Model Card + Evaluation Report

(MetroGo – AI/ML Decision Support System)

مقدمه‌ی چرا این سند برای MetroGo حیاتی است

در سامانهMetroGo ، مدل‌های یادگیری ماشین نقش «ابزار کمکی» صرف ندارند، بلکه به‌صورت مستقیم در زنجیره تصمیم‌سازی کاربر مداخله می‌کنند. حتی اگر تصمیم نهایی توسط کاربر گرفته شود، پیشنهادهای ارائه‌شده توسط سیستم می‌توانند رفتار کاربر را به‌شدت تحت تأثیر قرار دهند. به همین دلیل، هر خطای الگوریتمی یا فرض نادرست در مدل، می‌تواند اثرات تجمعی در مقیاس شهری ایجاد کند.

این سند با این فرض نوشته شده است که مدل‌ها نه بی‌خطا هستند و نه ذاتاً قابل اعتماد. اعتماد به مدل باید کسب شود، مستند شود و محدود شود Model Card .دقیقاً ابزاری است برای شفاف‌سازی این اعتماد: اینکه مدل چه کاری می‌تواند انجام دهد، چه کاری را نباید انجام دهد، و در چه شرایطی نباید به آن اتکا کرد.

در پروژه‌های شهری، نبود چنین سندی معمولاً به دو حالت منجر می‌شود: یا مدل بیش‌ازحد بزرگ‌نمایی می‌شود و شکست پرهزینه رخ می‌دهد، یا مدل کنار گذاشته می‌شود چون کسی به آن اعتماد ندارد MetroGo .تلاش می‌کند مسیر سومی را طی کند: استفاده آگاهانه، محدود و مسئولانه از AI.

۱ .فرآیند آموزش مدل(Expanded Model Training Description)

فرآیند آموزش مدل در MetroGo به‌عنوان یک فعالیت «یک‌باره» یا آزمایشی در نظر گرفته نشده است، بلکه بخشی از چرخه عمر محصول محسوب می‌شود. از همان ابتدای طراحی، این فرض وجود داشته که داده‌های رفتاری کاربران و شرایط شهری پویا هستند و مدل باید توان تطبیق تدریجی با این تغییرات را داشته باشد.

آموزش مدل بر اساس داده‌های تاریخی انجام می‌شود که از نظر زمانی برچسب‌گذاری شده‌اند. این موضوع امکان تحلیل تغییرات عملکرد مدل در بازه‌های مختلف زمانی را فراهم می‌کند. داده‌ها پیش از آموزش از چند فیلتر عبور می‌کنند: حذف داده‌های ناقص، شناساییoutlier ها، و بررسی هم‌خوانی داده با دامنه مسئله. این مرحله اهمیت ویژه‌ای دارد زیرا کیفیت مدل بیش از آنکه تابع الگوریتم باشد، تابع کیفیت داده است.

در MetroGo تصمیم گرفته شده که مدل‌ها به‌صورت کنترل‌شده و قابل بازتولید آموزش داده شوند. به همین دلیل، پارامترهای آموزش، نسخه داده و تنظیمات مدل همگی مستند می‌شوند تا امکان بازسازی نتایج وجود داشته باشد. این رویکرد از بروز حالتی جلوگیری می‌کند که مدل «تصادفی خوب» به نظر برسد ولی قابل تکرار نباشد.

۲ .متریک‌های ارزیابی: فراتر از اعداد خام

درMetroGo ، متریک‌ها صرفاً ابزار گزارش‌دهی نیستند، بلکه ابزار تصمیم‌گیری هستند. به همین دلیل، انتخاب متریک‌ها با دقت و با درنظرگرفتن کاربرد واقعی انجام شده است. هر متریک به این سؤال پاسخ می‌دهد: «این عدد دقیقاً چه چیزی از تجربه کاربر یا عملکرد سیستم را توضیح می‌دهد؟«

برای مثال، متریکی که میانگین خطا را کاهش می‌دهد اما توزیع خطا را ناپایدارتر می‌کند، ممکن است از نظر عددی بهتر باشد اما از نظر عملی بدتر عمل کند. به همین دلیل، تیم MetroGo همواره به توزیع خطا، و نه فقط مقدار میانگین آن، توجه کرده است.

همچنین عملکرد مدل در سناریوهای مختلف مقایسه شده است تا مشخص شود آیا مدل به‌صورت یکنواخت عمل می‌کند یا در برخی شرایط خاص دچار افت شدید می‌شود. این تحلیل‌ها به‌ویژه برای سیستم‌های شهری اهمیت دارند، زیرا رفتار کاربران در ساعات اوج، تعطیلات یا شرایط اضطراری تفاوت معناداری دارد.

۳ .ارزیابی مبتنی بر سناریوهای واقعی(Expanded)

ارزیابی مدل در MetroGo صرفاً محدود به دیتاست تست استاندارد نیست. سناریوهایی طراحی شده‌اند که بازتاب‌دهنده شرایط واقعی استفاده از سیستم هستند. این سناریوها شامل شرایطی هستند که داده‌ها ناقص، نویزی یا غیرمنتظره‌اند.

برای مثال، زمانی که بخشی از داده‌های شبکه مترو با تأخیر به‌روزرسانی می‌شود، مدل باید بتواند خروجی‌ای ارائه دهد که همچنان منطقی و قابل استفاده باشد. بررسی عملکرد مدل در چنین شرایطی به تیم کمک کرده است بفهمد مدل تا چه حد به «کامل‌بودن داده» وابسته است.

نتایج این ارزیابی‌ها نشان داده‌اند که برخی مدل‌ها اگرچه در شرایط ایده‌آل عملکرد خوبی دارند، اما در شرایط واقعی شکننده‌اند. این یافته‌ها باعث شده‌اند برخی مدل‌ها صرفاً برای تحلیل داخلی استفاده شوند و وارد مسیر تصمیم‌سازی کاربر نشوند.

۴ :Robustness .مقاومت در برابر تغییرات جزئی

Robustnessدر MetroGo به‌عنوان یکی از معیارهای کلیدی پذیرش مدل تعریف شده است. مدل باید در برابر تغییرات کوچک و طبیعی داده‌ها رفتار معقولی داشته باشد. اگر خروجی مدل با تغییر جزئی در ورودی به‌طور نامتناسب تغییر کند، این نشانه ضعف بنیادی در مدل است.

برای بررسی این موضوع، آزمایش‌هایی انجام شده که در آن داده‌های ورودی به‌صورت کنترل‌شده دچار نویز یا تغییر جزئی شده‌اند. تحلیل نتایج این آزمایش‌ها نشان داده که برخی ویژگی‌ها وزن بیش‌ازحدی در تصمیم‌گیری مدل دارند. شناسایی این ویژگی‌ها به تیم کمک کرده است تا ریسک وابستگی بیش‌ازحد به یک سیگنال خاص را کاهش دهد.

۵ :Bias. تحلیل عمیق‌تر سوگیری‌ها

سوگیری در MetroGo نه‌تنها به‌عنوان یک مسئله فنی، بلکه به‌عنوان یک مسئله اجتماعی در نظر گرفته شده است. داده‌های آموزشی بازتاب رفتار کاربرانی هستند که از سیستم استفاده کرده‌اند، نه کل جمعیت شهری. این تفاوت می‌تواند منجر به نادیده‌گرفتن نیازهای برخی گروه‌ها شود.

تحلیل bias به تیم کمک کرده بفهمد مدل در چه شرایطی ممکن است به‌صورت سیستماتیک برخی مسیرها، مناطق یا الگوهای رفتاری را کم‌اهمیت تلقی کند. این آگاهی باعث شده محدودیت‌های استفاده از مدل به‌صورت شفاف مستند شود.

۶ :Failure Modes. پذیرش شکست به‌عنوان واقعیت

در این سند به‌طور صریح پذیرفته شده که مدل‌ها شکست می‌خورند. مسئله اصلی این نیست که آیا مدل شکست می‌خورد یا نه، بلکه این است که آیا تیم می‌داند کِی و چرا شکست می‌خورد. شناسایی failure modeها باعث شده MetroGo بتواند برای هر حالت شکست، واکنش مناسب تعریف کند.

۷ .خطاهای بحرانی (Critical Errors) و پیامدهای سیستمی آن‌ها

در سامانهMetroGo ، همه‌ی خطاها ارزش و اثر یکسانی ندارند. یکی از اشتباهات رایج در پروژه‌های مبتنی بر یادگیری ماشین این است که تمام خطاها صرفاً به‌عنوان «کاهش دقت» یا «افت عملکرد» دیده می‌شوند، در حالی که در سیستم‌های واقعی، به‌ویژه سیستم‌های شهری و پرداخت‌محور، برخی خطاها می‌توانند پیامدهای زنجیره‌ای، حقوقی و حتی اجتماعی ایجاد کنند.

خطای بحرانی در MetroGo به خطایی اطلاق می‌شود که نه‌تنها باعث نارضایتی کاربر، بلکه موجب از دست رفتن اعتماد، بروز ریسک حقوقی یا ایجاد اختلال عملیاتی در مقیاس گسترده شود. برای مثال، اگر مدل در شرایط خاصی پیشنهاد نادرستی ارائه دهد که منجر به تصمیم اشتباه کاربر شود، حتی اگر این اتفاق به‌ندرت رخ دهد، اثر آن می‌تواند بسیار پرهزینه‌تر از صدها خطای کوچک باشد.

در تحلیل خطاهای بحرانی، تیم MetroGo به این نتیجه رسیده است که شدت اثر خطا از فراوانی آن مهم‌تر است. به همین دلیل، ارزیابی مدل صرفاً بر اساس میانگین متریک‌ها انجام نمی‌شود، بلکه سناریوهایی بررسی می‌شوند که در آن‌ها مدل ممکن است در شرایط خاص به‌شدت دچار خطا شود. این نگاه باعث شده است برخی خروجی‌های مدل به‌صورت مستقیم در اختیار کاربر قرار نگیرند و ابتدا از فیلترهای کنترلی عبور کنند.

یکی دیگر از ابعاد خطاهای بحرانی، اثر آن‌ها بر برند و اعتماد عمومی است MetroGo .به‌عنوان یک سرویس شهری، حتی در مراحل اولیه، با انتظارات متفاوتی نسبت به یک اپلیکیشن صرفاً سرگرمی مواجه است. بنابراین، خطایی که در یک محصول غیرحساس قابل چشم‌پوشی است، در اینجا می‌تواند به کاهش اعتماد کلی به پرداخت دیجیتال یا خدمات هوشمند شهری منجر شود.

۸ .سناریوهای شکست (Failure Modes) و تحلیل شرایط وقوع آن‌ها

پذیرش این واقعیت که مدل‌های یادگیری ماشین ذاتاً مستعد شکست هستند، یکی از بلوغ‌های مهم تیم MetroGoبوده است. به‌جای تلاش برای ساخت مدلی «بی‌نقص»، تمرکز اصلی بر شناسایی سناریوهایی قرار گرفته که مدل در آن‌ها بیشترین احتمال شکست را دارد.

Failure modeها در MetroGo نه به‌عنوان نقص، بلکه به‌عنوان ویژگی‌های قابل پیش‌بینی سیستم در نظر گرفته می‌شوند. برای هر مدل، شرایطی مشخص شده که در آن داده‌های ورودی از دامنه‌ای که مدل بر اساس آن آموزش دیده خارج می‌شوند. این خروج از دامنه می‌تواند ناشی از تغییر رفتار کاربران، تغییر سیاست‌های شهری یا حتی شرایط غیرمنتظره مانند اختلال‌های گسترده باشد.

تحلیل این سناریوها به تیم کمک کرده است بفهمد مدل در چه شرایطی نباید مبنای تصمیم‌گیری قرار گیرد. در چنین مواردی، سیستم به‌صورت آگاهانه به حالت محافظه‌کارانه‌تر بازمی‌گردد یا نقش مدل را به سطح توصیه غیرالزامی کاهش می‌دهد. این رویکرد از استفاده ناآگاهانه از خروجی مدل جلوگیری می‌کند.

یکی از نتایج مهم این تحلیل، تعریف مرزهای عملکرد قابل قبول برای مدل بوده است. به این معنا که مدل فقط زمانی فعال است که شرایط داده‌ای، رفتاری و سیستمی در محدوده‌ای قرار داشته باشند که عملکرد مدل در آن‌ها قبلاً ارزیابی و تأیید شده است.

۹ .تصمیم‌گیری‌های حساس و ضرورت کنترل‌های ایمنی(Safety Controls)

درMetroGo ، هرچند مدل‌های AI به‌طور مستقیم تصمیم نهایی را اتخاذ نمی‌کنند، اما خروجی آن‌ها می‌تواند تصمیم کاربر را جهت‌دهی کند. همین موضوع باعث می‌شود برخی خروجی‌ها در دسته تصمیم‌های حساس قرار بگیرند. تصمیم حساس به تصمیمی اطلاق می‌شود که خطا در آن می‌تواند منجر به ضرر مالی، بی‌اعتمادی کاربر یا پیامدهای حقوقی شود.

برای چنین تصمیم‌هایی، کنترل‌های ایمنی چندلایه طراحی شده‌اند. این کنترل‌ها شامل محدودسازی دامنه اثر خروجی مدل، الزام به تأیید انسانی در شرایط خاص، و استفاده از قواعد مبتنی بر منطق تجاری در کنار مدل هستند. هدف از این کنترل‌ها حذف نقش مدل نیست، بلکه جلوگیری از تصمیم‌گیری کورکورانه بر اساس خروجی الگوریتم است.

وجود این کنترل‌ها همچنین باعث شده است تیم بتواند با اطمینان بیشتری مدل را در محیط‌های واقعی آزمایش کند. بدون این لایه‌های ایمنی، هر تست میدانی می‌توانست ریسک بالایی برای اعتبار پروژه داشته باشد.

۱۰ .توضیح‌پذیری مدل (Explainability) و نقش آن در اعتمادسازی

توضیح‌پذیری در MetroGo صرفاً یک ویژگی فنی یا تزئینی نیست، بلکه یکی از ستون‌های اصلی پذیرش مدل در تیم و نزد ذی‌نفعان است. اگر تیم نتواند توضیح دهد چرا مدل به خروجی خاصی رسیده است، عملاً امکان دفاع از تصمیم‌ها، بهبود مدل یا پاسخ‌گویی به انتقادات وجود نخواهد داشت.

به همین دلیل، از ابتدا تلاش شده است مدل‌ها به‌گونه‌ای انتخاب و طراحی شوند که تفسیرپذیر باقی بمانند. حتی در مواردی که استفاده از مدل‌های پیچیده‌تر می‌توانست دقت را افزایش دهد، تصمیم گرفته شده است شفافیت و قابلیت توضیح بر دقت مطلق ترجیح داده شود.

توضیح‌پذیری همچنین نقش مهمی در شناسایی خطاهای پنهان داشته است. در چند مورد، بررسی دلایل خروجی مدل نشان داده که مدل به سیگنال‌هایی وابسته شده که از نظر منطقی نباید این‌قدر تأثیرگذار باشند. این کشف‌ها بدون ابزارهای توضیح‌پذیری امکان‌پذیر نبود.

۱۱ .محدودیت‌های استفاده (Usage Limitations) و جلوگیری از سوءبرداشت

یکی از بخش‌های کلیدی Model Card درMetroGo ، مستندسازی محدودیت‌های استفاده از مدل است. این محدودیت‌ها به‌صورت شفاف توضیح می‌دهند که مدل برای چه شرایطی طراحی شده و برای چه شرایطی مناسب نیست.

نبود این شفافیت معمولاً باعث می‌شود مدل در موقعیت‌هایی استفاده شود که خارج از دامنه اعتبار آن است. چنین استفاده‌ای نه‌تنها عملکرد مدل را زیر سؤال می‌برد، بلکه می‌تواند باعث تصمیم‌های نادرست و در نهایت بی‌اعتمادی به کل سیستم شود.

MetroGoبا مستندسازی این محدودیت‌ها تلاش کرده است از همان ابتدا انتظارات واقع‌بینانه‌ای نسبت به توانایی‌های AI ایجاد کند.

۱۲ .پایش پس از استقرار(Post-deployment Monitoring)

مدل پس از استقرار پایان نمی‌یابد؛ بلکه وارد مرحله‌ای می‌شود که شاید حتی مهم‌تر از آموزش اولیه باشد. درMetroGo ، عملکرد مدل به‌صورت مستمر پایش می‌شود تا تغییرات تدریجی در داده‌ها یا رفتار کاربران شناسایی شود.

این پایش به تیم امکان می‌دهد قبل از آنکه افت عملکرد به بحران تبدیل شود، مداخله کند. چنین رویکردی به‌ویژه در سیستم‌های شهری که تغییرات رفتاری تدریجی اما مداوم هستند، اهمیت بالایی دارد.

۱۴ .ارزیابی Robustness مدل در شرایط غیرایده‌آل(Stress & Edge Case Evaluation)

یکی از ضعف‌های رایج در گزارش‌های مدل این است که ارزیابی صرفاً روی داده‌های تمیز، کامل و مشابه داده‌های آموزش انجام می‌شود. در حالی که محیط واقعیMetroGo ، به‌خصوص در فاز عملیاتی شهری، سرشار از داده‌های ناقص، پرنویز و گاه متناقض است Robustness .مدل به این معناست که سیستم تا چه حد می‌تواند در مواجهه با چنین شرایطی همچنان رفتار قابل قبولی از خود نشان دهد.

درMetroGo ، ارزیابی robustness نه به‌عنوان یک تست تکمیلی، بلکه به‌عنوان بخشی از طراحی مدل در نظر گرفته شده است. داده‌هایی که در شرایط واقعی تولید می‌شوند، لزوماً همگن نیستند؛ رفتار کاربران در ساعات اوج ترافیک، در مناطق مختلف شهری، یا در شرایط خاص (مانند اختلال‌های مقطعی) می‌تواند تفاوت معناداری داشته باشد. مدل باید بتواند بدون فروپاشی عملکرد، این تغییرات را تحمل کند.

برای همین، سناریوهایی تعریف شده‌اند که عمداً داده‌های ناقص یا نویزی به مدل داده می‌شود. هدف این نیست که مدل همیشه خروجی دقیق بدهد، بلکه این است که در صورت عدم اطمینان، رفتار محافظه‌کارانه‌تری اتخاذ کند. برای مثال، در شرایطی که ورودی‌ها از نظر آماری با داده‌های آموزش تفاوت چشمگیری دارند، مدل به‌جای ارائه خروجی قاطع، سطح اطمینان پایین‌تری گزارش می‌دهد یا تصمیم را به لایه‌های کنترلی بالاتر واگذار می‌کند.

این رویکرد باعث می‌شود مدل به‌جای اینکه در شرایط بحرانی «اشتباه ولی مطمئن» باشد، «مردد اما ایمن» عمل کند؛ که برای یک سیستم شهری ارزشمندتر است.

۱۵ .تحلیل بایاس (Bias Analysis) و پیامدهای نابرابری تصمیم‌ها

یکی از حساس‌ترین ابعاد استفاده از مدل‌های یادگیری ماشین، مسئله بایاس است. درMetroGo ، بایاس صرفاً یک مفهوم اخلاقی انتزاعی تلقی نمی‌شود، بلکه یک ریسک عملیاتی و اعتباری جدی است. اگر مدل به‌صورت سیستماتیک برای گروه خاصی از کاربران خروجی متفاوتی ارائه دهد، حتی اگر این تفاوت ناخواسته باشد، می‌تواند منجر به نارضایتی، بی‌اعتمادی و حتی واکنش‌های حقوقی شود.

تحلیل بایاس در MetroGo بر پایه این اصل انجام شده که «داده‌های تاریخی الزاماً منصفانه نیستند». رفتار گذشته کاربران، سیاست‌های شهری، یا محدودیت‌های زیرساختی می‌توانند الگوهایی ایجاد کرده باشند که اگر بدون اصلاح وارد مدل شوند، به بازتولید نابرابری منجر شوند.

به همین دلیل، خروجی مدل‌ها به تفکیک بخش‌های مختلف کاربران تحلیل شده است. هدف از این تحلیل یافتن تفاوت‌های غیرقابل توجیه در عملکرد مدل است، نه حذف هر نوع تفاوت. تفاوت زمانی مشکل‌ساز تلقی می‌شود که ناشی از سوگیری داده باشد، نه تفاوت واقعی در رفتار یا نیاز کاربران.

این تحلیل‌ها همچنین به تیم کمک کرده‌اند تصمیم بگیرد در چه مواردی نباید از مدل برای تصمیم‌گیری مستقیم استفاده شود و نقش مدل صرفاً در حد پشتیبان باقی بماند.

۱۶ .ارزیابی سناریوهای واقعی(Real-world Scenario Testing)

یکی از نقاط تمایز این گزارش، تمرکز آن بر سناریوهای واقعی به‌جای سناریوهای مصنوعی آزمایشگاهی است. مدل MetroGo نه برای رقابت در بنچمارک‌ها، بلکه برای کار در محیط واقعی طراحی شده است. به همین دلیل، ارزیابی‌ها بر اساس سناریوهایی انجام شده‌اند که مستقیماً از تجربه‌های پایلوت، تست‌های میدانی یا شبیه‌سازی‌های نزدیک به واقعیت استخراج شده‌اند.

در این سناریوها، مدل با ترکیبی از چالش‌ها مواجه می‌شود: داده‌های ناقص، تأخیر زمانی، رفتار غیرقابل پیش‌بینی کاربران و محدودیت‌های زیرساختی. بررسی عملکرد مدل در چنین شرایطی تصویر دقیق‌تری از قابلیت واقعی آن ارائه می‌دهد تا صرفاً گزارش دقت روی داده‌های تمیز.

نتیجه این ارزیابی‌ها نشان داده که مدل در برخی شرایط عملکرد پایداری دارد و در برخی شرایط نیازمند محدودسازی یا بازطراحی جریان تصمیم‌گیری است. این شفافیت در بیان نقاط ضعف، بخشی از تعهد MetroGo به استفاده مسئولانه از AI است.

۱۷ .مدیریت ریسک استفاده نادرست(Misuse & Over-reliance Risk)

یکی از ریسک‌های کمتر دیده‌شده در پروژه‌هایAI ، نه خطای مدل، بلکه استفاده نادرست از آن است. اگر کاربران داخلی یا حتی مدیران پروژه به خروجی مدل بیش از حد اعتماد کنند، مدل عملاً به یک «مرجع حقیقت» تبدیل می‌شود؛ نقشی که هیچ مدل آماری نباید داشته باشد.

درMetroGo ، این ریسک به‌طور جدی شناسایی شده است. به همین دلیل، مستندات مدل به‌گونه‌ای نوشته شده‌اند که محدودیت‌ها و شرایط استفاده به‌وضوح بیان شوند. همچنین در طراحی سیستم، خروجی مدل به‌صورت خام و بدون زمینه در اختیار تصمیم‌گیران قرار نمی‌گیرد، بلکه همراه با توضیح، سطح اطمینان و هشدارهای لازم ارائه می‌شود.

این کار باعث می‌شود مدل به‌عنوان یک ابزار تصمیم‌یار دیده شود، نه تصمیم‌گیر نهایی.

۱۸ .بازآموزی مدل و مدیریت Drift در بلندمدت

مدل MetroGo در محیطی فعالیت می‌کند که ذاتاً پویاست. رفتار کاربران، سیاست‌های شهری و حتی فناوری‌های پرداخت در طول زمان تغییر می‌کنند. اگر مدل بدون بازبینی و بازآموزی رها شود، به‌تدریج از واقعیت فاصله می‌گیرد و تصمیم‌های آن نامعتبر می‌شود.

به همین دلیل، برنامه مشخصی برای پایش drift و تصمیم‌گیری درباره بازآموزی مدل تعریف شده است. این تصمیم نه بر اساس زمان ثابت، بلکه بر اساس شواهد داده‌ای اتخاذ می‌شود. زمانی که الگوهای ورودی یا خروجی به‌طور معناداری تغییر کنند، تیم وارد فاز بازبینی می‌شود.

این رویکرد باعث می‌شود مدل همواره با واقعیت‌های جاری هم‌راستا باقی بماند و به بدهی فنی یا تصمیمی تبدیل نشود.

۱۹ .ملاحظات اخلاقی و مسئولیت‌پذیری تیم

در نهایت، این گزارش تأکید می‌کند که مسئولیت استفاده از مدل صرفاً بر عهده الگوریتم نیست، بلکه بر عهده تیمی است که آن را طراحی، آموزش و استفاده می‌کند MetroGo .تلاش کرده است از همان ابتدا این مسئولیت‌پذیری را در فرهنگ فنی خود نهادینه کند.

این مسئولیت‌پذیری در تصمیم برای عدم استفاده از AI در برخی بخش‌ها نیز دیده می‌شود؛ جاهایی که هزینه خطا بیش از منفعت دقت بالاتر بوده است.

۲۰ .جمع‌بندی نهایی: چرا این Model Card قابل دفاع است؟

این Model Card و گزارش ارزیابی نه برای نمایش پیچیدگی فنی، بلکه برای ایجاد شفافیت، اعتماد و قابلیت دفاع طراحی شده است. هدف آن پاسخ به این سؤال کلیدی است:

»اگر فردی بیرون از تیم این مدل را بررسی کند، آیا می‌تواند بفهمد چه کار می‌کند، کجا شکست می‌خورد و چرا هنوز قابل استفاده است؟«

MetroGoبا این سند نشان می‌دهد که AI برای آن ابزار رشد است، نه ریسک پنهان.