

HW3 TrustWorthy

علیرضا شیری
۸۱۰۱۰۳۱۶۹

سوال ۱

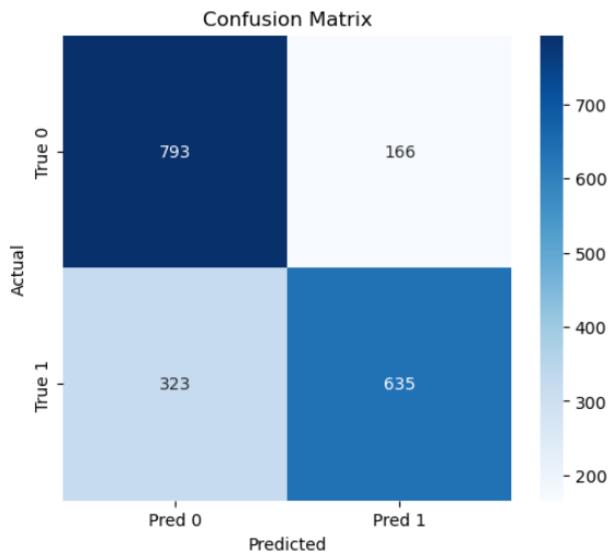
LOGISTIC REGRESSION

برای نرمال سازی داده های minmaxScaler استفاده کردم. با توجه به اینکه مدل ما بر اساس این ویژگی ها دسته بندی را انجام

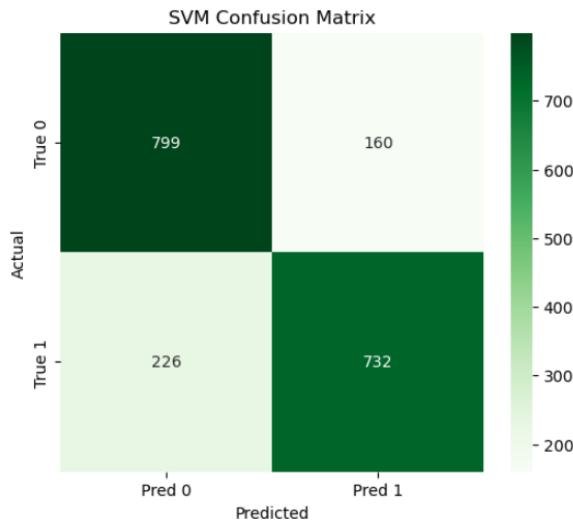
جدول ۱: ضرایب ویژگی ها (مرتب شده بر اساس اهمیت)

Feature	Coefficient
longest_words_raw	10.0256
length_hostname	5.6420
shortest_word_host	5.4088
nb_slash	2.3298
nb_dots	2.1356
shortest_words_raw	-0.3744
avg_words_raw	-0.5394
longest_word_host	-2.7801
avg_word_host	-2.7891
length_url	-4.0871

میدهد ، مقدار قدر مطلق ضرایب ویژگی ها نشان دهنده اهمیت آن ویژگی در تعیین کلاس میباشد ، یعنی هرچه قدر مطلق این ضرایب بیشتر باشد در تعیین مقدار تابع لاجستیک اهمیت بیشتری دارد. به ترتیب بزرگی از مثبت به منفی ضرایب را مرتب کردم. یکسری از ضرایب مقداری بیشتری از دیگری دارند که اهمیت آن هارا نشان میدهد.



شکل ۱: ماتریس آشفتگی مدل لاجستیک

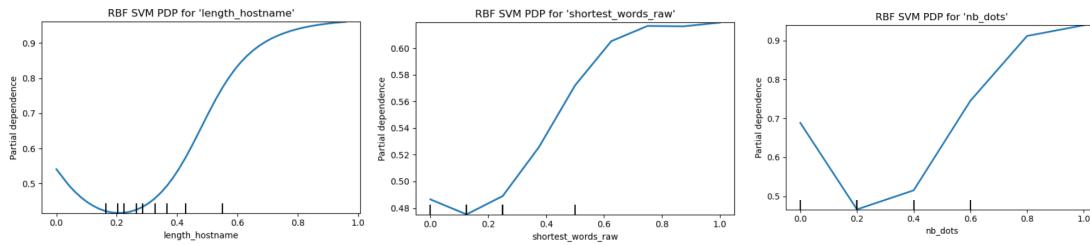
SVM

شکل ۲: ماتریس آشفتگی SVM

برای مدل SVM از کرنل rbf استفاده کردم..

تفسیر مدل ها

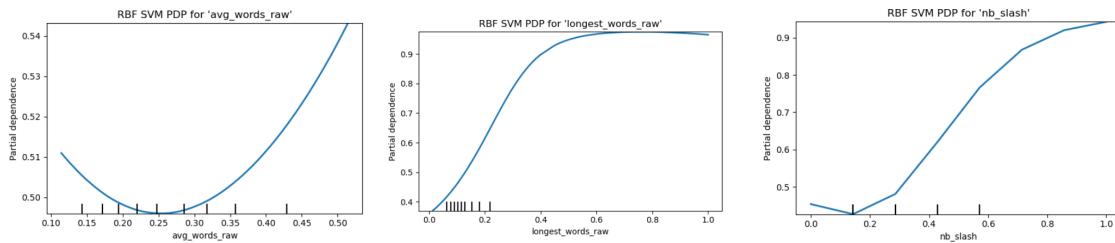
PARTIAL DEPENDENCE PLOT



شكل :۵ length hostname

شكل :۴ shortest words raw

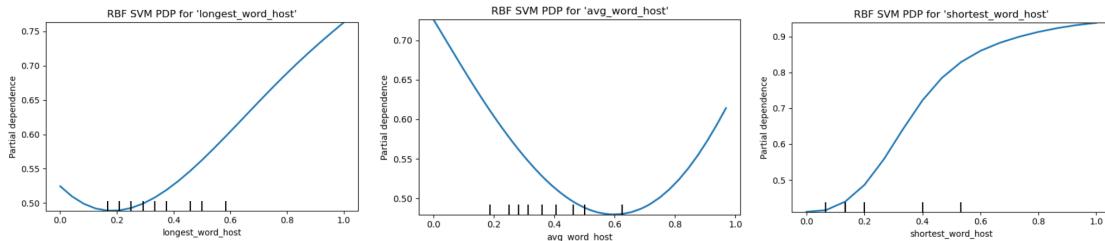
شكل :۳ nbdots



شكل :۸ avg words raw

شكل :۷ longest words raw

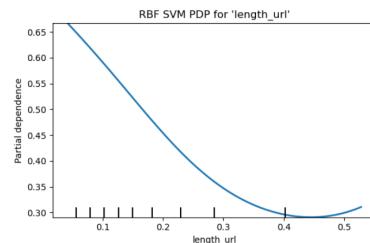
شكل :۶ nb slash



شكل :۱۱ longest word host

شكل :۱۰ avg word host

شكل :۹ shortest word host

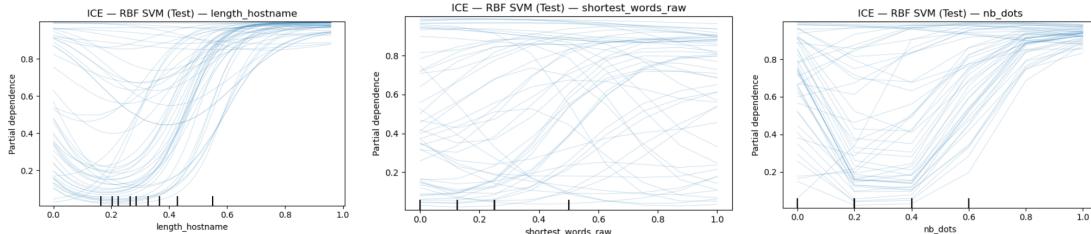


شكل :۱۲ length url

در PDP ما به این صورت عمل میکنیم که برای بررسی اهمیت یک ویژگی با ثابت نگهداشتن بقیه ویژگی ها مقدار آن ویژگی را تغییر میدهیم و اهمیت آن آنرا در تعیین کلاس (مثلا احتمال کلاس ۱) را ثبت میکنیم و نمودار را میکشیم. این نمودار ها دقیقا با

ضرایب بدست آمده همخوانی دارد مثلا اولین ویژگی که بیشترین شب و بیشترین احتمال بدست آمده برای کلاس مثبت را داشت (longestwordsraw) در نمودار pdp آن نیز این اهمیت مشهود است، هرچه مقدار آن بیشتر شده، احتمال تعلق به کلاس مثبت هم به ۱ میل کرد با شب زیاد. بقیه ویژگی ها مانند lengthurl که ضریب منفی برای کلاس مثبت را دارد شب نزولی را دارد.

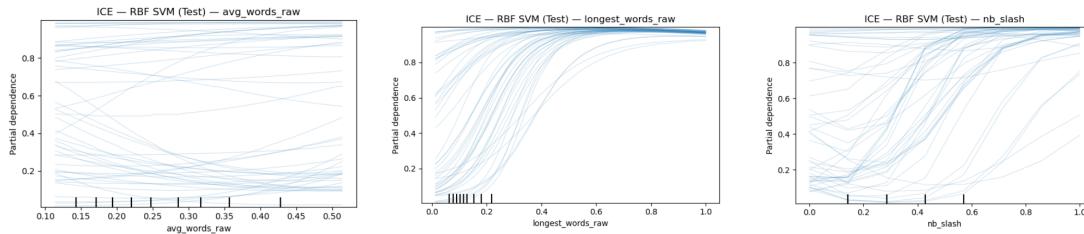
INDIVIDUAL CONDITIONAL EXPECTATION PLOT



شكل ۱۵ : length hostname

شكل ۱۶ : shortest words raw

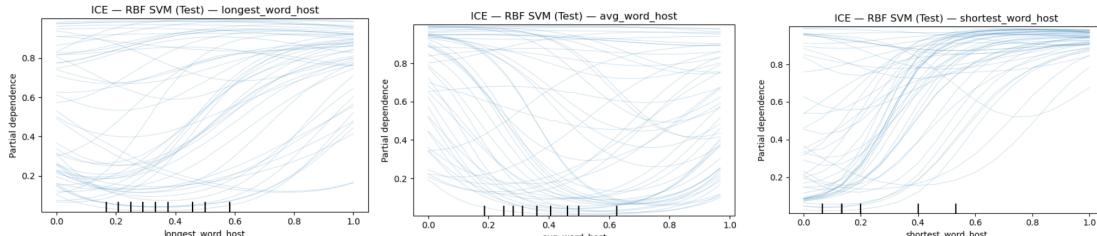
شكل ۱۷ : nb_dots



شكل ۱۸ : avg words raw

شكل ۱۹ : longest words raw

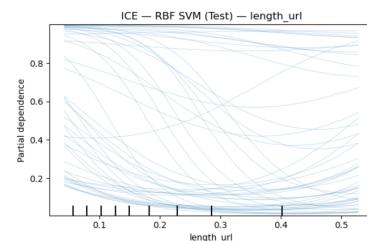
شكل ۲۰ : nb slash



شكل ۲۱ : longest word host

شكل ۲۲ : avg word host

شكل ۲۳ : shortest word host



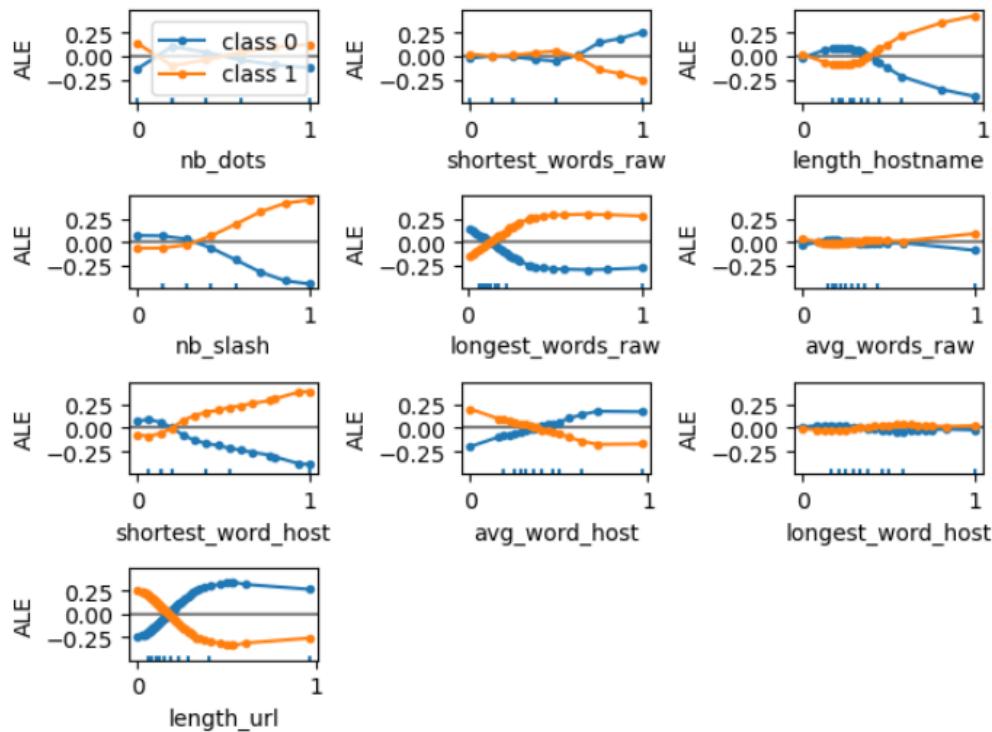
شكل ۲۴ : length url

. ICE اثر تغییرات هر ویژگی را در سطح تک تک نمونه‌ها نشان می‌دهد (individual-level analysis)، در حالی که PDP میانگین این اثرات را در کل مجموعه داده نمایش می‌هد (population-level analysis).

تفاوت کلیدی در این است که ICE می‌تواند تأثیرات مختلف تأثیرات متغروتی بگذارد. این در حالی است که PDP فقط دیدگاه میانگین‌گیری تغییر یک ویژگی ممکن است بر نمونه‌های مختلف تأثیرات متغروتی بگذارد. شده ارائه می‌دهد و ممکن است تغییرات مهم در زیرگروه‌های داده را پنهان کند.

از لحاظ پیاده‌سازی، ICE به ازای هر نمونه و هر مقدار از ویژگی مورد نظر، پیش‌بینی مدل را محاسبه می‌کند، در حالی که PDP میانگین این پیش‌بینی‌ها را گزارش می‌دهد. این باعث می‌شود ICE برای تشخیص feature interactions (تعاملات بین ویژگی‌ها) مناسب‌تر باشد، اما تفسیر آن پیچیده‌تر و محاسبات آن سنگین‌تر است.

ACCUMULATED LOCAL EFFECT



شکل ۲۳: ALE

روش ALE با تقسیم‌بندی بازه‌های یک ویژگی و محاسبه متوسط تغییرات محلی پیش‌بینی مدل در هر بازه، به تحلیل دقیق‌تر اثر واقعی آن ویژگی می‌پردازد. این رویکرد از ثبت تغییرات موضعی استفاده می‌کند تا تأثیر نقاط دور از توزیع داده‌ها را حذف کرده و در مواجهه با همبستگی بین ویژگی‌ها عملکرد قابل اعتمادتری ارائه دهد. در مقابل، نمودار PDP با ثابت نگه داشتن سایر ویژگی‌ها و میانگین‌گیری خروجی مدل، چشم‌اندازی کلی اما گاه گمراه‌کننده از تأثیر هر ویژگی می‌دهد؛ زیرا ممکن است از مقادیری استفاده کند که در داده‌های واقعی وجود ندارند. بنابراین ALE برای درک بهتر تعاملات محلی و جلوگیری از خطاهای ناشی از بروز نمونه مناسب‌تر است. در ALE اثر یک ویژگی مثل «تعداد نقطه‌ها در آدرس» را با مقایسه محلی و واقعی نمونه‌ها می‌سنجدیم؛ مثلاً می‌بینیم وقتی تعداد نقطه‌ها از ۳ به ۴ تغییر می‌کند، احتمال پیش‌بینی «فیشینگ» چقدر افزایش یا کاهش می‌یابد، و این تغییرات را فقط

در بازه‌های واقعی داده شده محاسبه می‌کنیم. اما در PDP برای تعداد نقطه‌ها همان مقدار ۳ یا ۴ را روی همه نمونه‌ها ثابت می‌کنیم و میانگین احتمال «فیشینگ» را می‌گیریم، حتی اگر ترکیب تعداد نقطه‌ها با سایر ویژگی‌ها در داده‌های واقعی وجود نداشته باشد. بنابراین ALE با محاسبه تغییرات موضوعی به ما نشان می‌دهد که بهطور واقعی و در داده‌های نزدیک به هم چه اتفاقی در احتمال فیشینگ می‌افتد، در حالی که PDP نتیجه‌ای کلی و گاهی گمراه کننده ارائه می‌دهد. در ادامه برای هر یک از ویژگی‌های مدل تشخیص فیشینگ، ابتدا رفتار PDP و سپس رفتار ALE توضیح داده شده است:

مقایسه ALE و PDP برای تمام ویژگی‌ها

(تعداد نقطه‌ها در آدرس) **PDP**: نمودار میانگین احتمال فیشینگ با افزایش تعداد نقطه‌ها به‌طور یکنواخت و شدید صعودی است. ALE: تا حدود ۲۰ نقطه تأثیر قابل توجهی ندارد و پس از آن با شیبی قوی افزایش می‌یابد؛ بنابراین مدل ابتدا نسبت به تغییرات کوچک بی‌حس است.

(طول کوتاه‌ترین کلمه در نام هاست) **PDP**: منحنی تقریباً صعودی پیوسته با افزایش طول کوتاه‌ترین کلمه‌ها احتمال فیشینگ بالا می‌رود. ALE: تا بازه‌های میانی اثر نزدیک به صفر است و فقط در دو انتهای دامنه (خیلی کوچک یا خیلی بزرگ) تغییر معنی‌داری دیده می‌شود.

(میانگین طول کلمات در نام هاست) **PDP**: یک منحنی U شکل با کمترین احتمال حوالی ۵.۰. ALE: تقریباً خطی و بی‌اثر؛ یعنی مدل در عمل واکنش موضوعی به میانگین طول کلمات نشان نمی‌دهد.

(طول بلندترین کلمه در نام هاست) **PDP**: روند صعودی آرام تا میانه دامنه و سپس سریع‌تر شدن. ALE: ابتدا کاهش اندک و پس از آستانه‌ای (حدود ۲۰) شیب مثبت قوی؛ اثر آستانه‌ای موضوعی که در PDP پخش شده است.

(طول کل نام هاست) **PDP**: تا حدود ۲۰ کاهش اندکی دارد و بعد به تدریج صعودی می‌شود. ALE: دارای شیب مثبت یکنواخت و بدون فلات یا کاهش موضوعی؛ یعنی هرچه نام هاست بلندتر، احتمال فیشینگ بالاتر.

(تعداد اسلش‌ها در آدرس) **PDP**: تقریباً خطی صعودی. ALE: تا حدود ۱۵.۰ اسلش‌ها تأثیر منفی کوچک و پس از آن افزایش تدریجی تا شیب قوی در مقادیر بالا.

PDP shortest_words_raw: روند صعودی یکنواخت با احتمال فیشینگ بیشتر در مقادیر بالاتر. ALE: تا حدود ۲۰ بی‌اثر، سپس یک جهش صعودی قابل ملاحظه؛ اثر موضوعی که در PDP منتشر شده.

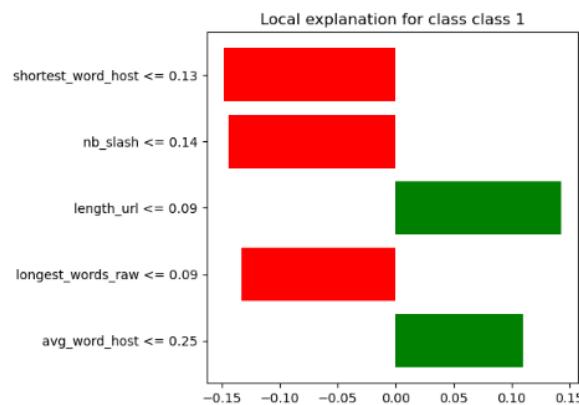
PDP longest_words_raw: شیب قوی صعودی تا میانه دامنه و سپس تقریباً صاف شدن. ALE: در بازه‌های اولیه شیب مثبت زیادی و پس از حدود ۳۰ بی‌اثر؛ نشان‌دهنده حساسیت موضوعی اولیه.

PDP avg_words_raw: U شکل خفیف کاهش تا میانه و بعد افزایش. ALE: تقریباً بدون واکنش موضوعی؛ مدل محلی نسبت به میانگین طول کلمات خام تغییر خاصی نشان نمی‌دهد.

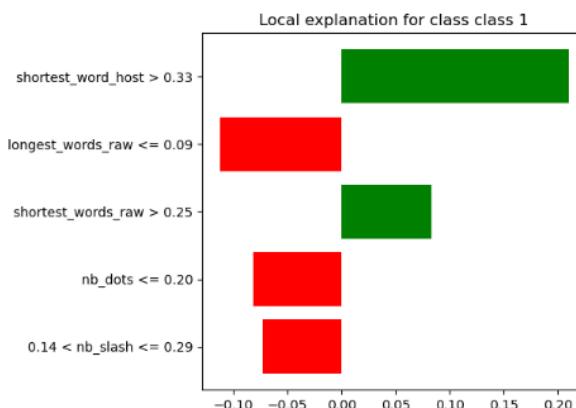
PDP length_url: با افزایش طول URL تا حدود ۴۰ احتمال فیشینگ کاهش می‌یابد و بعد تقریباً مستطح می‌شود. ALE: به صورت یکنواخت و مداوم با افزایش طول URL احتمال فیشینگ بالا می‌رود، بدون کاهش اولیه یا فلات انتهایی.

LIME

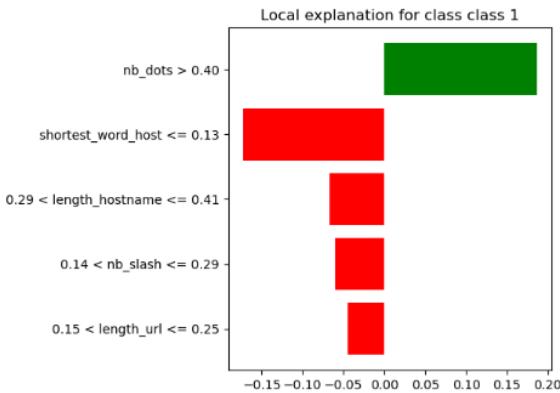
یک روش توضیح‌پذیری محلی است که با ساختن یک مدل ساده (مثل رگرسیون خطی یا درخت تصمیم کم‌عمق) در اطراف نقطه مورد نظر، تاثیر هر ویژگی را بر پیش‌بینی مدل پیچیده اصلی برآورد می‌کند. این روش برای هر نمونه به طور مستقل یکتابع توضیح‌پذیر تخمین می‌زند تا توضیح قابل فهمی از نقش ویژگی‌ها ارائه دهد.



شکل ۲۴: first data point



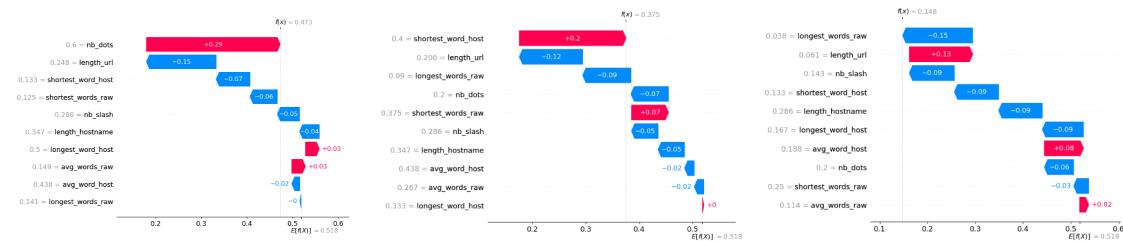
شکل ۲۵: second data point



شکل ۲۶ : third data point

برای تحلیل بهتر و واضح تر فقط ۵ ویژگی که بیشترین تاثیر را داشتند را بررسی کردیم. برای نمونه اول، $\text{avg_word_host} \leq 0.25$ و $\text{length_url} \leq 0.09$ احتمال فیشینگ را کاهش داده و در مقابل $\text{longest_words_raw} \leq 0.09$ و $\text{nb_slash} \leq 0.14$ آن را افزایش داده‌اند. در نمونه دوم، $\text{shortest_word_host} > 0.33$ و $\text{shortest_words_raw} > 0.25$ عامل اصلی افزایش پیش‌بینی فیشینگ بوده و $0.14 < \text{nb_slash} \leq 0.29$ و $\text{nb_dots} \leq 0.20$ اثر کاهشی داشته‌اند. در نمونه سوم، $0.14 < \text{nb_slash} \leq 0.29$ ، $0.29 < \text{length_hostname} \leq 0.41$ ، $\text{shortest_word_host} \leq 0.13$ ، $\text{nb_dots} > 0.40$ بهشت احتمال فیشینگ را بالا برد و $0.15 < \text{length_url} \leq 0.25$ مدل را به سمت کلاس غیر فیشینگ سوق داده‌اند.

SHAP



شکل ۲۷ : third data point

شکل ۲۸ : second data point

شکل ۲۹ : first data point

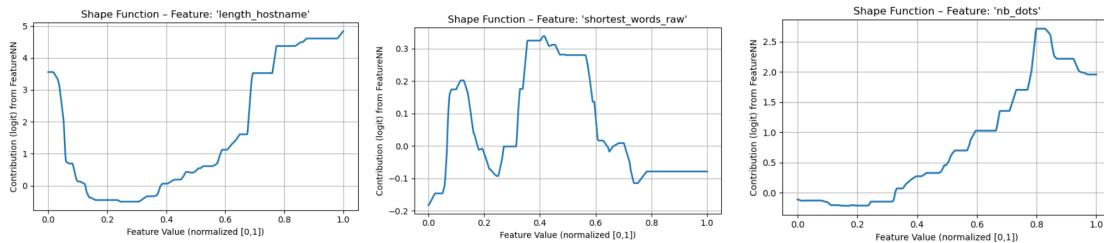
در SHAP سهم هر ویژگی در پیش‌بینی نهایی برای هر نمونه به‌وضوح مشخص می‌شود؛ برای مثال در نمونه اول lengthurl بیشترین تأثیر مثبت و longestwordsraw بیشترین تأثیر منفی را داشته‌اند، در نمونه دوم shortestwordhost قوی‌ترین حرک مثبت و lengthurl قوی‌ترین حرک منفی بوده است، و در نمونه سوم nb_dots بازترین افزایش احتمال فیشینگ بوده است. برخلاف LIME که تغییرات موضعی هر ویژگی را در بازه‌های مشخص شده نشان می‌دهد، SHAP با محاسبه سهم دقیق شاپلی از نظریه بازی، تفکیک واضح‌تری از تأثیر مثبت یا منفی هر ویژگی ارائه کرده و امکان مقایسه مستقیم اهمیت ویژگی‌ها را در سطح نمونه فراهم می‌سازد.

روش LIME با ساخت یک مدل ساده محلی، مفاهیم پیچیده مدل اصلی را در اطراف هر نمونه با تقریب خطی یا درخت کم‌عمق توضیح می‌دهد که تفسیر آن ساده‌اما غیرمنسجم و وابسته به نمونه‌های نزدیک است. در مقابل، SHAP با بنیان در نظریه بازی مقادیر شاپلی، سهم هر ویژگی را به صورت افزایشی و یکنواخت محاسبه می‌کند تا جمع سهم‌ها دقیقاً پیش‌بینی مدل را بازسازی کند؛

این رویکرد عموماً منسجم‌تر، قابل توزیع و قابل مقایسه بین نمونه‌هاست اما هزینه محاسباتی بالاتری دارد.

NAM

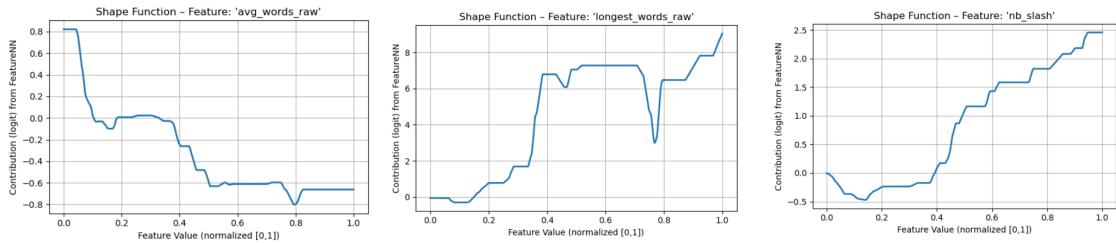
مدل‌های افزودنی عصبی (NAM) با یادگیری جداگانه یکتابع غیرخطی برای هر ویژگی ورودی و جمع‌بندی خروجی این توابع، تفسیرپذیری بالاتری ارائه می‌کنند؛ در مقابل، شبکه‌های پرسپترون چندلایه مرسوم (MLP) همه ویژگی‌ها را همزمان و از طریق لایه‌های متراکم پردازش کرده و تعاملات پیچیده را به صورت درون‌مدل می‌آموزند اما به «جعبه‌سیاه» تبدیل می‌شوند. NAM به کمک ساختار افزودنی ساده و امکان رسم مجزای تابع هر ویژگی، سهم تک‌تک متغیرها را واضح می‌سازد و ریسک بیش‌برازش را کاهش می‌دهد، ولی در مسائلی که تعاملات میان ویژگی‌ها نقش تعیین‌کننده دارند، نیاز است این تعاملات صریحاً به مدل اضافه شوند تا قدرت مدل‌سازی MLP حفظ گردد.



شكل ۳۲ : length hostname

شكل ۳۱ : shortest words raw

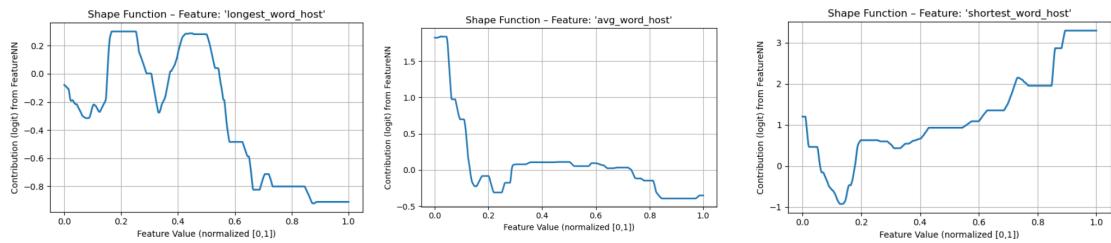
شكل ۳۰ : nb_dots



شكل ۳۵ : avg words raw

شكل ۳۴ : longest words raw

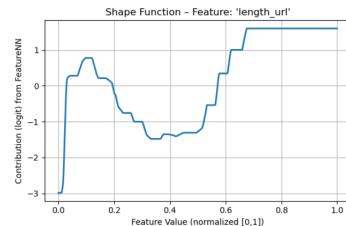
شكل ۳۳ : nb slash



شكل ۳۸ : longest word host

شكل ۳۷ : avg word host

شكل ۳۶ : shortest word host



شكل ۳۹ : length url

مقایسه NAM و PDP

شباهت‌ها

هر دو، PDP و تابع‌های شکل NAM، بر تأثیر یک ویژگی خاص بر پیش‌بینی تمرکز دارند. آن‌ها از نمودارهای خطی برای نمایش رابطه بین مقادیر ویژگی و اثرات آن استفاده می‌کنند.

تفاوت‌ها

□ منشأ داده: PDP از یک مدل SVM با هسته RBF استخراج شده و میانگین تأثیر ویژگی را با نادیده گرفتن تعاملات با سایر ویژگی‌ها نشان می‌دهد (تخمینی است).تابع‌های شکل NAM مستقیماً از مدل آموخته می‌شوند و هر (x_i, f_i) توسط یک شبکه عصبی کوچک آموزش داده شده است (دقیق و ذاتی).

□ پیچیدگی و انعطاف‌پذیری: نمودارهای PDP (مثلًاً افزایش نرم در 'lengthhostname') معمولاً صاف‌تر هستند و پرس‌های تیز را به خوبی نشان نمی‌دهند.تابع‌های شکل NAM (مثلًاً 'nbdots' با پرس‌های واضح) از واحدهای ExU برای مدل‌سازی الگوهای پیچیده و پرس‌دار استفاده می‌کنند.

□ دامنه و مقیاس: مقیاس‌های PDP (مثلًاً ۵.۰ تا ۹.۰ برای 'nbdots') معمولاً در بازه محدودی از وابستگی جزئی هستند. مقیاس‌های NAM (مثلًاً ۵.۰ تا ۵.۲ برای 'nbdots') بازه گسترده‌تری از مشارکت لگاریتم odds را پوشش می‌دهند.

مثال‌های خاص

: PDP یک افزایش پیوسته نشان می‌دهد، در حالی که تابع شکل NAM پرس‌های مشخصی (مثلًاً در ۶.۰ و ۸.۰) دارد که توانایی NAM در گرفتن الگوهای واقعی را نشان می‌دهد. PDP یک افزایش نرم دارد، اما تابع شکل NAM ابتدا کاهش می‌یابد و سپس پرس‌های صعودی دارد، که نشان‌دهنده حساسیت بیشتر NAM به تغییرات محلی است.

مزایا و معایب

مزایا

□ دقت و انعطاف: تابع‌های شکل NAM با استفاده از واحدهای ExU، پرس‌ها و الگوهای غیرخطی را بهتر از نمودارهای صاف PDP (مثلًاً 'NAM' نشان می‌دهند).

□ تفسیرپذیری ذاتی: برخلاف PDP که تخمینی است، تابع‌های شکل NAM بینش مستقیمی از مدل ارائه می‌دهند و فهم را بهبود می‌بخشند.

□ شفافیت: NAM مشارکت لگاریتم odds را نمایش می‌دهد که به تحلیلگران اجازه می‌دهد تأثیر دقیق هر مقدار ویژگی را ارزیابی کنند، برخلاف میانگین کلی PDP.

معایب

□ محدودیت تعاملات NAM: ساختار افزونهای NAM تعاملات بین ویژگی‌ها را نادیده می‌گیرد، در حالی که PDP ممکن است آن‌ها را به صورت ضمنی (هرچند ناقص) در نظر بگیرد.

□ پیچیدگی بیشتر NAM: تفسیر تابع‌های شکل NAM با پرس‌ها و نوسانات (مثلًاً 'longestwordsraw') ممکن است برای کاربران غیرحرفه‌ای سخت‌تر از نمودارهای ساده PDP باشد.

□ وابستگی به داده NAM: تابع‌های شکل به شدت به توزیع داده وابسته هستند و ممکن است در داده‌های پر سرو صدا یا ناقص بیش‌برازش کنند.

تأثیر بر تفسیرپذیری

نمودارهای PDP برای مدل‌های سیاه‌جعبه مثل SVM مفیدند و دید کلی از تأثیر ویژگی ارائه می‌دهند، اما نادیده گرفتن تعاملات می‌تواند به سوءتفاهم منجر شود (مثلاً ساده‌سازی 'avgwordsraw'). تابع‌های شکل NAM به دلیل تفکیک‌پذیری ویژگی‌ها و توانایی مدل‌سازی پرس‌ها (مثلاً 'lengthurl' NAM)، تفسیرپذیری بالاتری دارند. آن‌ها امکان تحلیل دقیق مقادیر خاص (مثلاً ۸۰ در 'nbdots') را فراهم می‌کنند و در حوزه‌های حساس مثل پزشکی اعتماد بیشتری ایجاد می‌کنند (بخش ۱.۴ مقاله).

ANCHOR

روش Anchors یک تکنیک توضیح‌پذیری محلی است که به جای تقریب خطی، با جستجوی قیدهای باینری ساده («لنگر») برای یک پیش‌بینی، تضمین می‌کند که تا زمانی که این قیدها برقرار باشند، خروجی مدل بدون تغییر باقی می‌ماند. این روش برخلاف LIME که با نمونه‌سازی و وزن‌دهی فاصله‌ها به دنبال یک مدل ساده خطی می‌گردد و ممکن است توضیحات ناپایدار یا ابسته به نمونه‌های خارج از توزیع ارائه دهد، توضیحاتی شفاف و با دقت محلی بالا تولید می‌کند. از مزایای Anchors می‌توان به تکرارپذیری توضیحات، اجتناب از نمونه‌های نامعتبر و اعتمادسازی کاربر اشاره کرد؛ اما در فضای ویژگی‌های پیوسته یا بعد بالا یافتن لنگرهای کوتاه و دقیق دشوار شده و هزینه جستجوی قیدها ممکن است افزایش یابد.

Table 2: Anchor rule (Precision ≥ 0.8 , Coverage ≥ 0.4)

Found Anchor Rule

```
If shortest_word_host ≤ 0.13 AND shortest_words_raw ≤ 0.25 AND
nb_dots ≤ 0.40 AND nb_slash ≤ 0.43
→ Precision = 0.849, Coverage = 0.457
```

این شرط بالا باروش Anchors این قاعده را یافته است: اگر $\text{shortest_words_raw} \leq 0.25$ و $\text{shortest_word_host} \leq 0.13$ و $\text{nb_slash} \leq 0.43$ و $\text{nb_dots} \leq 0.40$ با هم برقرار باشند، آنگاه مدل با دقت (Precision) حدود ۰.۸۵ و این قانون جمعاً ۰.۴۵ از داده‌ها (Coverage) را پوشش می‌دهد؛ به عبارت دیگر، هرگاه این چهار شرط همزمان برقرار شود، می‌توان با احتمال زیاد آن URL را فیشینگ در نظر گرفت و این قیدها روی تقریباً نیمی از نمونه‌ها اعمال می‌شوند. قاعده Anchors تعیین می‌کند که وقتی nb_slash و nb_dots و $\text{shortest_words_raw}$ و $\text{shortest_word_host}$ کم هستند، مدل با دقت بالا (۰.۸۵) کلاس «فیشینگ» را پیش‌بینی می‌کند. این با آنچه پیش‌تر در LIME دیدیم که مقادیر پایین nb_slash و $\text{shortest_word_host}$ و $\text{shortest_words_raw}$ و $\text{shortest_word_host}$ را وکنش منفی محلی (کاهش احتمال فیشینگ) نشان می‌دادند در نگاه اول متناقض به نظر می‌رسد؛ اما ALE و SHAP نشان دادند که تأثیرات مثبت قوی ویژگی‌های مانند avg_word_host و length_url در همین ناحیه اغلب بر محرك‌های منفی کوچک غلبه می‌کنند. در نتیجه این قاعده لنگری، که تنها یک زیرمجموعه از فضای ویژگی را می‌پوشاند، با تحلیل سهم‌های شاپلی و تغییرات محلی ALE هماهنگ است و ناحیه‌ای را مشخص می‌کند که پیش‌بینی‌های مثبت مدل بیشترین ثبات و دقت را دارند.

سوال ۲

در انتخاب لایه مناسب برای استخراج ویژگی‌ها جهت Grad-CAM، لایه قبل از fc (Fully Connected) به دلیل ویژگی‌های خاص خود برای تحلیل و تفسیر مدل‌های پیچیده انتخاب می‌شود. این لایه عموماً ویژگی‌های عمیق و انتزاعی‌تری را در مقایسه با لایه‌های ابتدایی که ویژگی‌های ساده‌تری مانند لبه‌ها و بافت‌ها را استخراج می‌کنند، فراهم می‌آورد. لایه‌های fc عموماً برای تجزیه و تحلیل

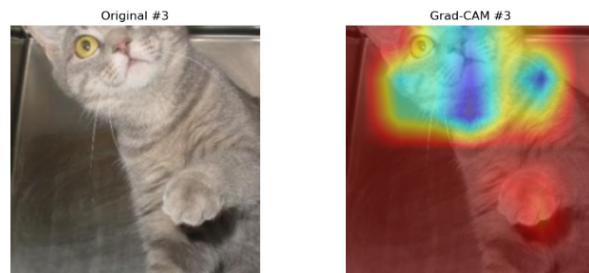
نهایی استفاده می‌شوند و درک وابستگی دقیق ویژگی‌ها در این لایه‌ها کمک می‌کند تا Grad-CAM تأثیرات واقعی ویژگی‌ها در تصمیم‌گیری مدل را نمایش دهد. از این‌رو، انتخاب لایه قبل از fc برای مشاهده تعاملات پیچیده‌تر بین ویژگی‌ها و دسته‌ها ضروری است.

GRAD-CAM

در انتخاب لایه مناسب برای استخراج ویژگی‌ها جهت Grad-CAM، لایه قبل از fc (Fully Connected) به دلیل ویژگی‌های خاص خود برای تحلیل و تفسیر مدل‌های پیچیده انتخاب می‌شود. این لایه عموماً ویژگی‌های عمیق و انتزاعی‌تری را در مقایسه با لایه‌های ابتدایی که ویژگی‌های ساده‌تری مانند لبه‌ها و بافت‌ها را استخراج می‌کنند، فراهم می‌آورد. لایه‌های fc عموماً برای تجزیه و تحلیل نهایی استفاده می‌شوند و درک وابستگی دقیق ویژگی‌ها در این لایه‌ها کمک می‌کند تا Grad-CAM تأثیرات واقعی ویژگی‌ها در تصمیم‌گیری مدل را نمایش دهد. از این‌رو، انتخاب لایه قبل از fc برای مشاهده تعاملات پیچیده‌تر بین ویژگی‌ها و دسته‌ها ضروری است.



شکل ۴۰: تصویر fine tune بدون grad cam



شکل ۴۱: تصویر fine tune بدون grad cam

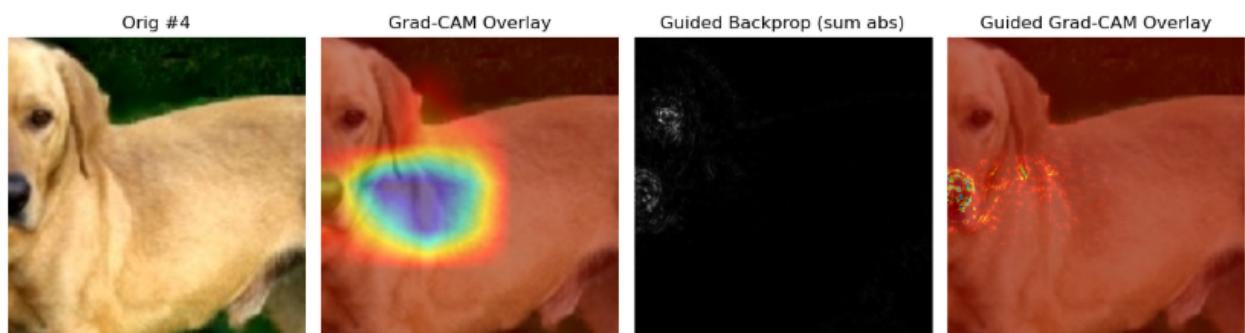


شکل ۴۲: تصویر fine tune بدون grad cam

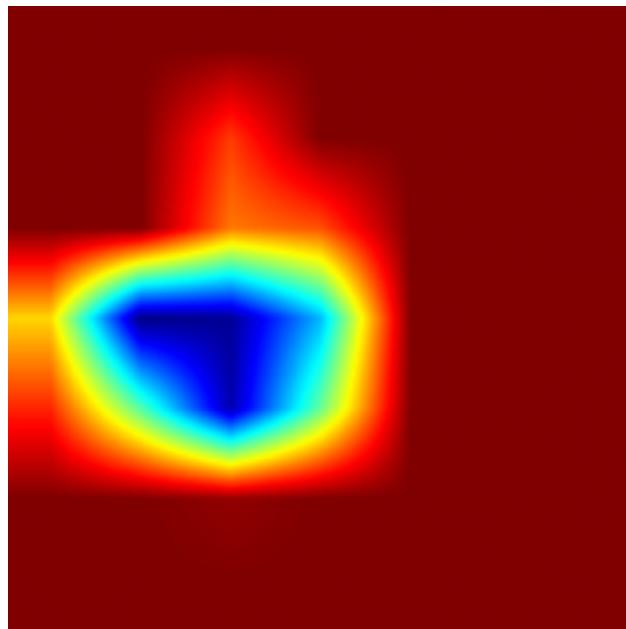


شکل ۴۳: تصویر با grad cam

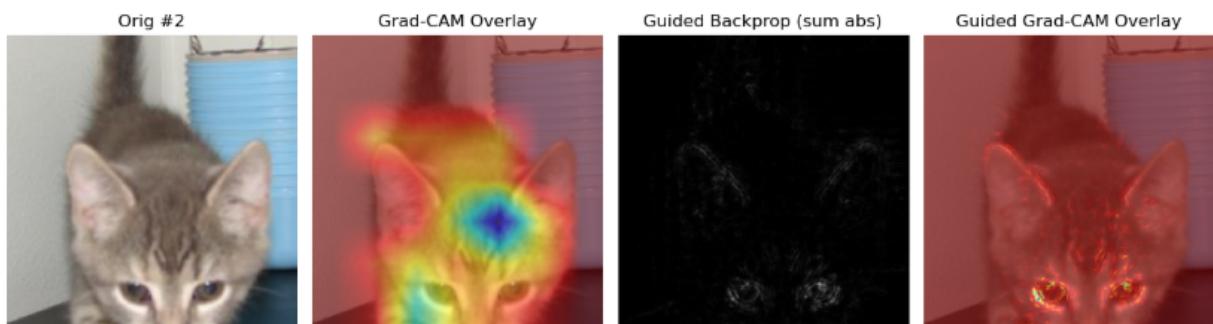
GUIDED BACKPROPAGATION



شکل ۴۴: GUIDED BACKPROPAGATION



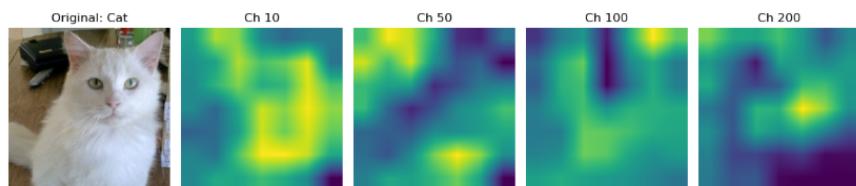
شکل ۴۵: grad cam heat map برای عکس بالا



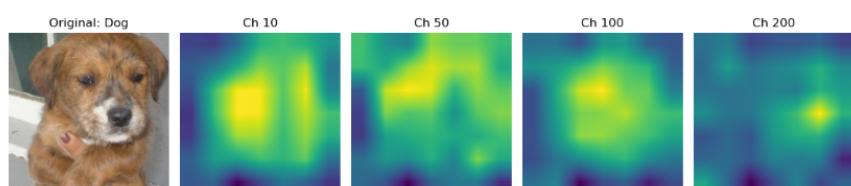
شکل ۴۶: GUIDED BACKPROPAGATION

در مقایسه تصویر اصلی با Grad-CAM اولی، مشاهده می کنیم که نواحی پررنگ عمدتاً بر روی بدن سگ (شانه و پشت) متمرکز است؛ این بخش ها با درک انسانی ما از ویژگی های مهم یک سگ (مثل شکل بدن و بافت خز) همپوشانی دارند. اگرچه در برخی نقاط جزئی یا پس زمینه نیز گرادیان هایی دیده می شود، اما تأکید اصلی روی نواحی معنادار و قابل فهم است. از طرفی، نقشه های Guided Backpropagation الگوهای بافتی و لبه های ظریف را برجسته می کنند که کمتر برای انسان شهودی اند. بنابراین می توان گفت مدل هم به ویژگی های قابل فهم انسانی توجه دارد و هم از الگوهای آماری پیچیده تر بهره می برد: Grad-CAM نشان می دهد تمرکز کلی مدل بر نواحی معنادار است، اما تحلیل دقیق تر گرادیان ها حکایت از اتکا به پترن های آماری در سطوح پایین تر دارد.

FEATURE VISUALIZATION

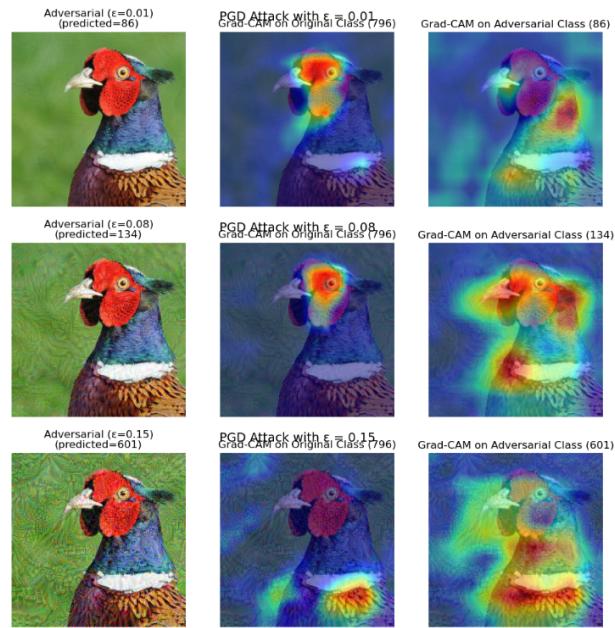


شکل ۴۷: layer3

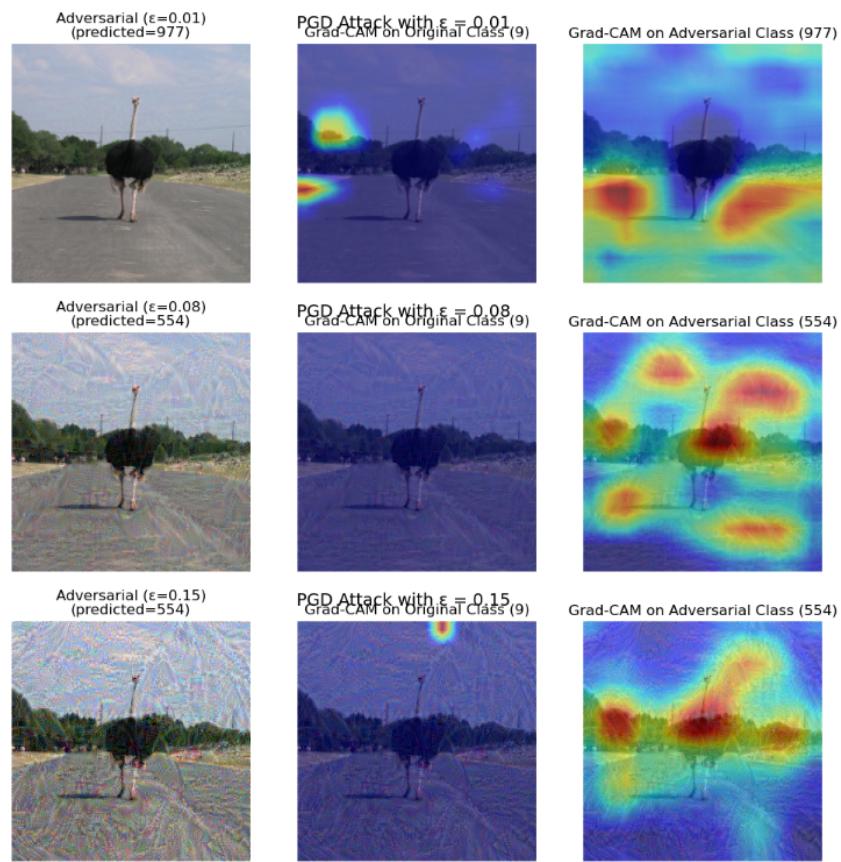


شکل ۴۸: layer3

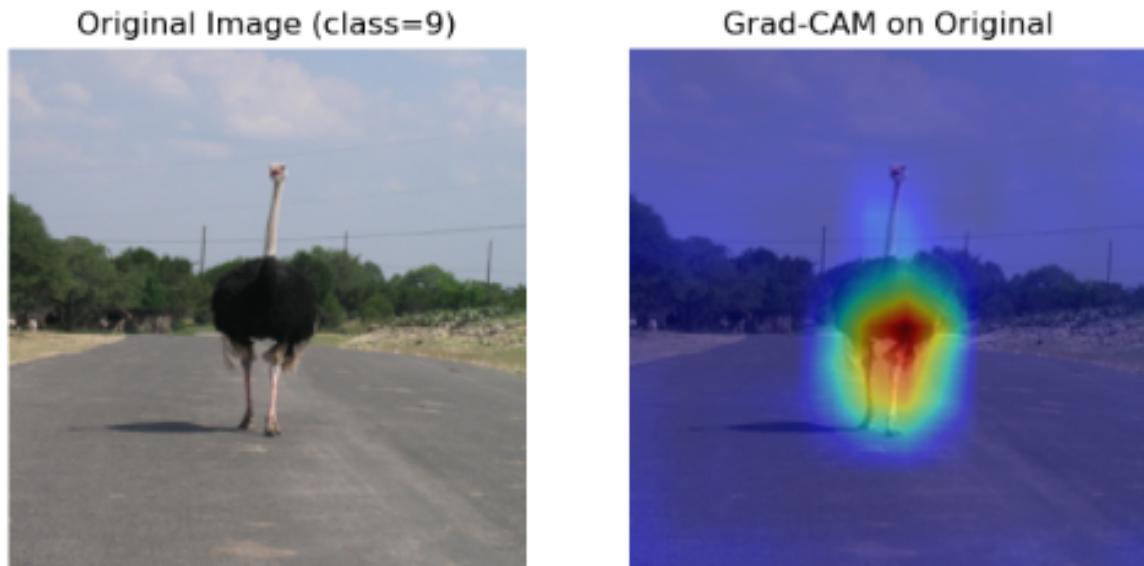
با مشاهده خروجی feature map ها از طریق hook لایه میانی، کanal های ۵۰ و ۱۰۰ بیشترین فعالیت را در نواحی کلی بدن سگ نشان می دهند و کanal ۱۰۰ عمدتاً به لبه ها و مرزهای رنگی حساس است؛ کanal ۲۰۰ واکنش ضعیف تری دارد و فقط نقاط مشخصی مثل چشم و بینی را برجسته می کند. این موضوع نشان می دهد که کanal های پایین تر (مثلاً ۱۰) به ویژگی های ساده مانند لبه و بافت، کanal های میانی (۵۰ و ۱۰۰) به فرم ها و ساختارهای کلی مانند شکل بدن و پوز سگ، و کanal های بالاتر (۲۰۰) به جزئیات منطقه ای کوچک تر مانند چشم یا بینی واکنش می دهند.

GRAD-CAM AND PGD

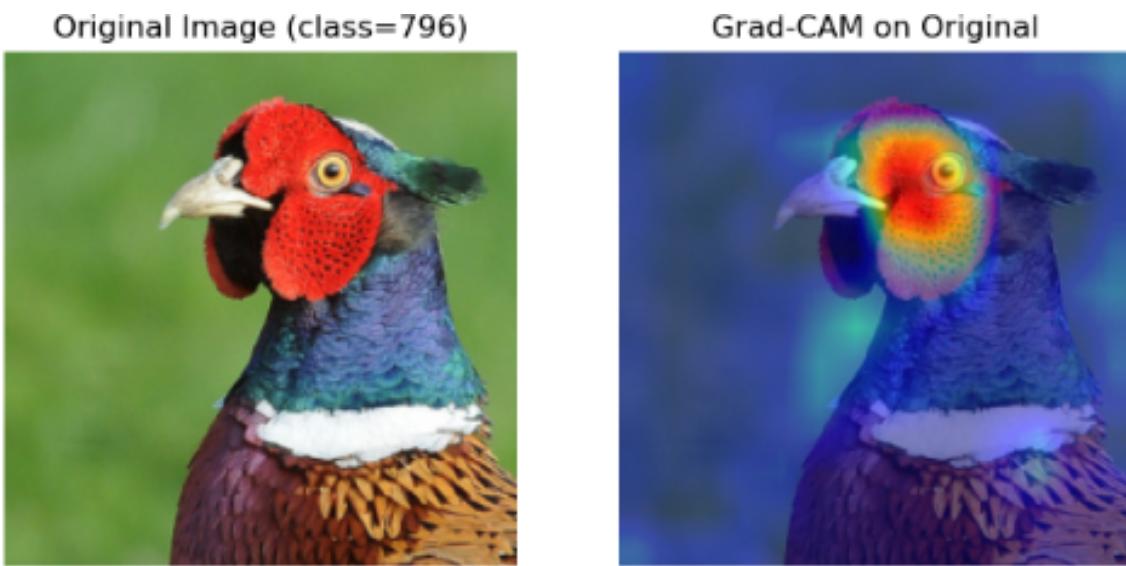
شکل ۴۹: تاثیرات PGD بر ناحیه تمرکز مدل



شكل ۵۰: تاثیرات PGD بر ناحیه تمرکز مدل



شكل ۵۱: grad cam قبل از حمله

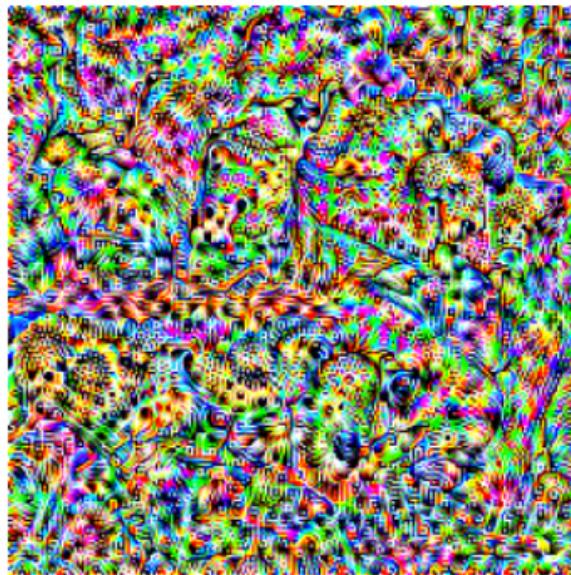


شکل ۵۲: grad cam قبل از حمله

در مواجهه با تصاویر خصمانه تولیدشده توسط حمله Grad-CAM، نقشه‌های نشان می‌دهند که با $\epsilon = 0.01$ تمرکز مدل همچنان روی مناطق معنادار پرنده (چشم و منقار) حفظ می‌شود اما به تدریج با افزایش مقدار ϵ توجه به لبه‌ها و بافت‌های نویزی در پس‌زمینه منتقل می‌شود؛ این جایه‌جایی در توجه موجب می‌گردد ویژگی‌های سطح بالا (مثل شکل ظاهری پرنده) جای خود را به الگوهای مصنوعی بدنه‌ند و مدل به کلاس‌های نادرست گرایش یابد. بنابراین شدت اختلال (مقدار ϵ) عامل اصلی تغییر ساختاری در توزیع توجه شبکه است و این تغییر به طور مستقیم فرآیند تصمیم‌گیری را از تکیه بر الگوهای معنایی به اتکا بر نویزهای دستکاری شده سوق می‌دهد. همچنین نتیجه grad cam برای همان کلا اصلی قبل حمله نشان میدهد که مدل به دنبال شباهت‌هایی با تصویر کلاس اصلی می‌گرد و چیز خاصی پیدا نمی‌کند. در مثال شترمرغ این مورد کاملاً مشهود است.

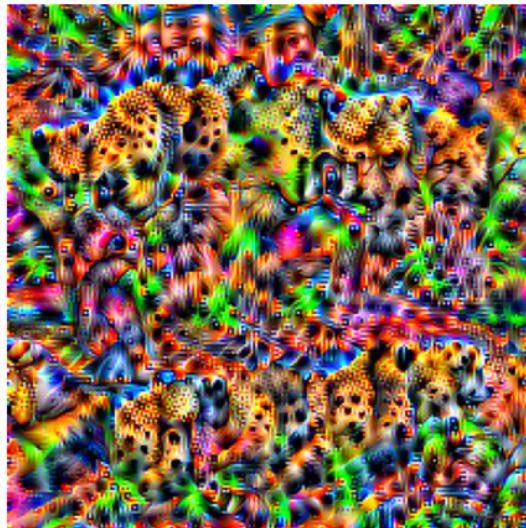
CLASS FEATURE VISUALIZATION

Visualization of Class 293



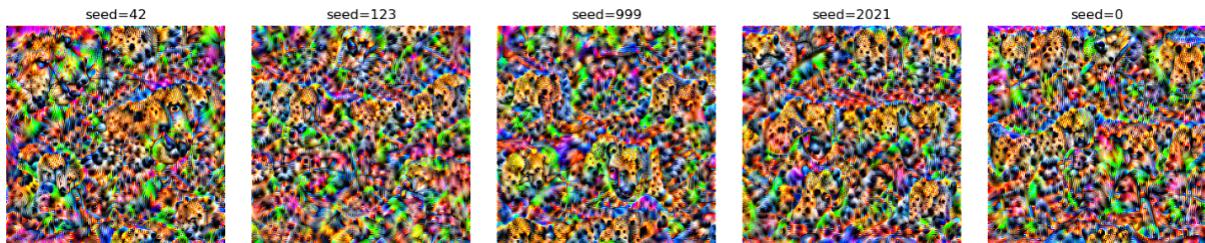
شکل ۵۳: بیشینه کردن logit چیتا

Class 293 Visualization (Improved)



شکل ۵۴: بیشینه کردن logit چیتا با random shift و tv

به صورت واضح طرح بدن و حتی صورت چیتا مشخص است.



شکل ۵۵: تولید تصویر با seed های مختلف

برای تولید تصویری که بیشترین پاسخ را برای کلاس cheetah (شماره ۲۹۳) در مدل VGG16 ایجاد کند، علاوه بر بهینه‌سازی مستقیم ورودی برای ماکریم کردن نورون خروجی آن کلاس، از دو تکنیک کمکی استفاده می‌شود:

۱. **Total Variation Regularization**: با اضافه کردن جمله‌ای که اختلاف میان پیکسل‌های مجاور را جرمیه می‌کند، از ایجاد نویز فرکانس بالا جلوگیری شده و تصویر نهایی یکنواخت‌تر و روان‌تر می‌شود. این عمل به برجسته‌شدن الگوهای ساختاری (مثل نقش‌های بدن چیتا) کمک می‌کند.

۲. **Random Shift (Jitter)**: در هر گام به روزرسانی ورودی، تصویر را به طور تصادفی چند پیکسل در جهت‌های مختلف جابه‌جا می‌کنیم و سپس دوباره به موقعیت اصلی بازمی‌گردانیم. این تغییر مکان کوچک موجب می‌شود مدل بر ویژگی‌های مکانی ثابت (و نه لبه‌های مصنوعی در مرز تصویر) حساس‌گردد و الگوهای کلی تر شی مورد نظر تقویت شوند.

ترکیب این دو روش باعث می‌شود به جای نویز پراکنده و پرهزینه، ساختارهای گویا و قابل تفسیر (مثلاً خطوط نرم بدن ببر، بافت پوست و الگوهای راهراه) در تصویر برجسته شوند و در نتیجه خروجی بهینه‌شده برای کلاس مورد نظر از نظر بصری معنادار‌تر گردد.