B. Zhou, and Q. Gu. On the convergence and robustness of adversarial training. In *ICML*, pp. 6586–6595, 2019.

Y. Wang, Y. Han, H. Bao, Y. Shen, F. Ma, J. Li, and X. Zhang. Attackability characterization of adversarial evasion attack on discrete data. In *KDD*, 2020.

T.W. Weng, H. Zhang, H. Chen, Z. Song, C.J. Hsieh, D. Boning, I. S.Dhillion, and L. Daniel. Towards fast computation of certified robustness for relu networks. In *ICML*, pp. 4910–4921, 2019.

Aolin Xu and Maxim Raginsky. Information-theoretic analysis of generalization capability of learning algorithms. In *NIPS*, pp. 2525–2534, 2017.

P. Yang, J. Chen, C. Hsieh, J. Wang, and M. I. Jordan. Greedy attack and gumbel attack: Generating adversarial examples for discrete data. *ArXiv*, abs/1805.12316, 2018.

D. Yin, K. Ramchandran, and P. Bartlett. Rademacher complexity for adversarially robust generalization. In *ICML*, pp. 7085–7094, 2019.

S.C. Zhu, X Zhang, and D Evans. Learning adversarially robust representations via worst-case mutual information maximization. In *ICML*, pp. 11609–11618, 2020.

D. Zugner and S. Gunnemann. Certifiable robustness and robust training for graph convolutional networks. In *KDD*, pp. 246–256, 2019.

A PROOF FOR THEOREM 1

Lemma 1 *Supposing A and B are two independent random variables and U is given by* $P_{UAB} = P_{AB} P_{U|AB}$*:*

$$I(U;A) + I(U;B) \leq 2I(U;A,B) \tag{11}$$

Proof: Given the definition of mutual information $I(X;Y) = H(X) - H(X|Y)$, we need to prove:

$$2H(A,B|U) \leq H(A|U) + H(B|U) \tag{12}$$

With the chain rule of conditional entropy:

$$H(A,B|U) = H(A|U) + H(B|A,U) \tag{13}$$

It indicates that $H(A,B|U) \leq H(A|U)$, as $H(B|U) \geq 0$. Similarly, we can derive $H(A,B|U) \leq H(B|U)$ as well. Combing both inequality relations, Eq.12 holds, which in turn proves Eq.11.

Considering the markov chain in Figure.1, we assume a class label Y and the training set S are statistically independent. f depends on S as $f = \mathcal{A}(S)$, where $\mathcal{A}$ is the training paradigm (could be randomized). The generation of the adversarial example X' depends on both the clean feature vector X and the targeted classifier f. This is consistent with the fact that generating adversarial samples needs to understand the local geometry property of the decision boundary of the targeted classifier. Once we get the perturbed feature vector X', the classification output $Y' = f(X')$.

We can derive the lower bound of the miss-classification probability with the perturbed categorical input x' following Fano inequality:

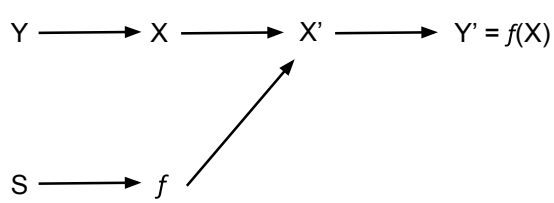

Figure 1: Markov chain of adversarial attack

$$\mathcal{R}_{\varepsilon}^{adv} = \underset{|diff(b,\hat{b} \leq \varepsilon)|}{P} (y' \neq y) \geq 1 - \frac{I(Y';Y,S) + const}{\log K} \tag{14}$$

where K denotes the number of the classes in the learning task. Based on Data Processing Inequality illustrated in Chapter 2 Mack (2006), we can further obtain:

$$\begin{aligned} \mathcal{R}_{\varepsilon}^{adv} &\geq 1 - \frac{I(Y';Y,S) + const}{\log K} \\ &\geq 1 - \frac{I(X';Y,S) + const}{\log K} (I(Y';Y,S) \leq I(X';Y,S)) \\ &= 1 - \frac{2I(X;Y,S) - 2\eta_X - I(X';Y,S) + const}{\log K} \end{aligned} \tag{15}$$

with $\eta_X = I(X;Y,S) - I(X';Y,S)$ denote the mutual information difference between the unperturbed X and the adversarially perturbed data instance X'. By appying Lemma.1 on $I(X';Y,S)$ in Eq.15, we

can obtain

$$\begin{aligned}\mathcal{R}_{\epsilon}^{adv} &\geq 1-\frac{2I(X;Y,S)-2\eta_X-I(X';S)/2}{\log K}\\ &\geq 1-\frac{2I(X;Y,S)-2\eta_X-I(f;S)/2}{\log K}(DataProcessingInequality, I(f;S)\geq I(X';S))\end{aligned} \tag{16}$$

B Proof for theorems on the weak submodularity

We introduce the notion of γ-weak submodularity Elenberg et al. (2018); Santiago & Yoshida (2020) used in Theorem 2.

Definition 2 ***Monotone and γ-weakly submodular function*** *Santiago & Yoshida (2020). A set function $g(S)$ is monotone, if for every $T\subset S$, we have $g(S)\geq g(T)$. Given a set cardinality threshold $k\geq 1$ and a scalar value $\gamma\in[0,1]$. We define γ_k of $g(.)$ w.r.t. a set S as:,*

$$\gamma_k = \min_{S\subseteq H, A:|A|\leq k, S\cap A=\emptyset} \frac{\sum_{a\in A} g(S\cup a)-g(S)}{g(S\cup A)-g(S)} \tag{17}$$

*If $g(S)$ is monotone and $\gamma_k>\gamma$ for any k, $g(S)$ is **a γ-submodular function**. Otherwise, $g(S)$ is **γ-submodular**, if $g(S)$ is non-monotone but $g(S)$ follows for any k:*

$$\sum_{a\in A} g(S\cup a)-g(S)\geq\{\gamma(g(S\cup A)-g(S)), 1/\gamma(g(S\cup A)-g(S))\} \tag{18}$$

It is easy to find, if $g(S)$ is monotone and $\gamma\geq 1$, $g(S)$ is submodular.

Proof of deriving the robustness assessment score under the pessimistic scenario in Eq.7: we first formulate pessimistic scenario as a set function optimization,

$$\begin{aligned}S^* &= \max_S\{\max_{k=1,2,3,...,K-1}\{F_k(S)\}+(G_K(S))\}\\ F_k(S) &= \max_{l\subset S} f_k(b_l)\\ G_K(S) &= \max_{l\subset S}\tilde{f}_K(b_l)\end{aligned} \tag{19}$$

where b_l denotes the modification of categorical features indicated by the index set l. $\tilde{f}_K(b_l)=-f_K(b_l)$. For each $k\in 1,2,3,...,K$, $\epsilon_k=f_k(q)-f_k(p)-\langle\nabla f_k(p), q-p\rangle$. $F_k(S)$ and $G_K(S)$ are monotonically non-decreasing set functions.

Supposing $\epsilon_k\geq 0$ for each $k\in\{1,2,3,...,K-1\}$, we assume further that the features indicated by l' are modified in addition to l, with $|l'|\leq\zeta$. For any b_l, $b_{l\cup l'}$ and $b_{l\cup j}$ denote the additional modification over the features indicated by l' and by j respectively to increase the output of f_k (increasing the miss-classification decision confidence) and $-f_K$ (decreasing the confidence of correct classification). In the following analysis, we relax the discrete indicator b to the continuous domain, as each $b_i\in[0,1]$. $\nabla f_k(b_l)$ denotes the gradient of the classifier function f_k with respect to the variable b at $b=b_l$. Modifying from b_l to $b_{l\cup l'}$ or $b_{l\cup j}$ follows the direction of gradient ascent, which gives $\langle\nabla f_k(b_l), b_{l\cup l'-b_l}\rangle\geq 0$ and $\langle\nabla f_k(b_l), b_{l\cup j-b_j}\rangle\geq 0$, $\langle\nabla\tilde{f}_K(b_l), b_{l\cup l'-b_l}\rangle\geq 0$ and $\langle\nabla\tilde{f}_K(b_l), b_{l\cup j-b_j}\rangle\geq 0$

$$\begin{aligned}&f_k(b_{l\cup l'})-f_k(b_l)\leq\langle\nabla f_k(b_l), b_{l\cup l'}-b_l\rangle+\frac{M_{k,\Omega_\zeta}}{2}\|b_{l\cup l'}-b_l\|_2^2\\ &\leq\|\nabla f_k(b_l)_\zeta\|_2+\frac{M_{k,\Omega_\zeta}}{2}|\zeta|\\ &\sum_{j\in\zeta} f_k(b_{l\cup j})-f_k(b_l)\geq\sum_{j\in\zeta}\langle\nabla f_k(b_l), b_{l\cup j}-b_l\rangle+\frac{m_{k,\Omega_1}}{2}\|b_{l\cup j}-b_l\|_2^2\\ &\geq\|\nabla f_k(b_l)_\zeta\|_2+\frac{m_{k,\Omega_1}}{2}|\zeta|(f_k \text{ isnon-decreasing})\end{aligned} \tag{20}$$

We can derive the lower bound of the submodularity ratio $\gamma_{k,\zeta}$ of f_k:

$$\gamma_{k,\Omega_\zeta}=\frac{\sum_{j\in\zeta} f_k(b_{l\cup j})-f_k(b_l)}{f_k(b_{l\cup l'})-f_k(b_l)}\geq\frac{\|\nabla f_k(b_l)_\zeta\|_2+\frac{m_{k,\Omega_1}}{2}|\zeta|}{\|\nabla f_k(b_l)_\zeta\|_2+\frac{M_{k,\Omega_\zeta}}{2}|\zeta|} \tag{21}$$

Supposing $\epsilon_k < 0$ for each $k \in \{1,2,3,...,K-1\}$, we can derive:

$$\begin{aligned}
&f_k(b_{l\cup l'}) - f_k(b_l) \leq \langle \nabla f_k(b_l), b_{l\cup l'} - b_l \rangle - \frac{m_{k,\Omega_\zeta}}{2} \|b_{l\cup l'} - b_l\|_2^2 \\
&\leq \frac{1}{2m_{k,\Omega_\zeta}} \|\nabla f_k(b_l)_\zeta\|_2^2 \\
&\sum_{j\in\zeta} f_k(b_{l\cup j}) - f_k(b_l) \geq \sum_{j\in\zeta} \{\langle \nabla f_k(b_l), b_{l\cup j} - b_l \rangle - \frac{M_{k,\Omega_1}}{2} \|b_{l\cup j} - b_l\|_2^2\} \\
&\geq \|\nabla f_k(b_l)_{\Omega_\zeta}\|_2 - \frac{M_{k,\Omega_1}}{2} |\zeta| (f_k \, is \, non\text{-}decreasing)
\end{aligned} \tag{22}$$

Given the smoothness assumption on the targeted classifier (See Definition.1), there exits a value of $M_{k,\Omega_1}|\zeta|/2 \leq \|\nabla f_k(b_l)_\zeta\|_2$, which allows that $\|\nabla f_k(b_l)_\zeta\|_2 - \frac{M_{k,\Omega_1}}{2}|\zeta| \geq 0$ holds. Therefore, we can derive the lower bound of the submodularity ratio of F_k and $-G_K$, $\gamma_{k,\zeta}$ (k=1,2,3,...,K):

$$\begin{aligned}
\gamma_{k,\Omega_\zeta} &= \frac{\sum_{j\in\zeta} f_k(b_{l\cup j}) - f_k(b_l)}{f_k(b_{l\cup l'}) - f_k(b_l)} \\
&\geq \frac{2m_{k,\Omega_\zeta}}{\|\nabla f_k(b)_\nu\|_2^2} (\|\nabla f_k(b)_\nu\|_2 - M_{k,\Omega_1}|\zeta|/2)
\end{aligned} \tag{23}$$

The submodularity ratio γ_ζ on Ω_ζ in Eq.19 is $\gamma_\zeta = \min_{k=1,2,3,...,K} \{\gamma_{k,\Omega_\zeta}\}$.

Proof of deriving the robustness assessment score under the optimistic scenario in Eq.8. Following the setting of Eq.19, we can formulate the robustness assessment problem at the optimistic scenario (with an oblivious adversary) :

$$\begin{aligned}
&S^* = \max_S \{G_K(S) + \min_{k=1,2,3,...,K-1} \{\tilde{F}_k(S)\}\} \\
&\tilde{F}_k(S) = \min_{l\subset S} h_k(b_l) \\
&G_K(S) = \min_{l\subset S} f_K(b_l)
\end{aligned} \tag{24}$$

Both $G_k(S)$ and $\tilde{F}_k(S)$ are non-increasing with $h_k(b_l) = -f_k(b_l)$. Eq.24 thus defines a non-monotone set function maximization problem (See Definition.2). We conduct the analysis as in Eq.20 and Eq.22. Modifying b_l to $b_{l\cup l'}$ and $b_{l\cup j}$ follows the direction of gradient descent, which gives $\langle \nabla f_K(b_l), b_{l\cup j} - b_l \rangle \leq 0$ and $\langle \nabla f_K(b_l), b_{l\cup l'} - b_l \rangle \leq 0$, $\langle \nabla h_k(b_l), b_{l\cup j} - b_l \rangle \leq 0$ and $\langle \nabla h_k(b_l), b_{l\cup l'} - b_l \rangle \leq 0$.

Supposing $\epsilon \geq 0$, we can derive:

$$\begin{aligned}
&f_K(b_{l\cup j}) - f_K(b_l) \geq \langle \nabla f_K(b_l), b_{l\cup j} - b_l \rangle + \frac{m_{K,\Omega_1}}{2} \|b_{l\cup j} - b_l\|_2^2 \\
&f_K(b_{l\cup j}) - f_K(b_l) \leq \langle \nabla f_K(b_l), b_{l\cup j} - b_l \rangle + \frac{M_{K,\Omega_1}}{2} \|b_{l\cup j} - b_l\|_2^2 \\
&f_K(b_{l\cup l'}) - f_K(b_l) \geq \langle \nabla f_K(b_l), b_{l\cup l'} - b_l \rangle + \frac{m_{K,\Omega_\zeta}}{2} \|b_{l\cup l'} - b_l\|_2^2 \\
&f_K(b_{l\cup l'}) - f_K(b_l) \leq \langle \nabla f_K(b_l), b_{l\cup l'} - b_l \rangle + \frac{M_{K,\Omega_\zeta}}{2} \|b_{l\cup l'} - b_l\|_2^2
\end{aligned} \tag{25}$$

and

$$\begin{aligned}
&h_k(b_{l\cup j}) - h_k(b_l) \leq \langle \nabla h_k(b_l), b_{l\cup j} - b_l \rangle - \frac{m_{k,\Omega_1}}{2} \|b_{l\cup j} - b_l\|_2^2 \\
&h_k(b_{l\cup j}) - h_k(b_l) \geq \langle \nabla h_k(b_l), b_{l\cup j} - b_l \rangle - \frac{M_{k,\Omega_1}}{2} \|b_{l\cup j} - b_l\|_2^2 \\
&h_k(b_{l\cup l'}) - h_k(b_l) \leq \langle \nabla h_k(b_l), b_{l\cup l'} - b_l \rangle - \frac{m_{k,\Omega_\zeta}}{2} \|b_{l\cup l'} - b_l\|_2^2 \\
&h_k(b_{l\cup l'}) - h_k(b_l) \geq \langle \nabla h_k(b_l), b_{l\cup l'} - b_l \rangle - \frac{M_{k,\Omega_\zeta}}{2} \|b_{l\cup l'} - b_l\|_2^2
\end{aligned} \tag{26}$$

From Eq.25, we conduct the similar analysis as in Eq.22 and 23, which gives the submodularity ratio at the optimistic case:

$$\begin{aligned}
&\sum_{j\in l'} f_K(b_{l\cup j}) - f_K(b_l) \geq \frac{m_{K,\Omega_1}|\zeta|}{2} - \|\nabla f_K(b_l)_\zeta\|_2 \\
&f_K(b_{l\cup l'}) - f_K(b_l) \leq \frac{M_{K,\Omega_\zeta}|\zeta|}{2} - \|\nabla f_K(b_l)_\zeta\|_2
\end{aligned} \tag{27}$$

$$\begin{aligned}
&\sum_{j\in l'} h_k(b_{l\cup j}) - h_k(b_l) \geq \sum_{j\in l'} \langle \nabla h_k(b_l), b_{l\cup j} - b_l\rangle - \frac{M_{k,\Omega_1}}{2}\|b_{l\cup j} - b_l\|_2^2 \\
&\geq -\frac{M_{k,\Omega_1}|\zeta|}{2} - \|\nabla h_k(b_l)_\zeta\|_2 \\
&h_k(b_{l\cup l'}) - h_k(b_l) \leq \langle \nabla h_k(b_l), b_{l\cup l'} - b_l\rangle - \frac{m_{k,\Omega_\zeta}}{2}\|b_{l\cup l'} - b_l\|_2^2 \\
&\leq \frac{-m_{k,\Omega_\zeta}|\zeta|}{2} - \|\nabla h_k(b_l)_\zeta\|_2
\end{aligned} \tag{28}$$

Let $\gamma_k \in (0,1)$ and $-\frac{M_{k,\Omega_1}|\zeta|}{2} - \|\nabla h_k(b_l)_\zeta\|_2 \geq \frac{1}{\gamma_k}(\frac{-m_{k,\Omega_\zeta}|\zeta|}{2} - \|\nabla h_k(b_l)_\zeta\|_2)$. The submodularity ratio of h_k, $\gamma_k \leq \frac{\frac{m_{k,\Omega_\zeta}|\zeta|}{2} + \|\nabla h_k(b_l)_\zeta\|_2}{\frac{M_{k,\Omega_1}|\zeta|}{2} + \|\nabla h_k(b_l)_\zeta\|_2}$.

Proofs to Theorem.3 (Eq.9) In the pessimistic scenario, assessing adversarial robustness can be formulated as a problem of monotone weakly submodular function maximization. Assuming that $m_f(l)$ is the solution l obtained by the *FSGS* method and $m_f(l^*)$ is the optimal solution l^* to Eq.4. Based on Theorem.3 in Elenberg et al. (2018), we can obtain:

$$1 - |m_f(l)| \geq (1 - |m_f(l^*)|)(1 - e^{-\gamma_\zeta^{pess}}) \tag{29}$$

Therefore, we can derive the first inequality (the pessimistic scenario) in Eq.9

$$|m_f(l)| + e^{-\gamma_\zeta^{pess}} \leq (1 - e^{-\gamma_\zeta^{pess}})|m_f(l^*)| \tag{30}$$

Assessing adversarial robustness in the optimistic scenario (Eq.5) is in nature a problem of non-monotone weakly submodular function maximization. Assuming that $m_f(l)$ is the greedy search based solution obtained by the *RandGS* (Randomized Greedy Search) and $m_f(l^*)$ is the optimal solution to Eq.5. Applying Theorem 1.9 Santiago & Yoshida (2020) directly, we can obtain:

$$|m_f(l)| \geq \frac{1}{\gamma_\zeta^{optim}} e^{-\gamma_\zeta^{optim}} |m_f(l^*)| \tag{31}$$

The proof to the approximation gap of *OMPGS* in Eq.10 follows the proof to Theorem 2.4 and Theorem 3 Santiago & Yoshida (2020).

C Revisit Orthogonal Matching Pursuit Greedy Search

We first provide the pseudo codes of *OMPGS* in Algorithm 1 in the appendix, which explains how it is adopted to solve Eq.4 and Eq.5. In each iteration, for each subset s of the previous support set l_{k-1}, we compute the gradient of f_y with respect to $\mathbf{b}_s$ (binary indicator matrix $\mathbf{b}$ with the entries in s changed). The top-k' attributes with the largest magnitudes in the gradient vector are selected to form a candidate set $\hat{s}$. We can then find out the optimal attribute $j \in \hat{s}$ to extend for each subset s and record the optimal attack objective value $g(l_{k-1} \cup j)$. In the outer iteration, we choose finally the attribute j^* producing the largest marginal gain to add into l_{k-1}.

The worst case cost of objective function evaluation in each iteration of *OMPGS* is bounded by $O((\sum_{i=1}^{k-1} C_k^i))|k'|$. We adjust k' to achieve a trade-off between enlarging the search range of greedy

Algorithm 1 Orthogonal Matching Pursuit based Greedy Search (OMPGS)

Input: The attack budget K, the set function based attack objective $g(l)$ and the set $H=\{(i,j), i=1...n, j=1...m\}$ of all the modifiable discrete attributes selected support set l_k, with $|l_k| \leq K$;
Output: $g(l_k)$ and the optimal subset of l_k achieving the attack goal $S_0 \leftarrow \emptyset$

for $k=1,2,...,K$ **do**
  $T=\emptyset$
  for $s \subset l_{k-1}$ **do**
    $r \leftarrow \nabla f_y(\mathbf{b}_s)$
    $\hat{s}=\{j_1,j_2,...,j_{k'}\} \leftarrow \underset{j \in \{H/l_{k-1}\}}{\text{argmax}} \ |<e_j, r>|$
    $\tilde{j} \leftarrow \underset{j \in \hat{s}}{\text{argmax}} g(l_{k-1} \cup \{j\})$
    $T=T \cup \{(\tilde{j}, g(l_{k-1} \cup \tilde{j}))\}$
  end for
  $j^* \leftarrow \underset{j \in T}{\text{argmax}} g(l_{k-1} \cup \{j\})$
  $l_k \leftarrow l_{k-1} \cup \{j^*\}$
end for

search and the cost of function evaluation. Usually k' is much less than $|H|-|l_k|$, especially when H is significantly large (e.g. $H>10$k). Thus *OMPGS* runs significantly faster than *FSGS*.

We further illustrate the rationality of using *OMPGS* to solve the weakly submodular maximization based assessment problem. In Eq. 32, we bound the approximation accuracy of *OMPGS* for the **non-monotone weakly submodular maximization** problem solving the optimistic assessment problem raised in Theorem 4. Comparing to Theorem 3, *OMPGS* achieves a lower but similar approximation quality to the underlying optimal solution to Eq. equation 5.

We assume that $\hat{b}$ and $\hat{b}'$ denote two sets of categorical attribute changes. b indicates the category value assignment of an unperturbed data instance $\hat{x}$. $|\text{diff}(b,\hat{b})| \leq \zeta, |\text{diff}(b,\hat{b}')| \leq \zeta, |\text{diff}(\hat{b},\hat{b}')| \leq \zeta, \zeta \geq 1$. The targeted classifier is denoted as $f_{y_k}(x)$ (k=1,...,**K**), in which **K** is the correct class label.

$$
\begin{aligned}
&|f_{y_k}(\hat{x},\hat{b}')-f_{y_k}(\hat{x},b)| \leq max\{\frac{1}{2m_{k,\zeta}}\|\nabla f_{y_k}(x,\hat{b})_\nu\|_2^2, \\
&\|\nabla f_{y_k}(x,\hat{b})_\nu\|_2+M_{k,\Omega_\zeta}|\zeta|/2\} \\
&|f_{y_{\mathbf{K}}}(\hat{x},\hat{b}')-f_{y_{\mathbf{K}}}(\hat{x},\hat{b})| \geq min\{\frac{1}{2M_{\mathbf{K},\zeta}}\|\nabla f_{y_{\mathbf{K}}}(x,\hat{b})_\nu\|_2^2, \\
&\|\nabla f_{y_{\mathbf{K}}}(x,\hat{b})_\nu\|_2+m_{k,\Omega_\zeta}|\zeta|/2\}
\end{aligned} \tag{32}
$$

where $\nu=\text{diff}(\hat{b},\hat{b}')$.

D Experimental study

Besides the multi-class datasets *Yelp*, *IPS* and *Splice*, we also include another two binary classification datasets collected from *Cyber Security* and *HealthCare* in the experiments. We add the feature sensitivity with the multi-class datasets *Yelp*, *IPS* and *Splice* in D.3 as the supplementary information to the main text. We add the the robustness assessment with the binary-class datasets *PEDec* and *EHR* in D.4. It includes the summarised table on the robustness assessment and the information-theoretic characterization of adversarial vulnerability verified in the experiment.

D.1 Dataset information

Yelp-5 (**Yelp**)(Asghar, 2016) We use the Yelp-5 provided in the torchtext package contains 650,000 training and 50,000 testing textual samples. In the classifier training process. We learn a 300-dimensional embedding for each word. Thus, each sentence is encoded as $x \in \mathbb{R}^{n*300}$, where n is the number of words. we treat every word as a categorical feature and put this data into the LSTM

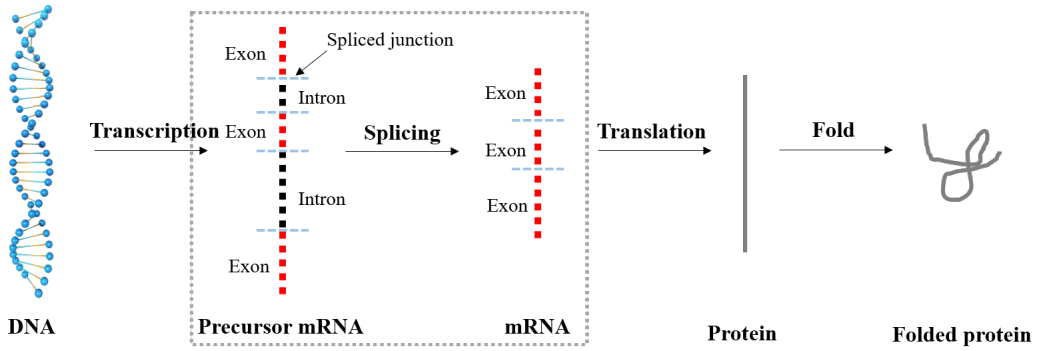

Figure 2: RNA synthesis and translation into protein.

model with five level rating classes of the reviews as the labels. The task is to predict the 5-class rating of each review. After learning the classifier, we start to attack the categorical input with modifying the words with those semantic neighbor words, which can contain the sentence is semantically correct. Here not all the words have neighbors. Here we choose the 200 data from the testing data for attack.

Intrusion Prevention System Dataset (IPS)(Wang et al., 2020). An adversary of network intrusion performs a series of actions to compromise the targeted devices. We collect one day of IPS records via a security service vendor from 242,467 endpoint devices. After removing repeated instances, the collected data set contains 4,101 time series of attack events. Each sequence instance is composed by 20 attack steps. On each attack step, the adversary can choose one of 1,103 different malicious actions registered as highly threatening ones. Thus one data instance of IPS data is given as $\mathbf{x} \in R^{20*1103*70}$ according to the definition given in Section.3. Each of the 1103 attack actions is projected to a 70-dimensional embedding vector. In our study, each sequence instance is used as input to the classification system, which predicts the most likely attack operation conducted at the immediately successive step of the sequence. We randomly select 80% of each dataset for training and others for testing. Based on the prediction output, security analysts can proactively take prevention actions. We focus on predicting the occurrence of the two most threatening actions related to recently uncovered vulnerability. We thus study a 3-class classification task: the 2 highly malicious actions and all the others as the third class. All the privacy-related information, such as names of the devices, IP addresses and hosting machines' domain names have been removed and anonymized.

Splice-junction Gene Sequences (Splice) (Noordewier et al., 1991).Splice junctions are the points on a DNA sequence at which redundant DNA segments are removed during the process of protein creation. As shown in Figure 2Htike & Win (2013), a precursor mRNA contains exons and introns. During splicing, introns get spliced out while exons get reunited. It is then important to analyze the role of splice junction sequences in mRNA. The *Splice* dataset contains 3,190 samples from Genbank release 64.1. Each of the instance is a gene fragment of 60 categorical features, which consists of a window that covers 30 nucleotides respectively before and after each donor and acceptor site. Each categorical feature can take one among 'A', 'G', 'C', 'T' or 'N'(stands for ambiguity). In our work, Each category is cast to a 30-dimension embedding. Then one data instance of the dataset is given as $\mathbf{x} \in R^{60*5*30}$. The dataset defines a 3-class classification task. Each gene instance can be an intro-exon boundary (labelled as *IE*), an exon-intro boundary (labelled as *EI*), or any other cases (labelled as *N*). There are 768 *IE* instances and 767 *EI* instances in total. All the rest are *N* instances. In the published dataset, the data records are anonymized. We randomly select 80% of each dataset for training and others for testing.

Windows PE Malware Detection (PEDec) We build a well-balanced dataset composed by full dynamic analysis reports of 20000 benign and malicious PE executables. This dataset is used for a binary classification scenario of PE malware detection. We collect 10,972 malware samples of 152 families randomly selected from those submitted to VirusTotal between 2018 and 2020. Each of the files https://www.overleaf.com/project/5f81ef055ce9280001ecdee8is classified as malicious by more than 21 antivirus engines. We further use AVClass [68] to ensure that no malware families are over-represented in the collected data. For the benign samples, we choose the fresh installation packages of all the community-maintained packages of Chocolatey [69], where the package submissions go through a strict review process to exclude malicious and pirated software. Therefore we obtain 9028 benign samples of third-party and Windows system files. Each executable

of the dataset is encoded into a binary feature vectors with 5000 signatures selected by human experts. Each signature covers presence or absence of specific windows API calls, URL access, registry table edits, file I/O operations and the status of IP ports. The ensemble of the signatures' binary states form a description of the file's behavior in execution within a sandbox. In our study, we use the 5000 binary codes directly as the features of a file instance. Simple as it is, the combination space of these signatures can still induce prohibitively expensive search cost, which prevent brute-force attacks. We randomly select 80% of each dataset for training and others for testing.

Electronic Health Records (EHR) (Ma et al., 2018) The real-world EHR dataset consists of time-ordered medical visit records of 7314 patients. Each patient has from 4 to 200 medical visits. Each visit record is composed by a subset of 4130 discrete ICD9 diagnosis codes[2]. Each diagnosis code represents occurrence of a disease, a symptom, or an abnormal finding. Using the historical EHR data of patients, we can predict the risk of patients suffering the target diseases. In this experiment, our target is a binary classification task: we forecast whether a patient will suffer heart failure disease in the future. In our experiments, a data instance of EHR data set is organized as a tensor $\mathbf{x} \in R^{200*4130*70}$ with each of the 4130 diagnosis codes projected to a 70-dimensional embedding vector. For the patients with less than 200 visits, we pad the empty observations by setting the corresponding $b_i^j = 0$.

We split randomly each dataset into two non-overlapped subsets: 80% of them are used for training and the left 20% form a testing set. In the testing set, we choose the correctly classified testing samples with the strongest probabilistic confidence provided by the classifier to perform evasion attack. There are thus 125, 500, 324, 3000 and 414 instances of *Yelp*, *IPS*, *Splice*,*PEDec* and *EHR* used in the evasion attack scenario. On *PEDec* and *EHR* data, since the discrete feature takes binary values, we focus on adversarial modification by simply flipping the codes. For *Yelp*, *IPS* and *Splice*, the adversarial attack is conducted by replacing the original category values of a targeted discrete attribute with a new one.

D.2 Experimental Setup

We instantiate the study of robustness characterization and assessment to the popularly deployed Convolution Neural Networks (CNN) and Long Short-Term Memory (LSTM) models. For *PEDec* dataset, we adopt a simple CNN model composed of one convolution layer followed by two linear layers. The rest datasets contain sequential instances, we thus apply standard LSTM as the classifier in the experiments. Without loss of generality, we use *ReLu* activation function in both the CNN and LSTM classifier with the dropout module. Theorem.2 applies in both cases. Both models define Lipschitz-continuous decision functions, which meets the smoothness constraint given in Definition 1. For the instances used for robustness assessment, the CNN and LSTM classifier achieve accurate classification over all the datasets. They achieve accuracy scores of *0.90*, *0.95*, *0.94*, *0.93* and *0.62* respectively for the datasets *IPS*, *Splice*, *PEDec*, *EHR*and *Yelp*.

In Section 4.1, we demonstrate robustness assessment of the two classifiers over all the datasets with varied tolerance levels against adversarial perturbation. We set the robustness threshold Γ to -0.4 and 0 respectively. The former defines a low tolerance to the decision perturbation caused by adversarial attack. A minor fluctuation of the classifier's output triggers an alert of potential evasion attack. In contrast, the latter setting only raises an alarm when the classifier produces an ambiguous decision close to the boundary or report a miss-classification incident. Furthermore, we show the derived robustness assessment scores with respect to an *oblivious adversary* (*OA*,the optimistic assessment) and an *knowledgeable adversary* (*KA*,the pessimistic assessment). The purpose is to show that 1) *our proposed robustness assessment method can adapt varied tolerance settings to adversarial threats in safety-critic applications*; 2) *we aim at confirming the impact of different query capabilities of the adversary over the robustness assessment scores*, as revealed by Theorem 2. For the instances used for robustness assessment, we report the median and mean of the derived robustness assessment scores. Both *FSGS*/*RandGS* and *OMPGS* are used to calculate the robustness assessment scores. They verify the effectiveness of greedy search for robustness assessment with categorical input. We implement the empirical study using the Python library PyTorch and conduct all the experiments on Linux server with 2 GPUs (GeForce 1080Ti) and 16-core CPU (Intel Xeon). For *RandGS* and *OMPGS*, we empirically set the number of candidate attributes in each iteration of greedy search to be 10 globally for all the datasets.

[2] `http://www.icd9data.com/`

Table 4: The feature sensitivity of different datasets

DataSet	Top3			Top5		
	max	average	min	max	average	min
IPS	1.00	0.53	0.00	1.00	0.53	0.00
Splice	1.00	0.38	0.00	1.00	0.33	0.00

Table 5: Top sensitive features vs frequently perturbed features

DataSet	Yelp	IPS	Splice
Sensitive	**not**, but, we, could,**They**	**19**,**0**,**17**,18,16	**29**,**28**,**30**,**31**,27
Perturbed	day,never, **They**,good,**not**	**19**,**0**,1,2,**17**	**29**,**28**,**30**,**31**,32

D.3 FEATURE SENSITIVITY

To further understand the robustness assessment scores, we conduct one-factor-at-a-time sensitivity analysis Campbell et al. (2008) to measure the sensitivity of features. Given a data instance, we change each feature while keeping all the others fixed. The averaged change of the classifier's output over all the testing instances is used as the feature-wise sensitivity measurement. A larger value indicates that the classifier's output is more sensitive to the change over the corresponding feature. In Table 4, we show the *maximum*, *average* and *minimum* decrease of the probabilistic output for *IPS* and *Splice*, induced by changing the **top-3** and **5**-ranked features.

First of all, we can find that the feature sensitivity of *IPS* lies in a narrow interval lower than all the other datasets. This resonates with the larger robust assessment scores in Table 1 and 2. Second, *Splice* also has a narrow interval with high sensitivity values, indicating a potential high vulnerability. Third, within dataset *IPS* and *Splice*, the sensitivity levels vary strongly, between the max and min value. That implies, changing the top sensitive features only is enough to deliver a successful attack. In real-world applications, the existence of such highly sensitive yet sparse sensitive attributes is widely witnessed. For example in *IPS*, the **top-5** ranked features are relevant to the key signatures of network traffic flows, and they are thus usually used by human security analysts to filer out suspicious incidents. The rest of the features in *IPS* are not explicitly relevant with the network I/O behaviours, which are less informative in the classification task. In *Splice*, the **top-5** sensitive features are found as the gene segments around the spliced junctions. They are particularly sensitive to the gene types. The rest of the features show little physical significance in the gene splicing process.

To further understand the association between the feature sensitivity and the derived robustness assesment scores, we conduct one-factor-at-a-time sensitivity analysis on each dataset Campbell et al. (2008) to measure the sensitivity of features in the classification task. Given a data instance, we change each feature while keeping all the others fixed. The averaged change of the classifier's output over all the testing instances is used as the feature-wise sensitivity measurement. A larger value indicates that the classifier's output is more sensitive to the change over the corresponding feature. We find that the top-sensitive features also appear as the most frequently selected features by *FSGS*,*GradAttack* and *OMPGS*. We further count the frequency of appearance for each feature selected in the pessimistic robustness assessment and compare them with the top-sensitive features, as given by Table 5. For *IPS*, the **top-3** sensitive features also appear in the list of the most frequently selected features. The **top-4** sensitive features in *Splice* are also the most frequently selected 4 features. The interesting overlapping between the attributes which are useful for attack and the top-ranked sensitive attributes confirms our intuition about the association between feature sensitivity and adversarial vulnerability of the classifier. On *Splice*, 4 of the most frequently attacked features (29,28,30 and 31 in Table 5) correspond to the junction of exons, which are also highly informative for gene categorization Htike & Win (2013). On *IPS*, 3 of the 5 most frequently attacked categorical features (19,0 and 17) are the also informative indicators of unusual DNS requests and DDos activities. They are ranked as the most sensitive features according to Table 5.

For *IPS*, the **top-3** sensitive features also appear in the list of the most frequently selected features. The **top-4** sensitive features in *Splice* are also the most frequently selected 4 features. The interesting overlapping between the attributes useful for attack and the top-ranked sensitive attributes confirms our intuition about the association between feature sensitivity and adversarial vulnerability of the classifier. Suppressing feature sensitivity helps improve adversarial robustness. However, these sensitive attributes are usually informative for classification in an adversary-free scenario. Excluding

the sensitive features causes accuracy loss. It indicates the balance between classification utility and adversary-resilience feature engineering in ML practices.

D.4 Robustness Assessment in Binary Classification

D.4.1 Robustness assessment with greedy search

Table 6 shows the same findings in binary-class datasets *PEDec* and *EHR* with the multi-class datasets *Yelp*, *IPS* and *Splice*. In details, firstly for both the pessimistic and optimistic robustness assessment problems, a lower tolerance threshold Γ results in lower scores over all the binary-class datasets. Secondly the pessimistic robustness assessment scores are always lower or equal to the optimistic assessment scores.

Table 6: Pessimistic and Optimistic Robustness Assessment Score $|l|$, reported with median and mean. A lower/higher value at a given percentile level indicates stronger/weaker robustness of the classifier. SR is the success rate of attack. OVF indicates SR=0.

GreedySeearch	**P/O**	Γ	**PEDec**		**EHR**	
			SR	**Med (Avg)**	**SR**	**Med (Avg)**
FSGS	P	-0.4	0.92	3 (3.2)	0.98	1 (1.2)
		0	0.87	3 (3.7)	0.94	2 (1.2)
	O	-0.4	0.46	4 (4.1)	0	OVF
		0	0.42	4 (4.7)	0	OVF
GradAttack	P	-0.4	0.56	7 (7.2)	0.63	2 (2.0)
		0	0.51	8 (6.1)	0.62	2 (2.3)
OMPGS	P	-0.4	0.87	6 (6.1)	0.95	2 (2.0)
		0	0.82	6 (6.3)	0.94	2 (2.4)
	O	-0.4	0.62	5 (6.1)	0.38	2 (2.7)
		0	0.54	6 (6.8)	0.42	2 (2.5)

D.4.2 Robustness vs Feature Informativeness

Table 7: Robustness Assessment vs Feature Informativeness, reported at $\Gamma=0$. $I(x_{l_{mi}};y,S) < I(x;y,S)$ results in higher adversarial risk (higher SR, smaller $|l|$) on $x_{l_{mi}}$ than original x.

GreedySeearch	**P/O**	**sample**	**PEDec**		**EHR**	
			SR	**Med(Avg)**	**SR**	**Med(Avg)**
FSGS	P	original x	0.87	3 (3.7)	0.94	2 (1.2)
		$x_{l_{mi}}$	0.96	3 (3.1)	1	1 (1)
	O	original x	0.42	4 (4.7)	0	OVF
		$x_{l_{mi}}$	0.69	3 (3.2)	0.92	2 (2.2)
OMPGS	P	original x	0.82	6 (6.3)	0.94	2 (2.4)
		$x_{l_{mi}}$	0.85	6 (6.1)	1	2 (1.9)
	O	original x	0.54	6 (6.8)	0.62	2 (2.8)
		$x_{l_{mi}}$	0.71	6 (6.4)	0.94	2 (2.1)

we take the $x_{l_{mi}}$ samples from the 62, 43 original samples for the dataset *PEDec*, *EHR* respectively on the half-way of pushing the m_f of the original samples (*OS*) to surpass Γ. In Table 7, we verified the impact of feature informativeness on the robustness assessment scores at $\Gamma=0$ with the binary-class datasets *PEDec* and *EHR*. The table shows that the classifier becomes less robust to adversarial modifications over $x_{l_{mi}}$ with less predicative information.

D.4.3 Robustness Assessment after Robustness Enhancement

From the parameter of the decreased **SR** in Table 8, we study the robustness enhancement by different model enhancement techniques. On *PEDec*, the most robustness assessment scores increase slightly. Notably, the decreased **SR** of *AdvC* and *NuR* are positive and brings the most robustness improvement. On *EHR*, *NuR* can prevent attack over all the testing instances. As seen in Table 8, the robustness-enhancing techniques induce more visible impact on the pessimistic robustness assessment scores using **OMPGS**. This result meets the intuitive interpretations in the main paper.

Table 8: Pessimistic Robustness Assessment ($\Gamma = 0$) after Model Robustness Enhancement by reducing $I(f_y;S)$. $\downarrow^{\mathbf{SR}}$ is the SR decreased amount comparing to the corresponding SR value with $\Gamma = 0$ in Table 6.

Model	**Algo.**	**PEDec**			EHR		
		SR	$\downarrow^{\mathbf{SR}}$	**Med (Avg)**	**SR**	$\downarrow^{\mathbf{SR}}$	**Med (Avg)**
AdvC	FSGS	0.65	0.22	4 (4.2)	0.94	0	2 (2.17)
	OMPGS	0.58	0.24	6 (6.4)	0.92	0.02	2 (2.82)
NuR	FSGS	0.81	0.06	3 (2.9)	0	0.94	OVF
	OMPGS	0.25	0.57	6 (6.1)	0	0.94	OVF
RS	FSGS	0.93	-0.05	3 (3.2)	0.96	-0.02	1 (1.97)
	OMPGS	0.86	-0.04	4 (4.1)	0.91	0.03	2 (2.20)
SRS (Boj. et al., 2020)		1.0	-	1	1.0	-	1