چالش های هشتمین دوره ICT Challenge

چالش اول: بد افزار

توضیح پروژه

در این پروژه به مسئله دسته بندی دوکلاسه خواهیم پرداخت. این مسئله یکی از پرکاربردترین و در عین حال قدیمی‌ترین کار (task)های یادگیری ماشین و هوش مصنوعی محسوب می‌شود. در مسائل دسته‌بندی دوکلاسه، یک کلاس مثبت) هدف، (targetو یک کلاس منفی داریم که می‌خواهیم با استفاده از ویژگی داده‌های موجود و الگوریتم‌های یادگیری ماشین میان آن دو تمایز قائل شویم. بعد از گرفتن خروجی الگوریتم برای داده ها، پیش‌بینی الگوریتم را با مقدار واقعی مورد انتظار برای آن داده مقایسه کرده و میزان خطایی که الگوریتم ایجاد کرده است را محاسبه می‌کنیم تا با استفاده از آن بتوانیم، الگوریتم یادگیر خود را در پیش بینی درست تر نتیجه، آموزش دهیم. در این مسابقه شما قرار است الگوریتم یادگیری ماشینی را توسعه دهید که بتواند میان فایل‌های اجرایی مفید (benign) یک سیستم اطلاعاتی و بدافزارها (maleware) تشخیص دهد. این تشخیص کمک شایانی به برنامه‌های آنتی‌ویروس خواهد کرد. در ادامه روند مسابقه را شرح می‌دهیم.

مجموعه داده

در این پروژه مجموعه داده هایی که قرار است الگوریتم یادگیری ماشین خود را برروی آن آموزش دهید، درون پوشه Data و دو فایل txt. با نام‌های malewares و benigns قرار داده‌ایم. فایل‌های txt شامل اطلاعاتی است که بدافزارها را از فایل‌های اجرایی مفید (benign) جدا می‌کنند. از تمامی رکوردهای این فایل‌های txt برای آموزش الگوریتم خود بهره ببرید.

برای هر کدام از رکوردهای موجود، ۱۷۵۲ ویژگی داریم. مهندسی ویژگی‌های مناسب در این باره می‌تواند با کم کردن ابعاد داده و رسیدن به ویژگی‌های مناسب‌تر، کار را برای الگوریتم‌های ماشین آسان‌تر کند.

در مجموع با در نظر گرفتن داده های هر دو دسته، ۱۲۰ هزار داده آموزشی داریم . (به صورت دقیق‌تر ۸۵هزار داده آموزشی برای maleware و ۳۵ هزار برای benign). این داده ها به طور مساوی میان دو دسته پخش نشده‌اند و تعداد داده های دسته-maleware بیشتر از benign است.

در کنار این فایلها، فایل Eval.txt نیز به شما داده شده است که شامل ۲۰ هزار رکورد است. شما قرار است بعد از آنکه مدل خود را برروی داده‌های آموزشی آموزش دادید، برروی داده‌های Eval.txt ارزیابی کنید. این فایل شامل تعداد مساوی از maleware ها و benign هاست اما بر روی رکوردهای آن، برچسب کلاسی که رکورد جاری به آن تعلق دارد، قید نشده است.

فرایند پروژه

ابتدا باید بتوانید مجموعه داده را از حالت text درآورده و به فرمت موردنیاز برای الگوریتم های یادگیری ماشین، تبدیل کنید.

سپس سعی کنید اگر می‌توانید با مهندسی ویژگی، ویژگی‌های مناسب را از درون داده های خود استخراج کنید تا بتوانید مدل‌های مناسب را آموزش دهید.

در این پروژه، هدف آن است که الگوریتم یادگیری ماشین با تشخیص بدافزارها به سیستم های آنتی ویروس در برخورد با آن‌ها کمک کند. بنابراین مهم است که الگوریتم، فایل‌های اجرایی مفید (benign) را به عنوان بدافزار تشخیص ندهد. زیرا امکان دارد یکی از فایل‌های مهم سیستم را بدافزار تشخیص داده و خسارت زیادی را به سیستم هدف وارد کند. سعی کنید مدل یادگیری ماشین قدرتمندی را آموزش دهید که بتواند این مسئله را به خوبی حل کند. بنابراین مدل شما باید precision بالایی (نرخ تشخیص فایل اجرایی مفید به عنوان بدافزار پایین) داشته باشد.

سعی کنید در برخورد با این مسئله، نامساوی بودن دو کلاس را در نظر بگیرید و از عملکرد قدرتمند مدل خود اطمینان حاصل فرمایید) در مسائلی که با imbalance label روبرو هستیم امکان دارد مدل با یادگرفتن و overfit شدن برروی کلاس برتر، دقت زیادی را ثبت کند اما در واقع نه تنها دقت بالایی ندارد بلکه موجب به بار آمدن خسارات فراوان نیز می‌شود.

در این مسئله، برای شما یک مدل baseline اولیه و ساده آماده شده است. این مدل برروی داده‌های آموزشی خام و با یک مدل Decision Treeساده آموزش دیده است .

در فاز اول پروژه، آنچه از شما انتظار می‌رود این است که برروی داده‌های آموزشی، مدل خود را آموزش دهید. سپس مدل آموزش دیده را بر روی داده‌های فایل Eval.txt ارزیابی کرده و نتایج ارزیابی را به فرمت زیر درآورده، خروجی‌ها و فایل مدل آموزش دیده را برای تیم پشتیبانی ارسال کنید. تیم پشتیبانی برای معیارهای عملکردی مدل شما را به شما گزارش خواهد کرد و شما باید بتوانید در این مرحله مدل baseline را پشت سر بگذارید. تیم‌هایی از این مرحله به مرحله‌ی بعدی صعود خواهند کرد که بتوانند مدلی را آموزش دهند که به این مهم دست یابد.

فرمت دریافت خروجی

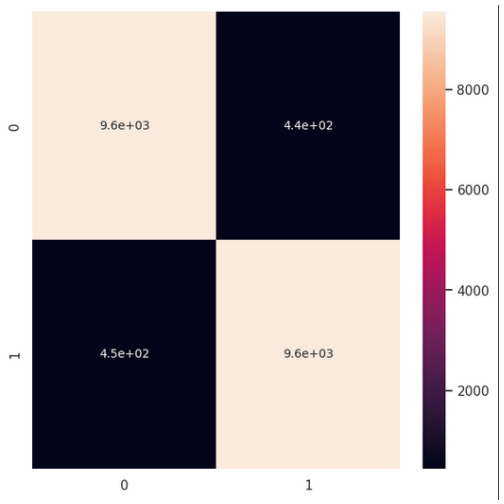
یک فایل csv. بدون header و index که مقدار سطر i ام کلاس پیش بینی شده توسط مدل برای i_Sample درون فایل Eval.txt را نشان می‌دهد.

به مثال زیر توجه کنید:

	A
1	1
2	1
3	0
4	1
5	0
6	0
7	0
8	1

در شکل زیر معیارهای مختلف برای مدل baseline را آورده‌ایم:

```
F1 Score: 0.956
Precision Score: 0.956
Recall Score: 0.955
Accuracy Score: 0.956
```




```
AUC_ROC Score: 0.956
```

فازهای پروژه

فاز اول:

در این فاز پروژه، مجموعه داده‌های داده شده را خوانده و به فرمت مورد نظر ببرید، سپس مدل‌های یادگیری ماشین خود را برروی داده‌های آموزشی آموزش داده و برروی داده‌های ارزیابی، ارزیابی کرده و خروجی تولید شده به فرمت گفته شده ذخیره کنید. (مدل‌هایی که با کتابخانه‌ی scikit-learn آموزش داده‌اید را بصورت pickle ذخیره کرده و ارسال کنید). بهترین مدل را انتخاب کرده و فاز دوم را شروع کنید.

فاز دوم:

در این فاز شما بهترین مدلی را که در فاز قبل بدست آورده‌اید را باید روی تمام ۱۴۰ هزار داده (هم آموزشی و هم ارزیابی که داده‌های ارزیابی به همراه برچسب واقعی به شما داده خواهند شد) آموزش دهید و سعی کنید عملکرد مدل نهایی را ارتقا دهید. در نهایت، تنها کافی است مدل نهایی را به عنوان خروجی در گیت قرار داده تا برروی یک مجموعه داده نهایی _که متفاوت از داده هایی است که در دست دارید_ تست شده و performance نهایی مدل شما بدست آید.

توجه داشته باشید

- ارزیابی تیم ها علاوه بر ملاک های معمول داوری، به دو معیار precision و recall بستگی دارد. اما از آنجایی که در این مسئله پایین بودن نرخ False positive از اهمیت بسیار بالایی برخوردار است تا سیستم فایل اجرایی مفیدی را به عنوان بدافزار تشخیص ندهد، precision در این مسئله وزن بیشتری دارد و سعی کنید مدل‌هایی را آموزش دهید که دارای precision بالایی باشند.
- هر چه این معیار پایین‌تر و به ۰ نزدیک‌تر باشد بهتر است. یعنی

$$metric = 1000 * (1 - precision) + (1 - recall)$$

$$= 1000 * \left(1 - \frac{TP}{TP + FP}\right) + (1 - \frac{TP}{TP + FN})$$

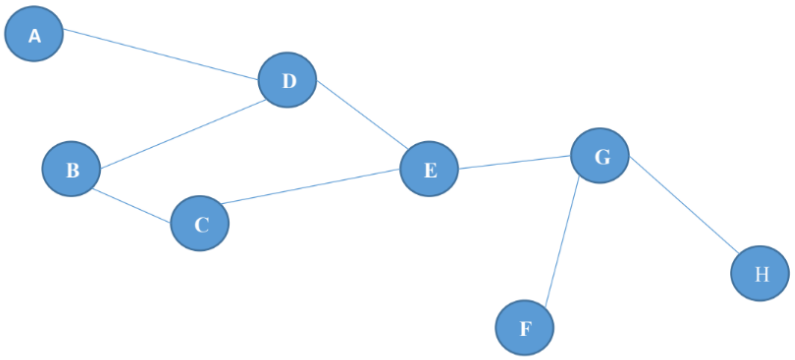
- در اینجا منظور از TP (True Positive) تعداد نمونه‌هایی است که مدل کلاس آن‌ها را درست تشخیص داده شده است.
- FPکه همان (False positive) است تعداد نمونه‌هایی است که مربوط به کلاس ۱ بودند اما ۰ در نظر گرفته شده‌اند.
- FN یا (False negative) نمونه‌هایی هستند که به کلاس ۰ تعلق داشتند اما مدل کلاس آن‌ها را ۱ تشخیص داده شده است.
- در این مسئله کلاس malware همان کلاس ۰ و کلاس benign کلاس ۱ است.
- رابط کاربری لازم نیست.
- در هنگام ارائه کد درgit ، لطفا Readme مناسب و حرفه ای که شرح روند کار شما، انتخاب های شما از لحاظ تکنولوژی و ابزار و همچنین چگونگی اجرای پروژه میباشد را بارگزاری فرمایید.
- پیاده سازی ناکامل و عدم پیاده سازی تمام بخش‌ها، با نظر داوران به نسبت امتیاز تعلق می‌گیرد.
- لطفا از صحت پاسخ مطمئن شده و سپس اقدام به ارسال پاسخ نمایید.

چالش دوم: سیستم توزیع شده

تولید یک سیستم محتوای اشتراکی به روشSystem Distributed

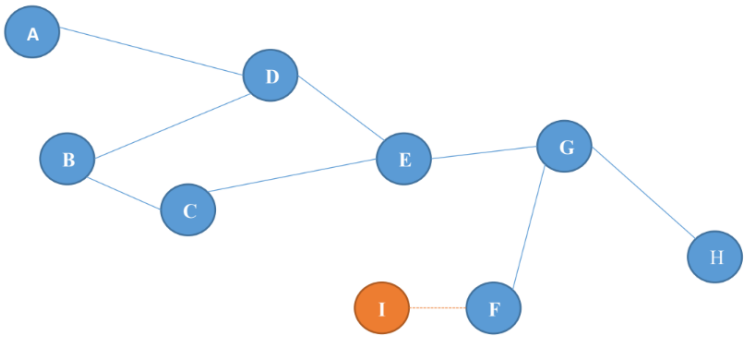
گراف و توپولوژی

در این پروژه هدف ایجاد یک سیستم اشتراک گذاری متن و تصویر بین کاربران می‌باشد. گراف زیر را در نظر بگیرید:



این گراف یک گراف هم بند است که یک مثال از توپولوژی شبکه ما می‌باشد.

توپولوژی تصویر شده فقط یک مثال است و مساله باید برای تمامی توپولوژی های هم بند کار کند. امکان اضافه کردن نود باید در مساله وجود داشته باشد. این کار از طریق درخواست به یکی از نودهای والد انجام می‌شود. به عنوان مثال اگر I به نود F درخواست بدهد و F آنرا قبول کند توپولوژی به صورت زیر در می‌آید.



نودها در این مساله از دو تا ۱۰۰۰ عدد می‌توانند باشند. هر نود دارای سه مشخصه زیر می‌باشد:

۱. IP Address

۲. Port Number

۳. Unique Name

به هنگام اضافه شدن یک نود، نام، IPو Port آن باید چک شود و در صورت تکراری بودن از اضافه شدن آن جلوگیری شود.

فاز یک

در این مساله در نظر بگیرید تمامی نودهای A تا H میتوانند برای یک دیگر یک متن را ارسال کنند. این سیستم ارسال متن مانند یک سیستم چت محلی میباشد. در این سیستم چت امکان ارسال متن به صورت Broadcast و Unicast وجود دارد. به عنوان مثال نود H میتوانند برای یک متن چت خصوصی ارسال کند و سایر نودها آنرا نمیتوانند ببینند. علاوه بر این نود H برای تمامی نودهای دیگر نیز میتوانند یک پیام ارسال کنند.

موارد زیر در این فاز باید مد نظر قرار بگیرند:

این پروژه فاقد دیتابیس مرکزی می‌باشد.

برای انتقال داده باید از Programming Socket استفاده شود.

نودها میتوانند در یک سیستم باشند.

پروژه باید قسمتی برای ورود فایل توپولوژی داشته باشد که برای تست بتوان توپولوژیهای متفاوت را به آن داد. بعد از ارسال پیام تا زیر یک ثانیه باید پیام توسط کاربر مقصد دیده شود.

پیام با ریست برنامه نباید حذف شود.

فاز دو

در این فاز کاربران میتوانند تصاویر را به اشتراک بگذارند. این تصاویر در نود محلی ذخیره شده اما مشخصات آن برای سایر نودها قابل مشاهده میباشد. به عنوان مثال نود A یک عکس را به اشتراک گذاشته که Description آن Tower Eiffel میباشد. نود H با جستجوی کلمه Eiffel متوجه وجود این عکس در نود A میشود که با انتخاب آن در این فاز، موارد زیر باید در نظر گرفته شوند:

مثال از نتیجه جستجوی کلمه Eiffel در نود H

Description	Node	Download	Size
The Eiffel Tower	A	Click	1K
Eiffel in Pars	B	Click	2K
Gustave Eiffel	E	Click	5K

این پروژه فاقد دیتابیس مرکزی است.

سیستم باید از اشتراک فایلهای تکراری جلوگیری کند.

حذف از اشتراک باید امکان پذیر باشد.

جستجو باید زیر یک ثانیه انجام شودب آن عکس از نود A برای نود H قابل نمایش میشود.

امتیازهای مجزا

قسمتهای زیر در صورت پیاده سازی امتیاز مجزا دارند:

پیاده سازی یک سیستم AAA

پیاده سازی یک سیستم Billing اعتباری برای تصاویر (به صورت پیش فرض هر کاربر ۱۰۰ Coin اعتبار دارد)

برنامه نویسی که بیشترین تعداد نود را برای تست نمایش دهد امتیاز ویژه دارد

توجه داشته باشید

در هنگام ارائه کد درgit ، لطفا Readme مناسب و حرفه ای که شرح روند کار شما، انتخاب های شما از لحاظ تکنولوژی و ابزار و همچنین چگونگی اجرای پروژه می‌باشد را بارگزاری فرمایید.

پیاده سازی ناکامل و عدم پیاده سازی تمام بخش‌ها، با نظر داوران به نسبت امتیاز تعلق می‌گیرد.

لطفا از صحت پاسخ مطمئن شده و سپس اقدام به ارسال پاسخ نمایید.

توجه داشته باشید

- رابط کاربری لازم نیست.
- در هنگام ارائه کد درgit ، لطفا Readme مناسب و حرفه ای که شرح روند کار شما، انتخاب های شما از لحاظ تکنولوژی و ابزار و همچنین چگونگی اجرای پروژه میباشد را بارگزاری فرمایید.
- پیاده سازی ناکامل و عدم پیاده سازی تمام بخش‌ها، با نظر داوران به نسبت امتیاز تعلق می‌گیرد.
- لطفا از صحت پاسخ مطمئن شده و سپس اقدام به ارسال پاسخ نمایید.

چالش سوم: توربین

طراحی سیستم کنترل مصرف سوخت در یک توربین

یک توربین نیروگاهی بطور متوسط در هر ثانیه ۱۳ تا ۱۴ لیتر گازوئیل میسوزاند. دمای خروجی اگزوز حدود ۵۳۰ درجه سانتیگراد است. دورگردش معمول ۳۰۰۰ دور بر دقیقه (rpm) است. جرم گاز خروجی از اگزوز حدود ۵۰۰ کیلوگرم بر ثانیه است. یکی از چالش هایی که امروزه در نیروگاه ها مطرح میشود بهینه سازی مصرف سوخت توربین است، بطور معمول بازده کلی تولید برق توربین حدود ۳۴ درصد است و با بهینه سازی مصرف سوخت میتوان بازده سیستم را افزایش داد. رابطه واقعی میزان مصرف سوخت توربین به پارامترهای مختلفی وابسته است و هدف این مسئله طرح یک فرمول ساده، بدون وابستگی به میزان برق تولیدی، برای حجم مصرفی گازوئیل و پیادهسازی سیستم کنترل شبیهسازی شده توسط شما است. این شبیه سازی شامل دو بخش سطح نیروگاه (field) و سطح سیستم کنترل است.

طرح مسئله:

فرض کنید که در یک نیروگاه حرارتی میزان سوخت مصرفی توربین توسط یک دریچه (Valve) کنترل میشود. بطور ساده میزان سوخت مصرفی در هر لحظه به دما، میزان مصرف در لحظه قبلی، دور گردش توربین و میزان گاز خروجی بستگی دارد که از طریق رابطه زیر محاسبه میشود:

$$V(t + 1) = aV(t) + bT(t) + cS(t) + dG(t)$$

که در آن V حجم مصرفی سوخت بر حسب لیتر، Tدما بر حسب درجه سانتیگراد، Sدور گردش برحسب دور بر ثانیه، و G جرم گاز خروجی بر حسب کیلوگرم در ثانیه میباشد. و زمان بر حسب ثانیه است. همچنین a, b, c, d ضرایب ثابت میباشند. یک سیستم کنترل در هر لحظه بر اساس مقدار V میزان باز بودن دریچه را محاسبه میکند. در ابتدا یک سیستم نرم افزاری طراحی کنید که بر روی سیستم کنترل توربین بصورت مقیم نصب شود و این قابلیت را داشته باشد که داده های مورد نیاز را ذخیره کرده و بر اساس رابطه بالا مقدار V در ثانیه t+1 ام را محاسبه کند و بر همین اساس میزان باز بودن دریچه سوخت را تنظیم نماید.

پس از اتمام این پروژه، یک سیستم کنترل دیگر در سمت سایت و در جایی که مهندسان کنترل مستقر هستند طراحی کنید. اطلاعات ذخیره شده در سمت توربین در هر ثانیه بصورت time Real باید به این سیستم منتقل شوند و در یک نرم افزار مانیتورینگ قابل نمایش به مهندسان سیستم کنترل باشد. مهندس سیستم کنترل بر اساس مقادیر مشاهده شده و میزان مصرف سوخت میتوانند دستور توقف توربین را به توربین ارسال کنند تا در همان لحظه توربین خاموش شود. همچنین کلیه اطلاعات دریافتی از توربین در سمت مانیتورینگ باید در دسترس باشد. توجه شود که داده ها باید در زمان تضمین شده به دست مهندس سیستم کنترل برسد تا او بتواند تصمیمات لازم را در زمان مقتضی اتخاذ کند. مهندسین مستقر در سایت دارای دو سطح دسترسی میباشد. سطح دسترسی اول تنها میتواند اطلاعات را مشاهده کند. سطح دسترسی دوم علاوه بر مشاهده اطلاعات میتواند دستور پاک کردن اطلاعات و Shutdownرا نیز صادر کند و یا کاربری با سطح دسترسی تعریف شده را به سیستم بیافزاید.

برای تضمین Time Real بودن ارسال اطلاعات از چه مکانیزمی استفاده میکنید؟

سطوح دسترسی در سایت را چگونه کنترل و پیاده سازی میکنید؟

برای ارسال امن اطلاعات از سمت توربین به سایت از چه مکانیزم و تکنولوژی استفاده میکنید؟

سیستم کنترل سمت توربین با چه مکانیزمی مطمئن میشود که دستور Shutdown از سوی کاربری صادر شده که سطح دسترسی مورد نظر را دارد؟

در سمت سیستم کنترل، چه سرویس هایی پیاده سازی میکنید؟ ارتباط آنها چگونه است؟ آیا به Availability فکر کرده اید؟

نظر شما در مورد سیستم هوشمند تنظیم دریچه سوخت چیست؟ آیا میتوانید سیستم هوشمند ساده ای طراحی کنید تا ضرایب معادله را براساس داده های قبلی و برخط بصورت بهینه محاسبه نماید؟ (این سیستم هوشمند را تشریح نمایید و در صورتیکه زمان داشتید پیاده سازی کنید)

توجه داشته باشید

- در هنگام ارائه کد درgit ، لطفا Readme مناسب و حرفه ای که شرح روند کار شما، انتخاب های شما از لحاظ تکنولوژی و ابزار و همچنین چگونگی اجرای پروژه میباشد را بارگزاری فرمایید.
- پیاده سازی ناکامل و عدم پیاده سازی تمام بخش‌ها، با نظر داوران به نسبت امتیاز تعلق می‌گیرد.
- لطفا از صحت پاسخ مطمئن شده و سپس اقدام به ارسال پاسخ نمایید.

چالش چهارم: خودرو های الکتریکی

فرض کنید یک میلیون خودروی برقی متصل (Connected Cars) وجود دارند که پیوسته در حال ارسال داده های عملکردی و خطاهای خود به سرور هستند. می خواهیم رفتار رانندگی رانندگان را تحلیل کنیم. داشتن سرعت غیرمجاز و برخی مقادیر پارامترهای دیگر مانند سرعت جانبی و جرک (مشتق شتاب) بر روی تحلیل رفتار تاثیرگذارند.

قصد داریم هم در بازه های زمانی مشخص و هم در هر سفر، رفتار راننده را تحلیل و اقدام متناسب را انجام دهیم. ما سابقه سفر رانندگان را ذخیره می‌کنیم. نیاز است هر سفر که از لیست سفرها انتخاب می‌شود، بر روی نقشه مسیر آن رسم شود و مناطقی که رانندگی پرخطر اتفاق افتاده، با رنگ قرمز نشان داده شود؛ مشابه حالتی که بین دو نقطه مسیریابی می‌کنیم و مناطق پرترافیک به رنگ قرمز نشان داده می‌شوند.

پیشنهادات پروتکل ارتباطی و نوع تبادل داده و توپولوژی تبادل داده و موارد مربوط به معماری از موارد مطرح در این چالش هستند.

یک شبیه‌ساز دارای ۱۰ خودرو بنویسید و حاصل کار خود را به صورت بصری نشان دهید.

هم سمت کلاینت، هم داشبورد و هم سرور مدنظر هستند.

فرض کنید:

- حداکثر سرعت مجاز: ۱۲۰ km/h کیلومتربرساعت)
- حداکثر سرعت جانبی: ۱ rad/s رادیان بر ثانیه)
- بازه جرک j :بین منفی ۳ m/s^3تا مثبت ۳m/s^3
- APIنقشه مورد استفاده: آزاد
- تعریف سفر: بازه بین دو توقف خودرو (سرعت صفر)

توجه داشته باشید

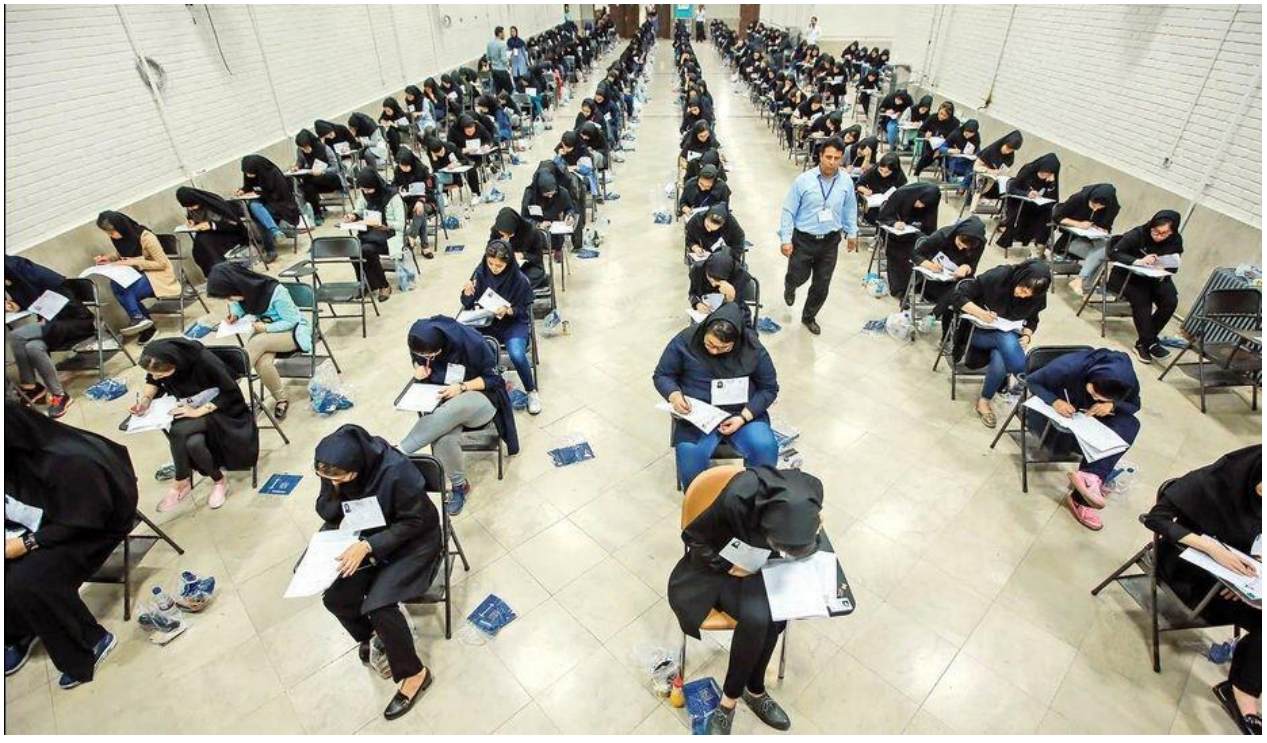
- در هنگام ارائه کد درgit ، لطفا Readme مناسب و حرفه ای که شرح روند کار شما، انتخاب های شما از لحاظ تکنولوژی و ابزار و همچنین چگونگی اجرای پروژه میباشد را بارگزاری فرمایید.
- پیاده سازی ناکامل و عدم پیاده سازی تمام بخش‌ها، با نظر داوران به نسبت امتیاز تعلق می‌گیرد.
- لطفا از صحت پاسخ مطمئن شده و سپس اقدام به ارسال پاسخ نمایید.

چالش پنجم: تشخیص تقلب در آزمون‌های تستی آنلاین

در عصر دیجیتال، فرصت‌های آموزشی بیشتری از طریق آزمون‌های آنلاین و کلاس‌های مجازی در دسترس قرار گرفته‌اند. اما با این فرصت‌ها، چالش‌های جدیدی نیز به وجود آمده‌است. یکی از این چالش‌ها، مبارزه با تقلب است، تقلب در آزمون‌ها مسئله‌ای جدی و پیچیده است که تمام سطوح آموزشی را تحت تأثیر قرار می‌دهد. از آموزش مقدماتی تا دانشگاه، از کلاس‌های حضوری تا آنلاین، هیچ کس از این چالش مصون نیست .

این چالش در آزمون‌های تستی آنلاین به ویژه برجسته است، زیرا فرصت‌های بیشتری برای تقلب وجود دارد. با دسترسی به اینترنت، شرکت‌کنندگان می‌توانند به طور مستقیم به منابع خارجی مراجعه کنند، پاسخ‌ها را از دیگران بگیرند، یا حتی از برنامه‌های کمکی استفاده کنند.

به همین دلیل، ما نیاز به راه‌حل‌های خلاقانه و پیشرفته داریم تا با این مشکل مقابله کنیم. در این چالش، شما دعوت می‌شوید تا یک راهکار برای تشخیص تقلب در آزمون‌های تستی آنلاین طراحی و توسعه دهید و با تکیه بر توانایی های برنامه‌نویسی و تحلیل داده‌های خود، می‌توانید به حل این معضل جدی کمک کنید.



آدرس دبیرخانه: خیابان آزادی،جنب دانشگاه صنعتی شریف، کوچه قدیر، پلاک ۵، واحد ۲
شماره تماس: ۵-۶۶۰۲۸۹۶۳-۰۲۱
www.ictchallenge.ir

برای کمک به فکر خلاقانه‌تان، برخی روش‌های متداول را بررسی می‌کنیم:

تحلیل الگوهای پاسخ‌دهی: یکی از راه‌هایی که می‌توانیم برای شناسایی تقلب استفاده کنیم، بررسی الگوهای پاسخ‌دهی است. این می‌تواند شامل تجزیه و تحلیل زمان پاسخ به سوالات، ترتیب پاسخ به سوالات و حتی الگوهای خاص از پاسخ‌های درست و نادرست باشد.

مقایسه پاسخ‌ها: با مقایسه‌ی پاسخ‌های فرد با پاسخ‌های دیگر شرکت‌کنندگان، می‌توان تشخیص داد اگر تقلب رخ داده است یا نه. به عنوان مثال، اگر دو یا چند فرد پاسخ‌های مشابه یا یکسانی را ارائه دهند، ممکن است این نشان‌دهنده‌ی تقلب باشد

تحلیل تغییرات ناگهانی در عملکرد: اگر عملکرد یک فرد ناگهان تغییر کند _مثلا، اگر فردی که معمولاً نمرات پایینی می‌گیرد، ناگهان نمرات بسیار بالایی بگیرد_ ممکن است این نشان‌دهنده‌ی تقلب باشد

لازم است با طراحی سامانه آزمونی، الگوریتم ها و روش های خلاقانه کشف تقلبی که طراحی کرده اید را به نمایش بگذارید.

توجه داشته باشید

رابط کاربری لازم است.

داکیومنت های لازم برای ارائه آنچه حاضر نموده اید ایجاد کنید.

در هنگام ارائه کد درgit ، لطفا Readme مناسب و حرفه ای که شرح روند کار شما، انتخاب های شما از لحاظ تکنولوژی و ابزار و همچنین چگونگی اجرای پروژه میباشد را بارگزاری فرمایید.

پیاده سازی ناکامل و عدم پیاده سازی تمام بخش‌ها، با نظر داوران به نسبت امتیاز تعلق می‌گیرد.

لطفا از صحت پاسخ مطمئن شده و سپس اقدام به ارسال پاسخ نمایید.