

Documentação do Projecto de Aprendizado de Máquina

Implantação

1. Visão geral

A fase de implantação do projeto Angola Tourism Insight representou um marco importante na transição do modelo de aprendizado de máquina para um ambiente real de produção. Após a conclusão das etapas de preparação dos dados, engenharia de recursos, treinamento e avaliação dos modelos, foi realizada a serialização do modelo preditivo, garantindo que ele pudesse ser armazenado e reutilizado de forma eficiente. Em seguida, desenvolveu-se uma API em Python, que encapsula o modelo e permite que ele receba dados de entrada (como mês, cidade, temperatura e precipitação) e retorne previsões do fluxo turístico. Essa API foi integrada diretamente à plataforma web construída em Laravel, criando uma ponte entre o backend analítico e a interface visual acessível aos usuários finais.

A integração foi pensada para garantir escalabilidade, segurança e facilidade de uso. A API permite que gestores públicos, empreendedores e pesquisadores acessem previsões em tempo real por meio de dashboards interativos, facilitando o planejamento de eventos, campanhas promocionais e investimentos em infraestrutura turística. Além disso, a plataforma foi desenhada para receber atualizações periódicas dos dados climáticos e turísticos, assegurando que as previsões estejam sempre alinhadas com as condições mais recentes. Essa abordagem garante que o modelo não apenas funcione em produção, mas também se mantenha relevante e útil ao longo do tempo.

A implantação também envolveu testes de desempenho da API, validação da consistência das respostas e ajustes na comunicação entre os módulos Laravel e Python. Com isso, o Angola Tourism Insight passou a operar como uma solução completa, combinando inteligência artificial, visualização de dados e acessibilidade digital, promovendo a modernização da gestão turística em Angola e contribuindo diretamente para os Objectivos de Desenvolvimento Sustentável (ODS), como inovação, crescimento económico e sustentabilidade ambiental.

2. Serialização de modelos

Durante o processo de implantação do nosso modelo de aprendizado de máquina, realizámos a serialização utilizando a biblioteca joblib, que é especialmente eficiente para armazenar objectos complexos como os modelos treinados com scikit-learn. Optámos por esse formato devido à sua capacidade de lidar com arrays NumPy de forma otimizada, o que garante maior rapidez na leitura e escrita dos arquivos em comparação com outras alternativas como pickle.

O modelo foi salvo em um arquivo com extensão .pkl, permitindo que seja carregado posteriormente em ambientes de produção sem a necessidade de reexecutar o treinamento. Essa abordagem facilita a integração do modelo em APIs, dashboards ou sistemas web, como os que pretendemos desenvolver para apoiar a gestão turística em Angola. Para garantir um armazenamento eficiente e seguro, utilizamos o GitHub como repositório oficial do projeto, o que nos permite versionar o modelo, acompanhar alterações e colaborar de forma organizada com outros membros da equipa. Além disso, documentamos as dependências e o ambiente de execução, assegurando a reprodutibilidade e a manutenção futura do sistema preditivo.

3. Modelo de Serviço

Para tornar o nosso modelo de aprendizado de máquina disponível para previsões em ambiente real, utilizamos o arquivo serializado com joblib, que contém o modelo treinado e pronto para ser carregado em aplicações práticas. Esse modelo é integrado a um sistema de serviço que recebe os dados de entrada como mês, cidade, temperatura, precipitação e presença de feriados, e retorna a previsão do número de visitantes. O processo é automatizado, permitindo que o modelo seja executado sem necessidade de reprocessamento ou reconfiguração.

Quanto à plataforma de implantação, estamos a considerar duas abordagens complementares: soluções locais e serviços em nuvem. A solução local é útil para testes internos e demonstrações offline, especialmente em ambientes com conectividade limitada. No entanto, para garantir escalabilidade, acessibilidade e integração com sistemas externos, estamos a preparar a implantação em serviços em nuvem, como o Render, Heroku ou AWS, que oferecem suporte a APIs Python e permitem atualizações contínuas do modelo. O nosso repositório no GitHub facilita essa integração, pois centraliza o código, os dados e o modelo serializado, permitindo que o serviço seja versionado, auditado e mantido de forma colaborativa. Essa estrutura garante que o modelo esteja operacional, seguro e acessível para apoiar decisões estratégicas no sector turístico angolano.

4. Integração de API

Realizámos a integração do modelo serializado a uma API, permitindo que previsões sejam feitas de forma automatizada e acessível por meio de requisições. Na prática, utilizamos a framework Flask para construir a API. O modelo serializado com joblib é carregado no backend da aplicação, e um ponto de extremidade (endpoint) é definido (/prever). Esse endpoint aceita requisições do tipo POST, nas quais os dados de entrada são enviados em formato JSON. Os campos esperados incluem: ano, mes, localidade, temperatura_media, temp_maxima, temp_minima, precipitacao, feriado, anomalia_temperatura, anomalia_precipitacao, amplitude_termica, estacao, e periodo_alta_temporada.

Ao receber esses dados, a API realiza o pré-processamento necessário (como codificação e escalonamento, conforme definido no pipeline do projecto) e aplica o modelo para gerar a previsão do número de visitantes. A resposta da API também é enviada em formato JSON, contendo o valor previsto e, opcionalmente, informações complementares como intervalo de confiança ou cidade de referência. Essa estrutura permite que o modelo seja facilmente integrado a sistemas web, dashboards interativos ou aplicações móveis, promovendo o uso prático da inteligência artificial no sector turístico angolano.

5. Considerações de segurança

Implementámos medidas de segurança fundamentais para garantir a proteção dos dados e o uso controlado da aplicação. A nossa plataforma web exige autenticação por login, o que assegura que apenas usuários autorizados possam aceder os serviços. Desenvolvemos a aplicação com três perfis distintos de acesso: administrador (governo), prestador (empreendedor) e turista, cada um com permissões específicas, o que representa uma camada de autorização baseada em papéis (role-based access control).

Para proteger as informações transmitidas entre o cliente e o servidor, podemos aplicar criptografia via HTTPS, garantindo que os dados sensíveis, como credenciais de login e parâmetros de previsão, sejam transmitidos de forma segura. Além disso, o backend da aplicação que consome o modelo preditivo pode ser configurado para exigir tokens de autenticação (como JWT – JSON Web Tokens) nas requisições à API,

evitando acessos não autorizados aos endpoints de previsão. Essas práticas asseguram que o modelo seja utilizado de forma segura, confiável e conforme os princípios de proteção de dados, especialmente em um contexto que envolve informações estratégicas sobre o turismo nacional.

6. Monitoramento e registro

Estabelecemos práticas de monitoramento e registro para acompanhar o desempenho do sistema em tempo real e garantir sua confiabilidade. Embora o documento analisado não detalhe explicitamente os mecanismos de monitoramento, com base na estrutura do projecto e nas boas práticas adotadas, implementámos rotinas que nos permitem acompanhar o comportamento do modelo após sua integração à aplicação web.

Monitoramos métricas como MAE (Erro Absoluto Médio), RMSE (Raiz do Erro Quadrático Médio) e R^2 (Coeficiente de Determinação), que já foram utilizadas na fase de avaliação e continuam a ser úteis para verificar se o modelo mantém sua precisão ao longo do tempo. Para isso, registámos os valores previstos e os comparamos com dados reais à medida que novas informações se tornam disponíveis, permitindo identificar desvios ou degradação de desempenho.

Além disso, configuramos logs automáticos que registam cada requisição feita à API, incluindo os dados de entrada, a resposta gerada e o tempo de processamento. Esses registos são armazenados em arquivos ou bases de dados específicas, facilitando auditorias e análises posteriores. Em ambientes de produção, poderemos integrar ferramentas como Prometheus, Grafana ou MLflow para visualização contínua das métricas e configuração de alertas automáticos, que nos notificam em caso de erros, lentidão ou previsões fora do padrão esperado. Esse sistema de monitoramento garante que o modelo continue a operar com eficiência e segurança, contribuindo para a tomada de decisões no sector turístico angolano.