AI Descision Tree Project Report

ابتدا وب‌سایت Kaggle را برای پیدا کردن یک دیتاست مناسب جستجو کردم. بعد از جستجو، دیتاستی با عنوان "Flight Passenger Satisfaction"پیدا کردم.

با اســتفاده از gitHubو Colab ، دیتاســت را به‌صــورت یک DataFrame (df) خواندم و آن را پاک‌ســازی (clean) کردم. پس از پاک‌سازی، یک df بدون مقادیر غیرعدد صحیح (non-integer) داشتم و ستون‌های پیوسته (continuous columns) را دسته‌بندی کردم.

پس از پاک‌سازیdf ، داده‌ها را به سه بخشtrain ، test، و valid تقسیم کردم و سپس برای هرکدام، ویژگی‌ها (X) و برچسب‌ها (y) را جدا نمودم.

اما دوباره به جستجو در Kaggle پرداختم و دیتاست دیگری برای پیش‌بینی دیابت بر اساس سن، BMIو غیره پیدا کردم.

ابتدا آن را پاک‌سازی کردم. این دیتاست قبلاً به کلاس‌های عددی نگاشت شده بود، بنابراین نیاز به نگاشت دوباره نداشت.

ستون‌های پیوسته را دسته‌بندی کردم.

برای پیش‌بینی دقیق‌تر، از نمودار دسته‌بندی BMI استفاده کردم.

پس از پاک‌سازی داده‌ها، کلاس مدل را برای ساخت مدل DecisionTree نوشتم.

پس از بررسی خروجی مدل(predictions) ، متوجه شدم داده‌هایم بیش‌ازحد ساده‌اند و نتوانستم دقت خوبی به‌دست بیاورم.

(.بود training و ۷۶٪ روی مجموعه‌ی validation بیشترین دقت فقط ۷۴٪ روی مجموعه‌ی)

تصمیم گرفتم دیتاستم را دوباره تغییر دهم، و پس از تغییر دیتاست، پاک‌سازی آن، و ساخت مدل، متوجه شدم دقت مدل بسیار بالاست.

در نتیجه، به بررســی مجدد در Kaggle پرداختم و متوجه شــدم این موضــوع طبیعی اســت. در واقع، مدل من با دقت ۹۸٪ دچار underfittingبود.

پس از انجام grid search روی مدل، بهترین مدل را با دقت ۹۹.۶۰٪ پیدا کردم.

دلیل این دقت بسیار بالا این است که دیتاست من برای شناسایی نوع حمله یا درخواست عادی به یک وب‌سایت استفاده می‌شود.

ما پارامترهای زیادی داریم، و این پارامترها درخواست‌های عادی را از درخواست‌های حمله جدا می‌کنند.

به همین دلیل، درخواست‌های حمله همیشه نشانه‌هایی دارند که آن‌ها را قابل‌شناسایی می‌سازد.