Previsão de Comportamento de Consumo com Deep Learning: Análise Comparativa e Impacto da Engenharia de Features

1. Visão Geral do Projeto

Este projeto foi concebido para desenvolver um modelo preditivo de altíssima performance para o comportamento de consumo diário, utilizando o dataset "Online Retail". A investigação evoluiu de modelos clássicos para uma arquitetura de Deep Learning sofisticada, culminando em uma solução de notável precisão. A análise comparativa entre diferentes algoritmos, em cenários com e sem engenharia de features, demonstrou que a combinação de uma **Rede Neural Recorrente Bidirecional (Bi-LSTM) com um Mecanismo de Atenção** e um conjunto de **features temporais enriquecidas** é a chave para a modelagem precisa da série.

O modelo final alcançou uma performance excepcional no conjunto de teste, com um erro médio (RMSE) de apenas **£3,635.27** e um coeficiente de determinação (R^2) de **0.964**, explicando 96.4% da variabilidade dos dados. Este resultado valida a hipótese de que arquiteturas avançadas, munidas com o contexto de negócio correto (feriados, sazonalidade), podem decodificar padrões complexos de consumo com grande acurácia.

2. Arsenal Tecnológico

A execução deste projeto exigiu um conjunto de ferramentas específicas, cada uma cumprindo um papel crítico no pipeline de desenvolvimento:

- **pandas**: Fundamental para a ingestão, higienização e transformação dos dados brutos em um formato estruturado e analisável.
- **numpy**: Utilizado para a manipulação de arrays multidimensionais, a estrutura de dados primária para as operações do TensorFlow.
- **scikit-learn**: Empregado para o pré-processamento (MinMaxScaler) e para a avaliação rigorosa do modelo através de métricas padrão (mean_squared_error, mean_absolute_error, r2_score).
- **tensorflow e keras**: O core do projeto, utilizado para a prototipagem, construção e treinamento da arquitetura de rede neural.
- **matplotlib**: Essencial para a visualização dos resultados, permitindo a análise qualitativa da aderência das previsões aos dados reais.
- **holidays**: Ferramenta utilizada na fase de engenharia de features para criar um indicador binário de feriados, agregando contexto de negócio ao modelo.
- **warnings**: Utilizado para suprimir avisos não-críticos, focando a saída nos resultados relevantes.

3. Pré-processamento e Engenharia de Features

A qualidade da previsão é diretamente dependente da qualidade dos dados. A etapa de engenharia de features provou ser o fator mais impactante na performance dos modelos.

- **Dataset**: Online Retail Dataset (UCI).
- **Higienização dos Dados**:
  - Remoção de transações de devolução (faturas com prefixo 'C').
  - Exclusão de registros com CustomerID nulo para garantir a rastreabilidade.
  - Filtragem de outliers em Quantity e UnitPrice.
- **Transformação e Agregação**: O dado bruto, a nível de transação, foi agregado para formar uma série temporal de vendas diárias (TotalPrice).
- **Engenharia de Features (Cenário Enriquecido)**: Para munir o modelo com contexto temporal, foram projetadas as seguintes features, que se mostraram cruciais:
  - day_of_week: Sinaliza o padrão semanal de vendas.
  - week_of_year: Captura a sazonalidade anual.
  - is_holiday: Indica a ocorrência de feriados no Reino Unido.

4. Análise Comparativa de Modelos

Para mensurar o impacto da arquitetura e da engenharia de features, os modelos foram avaliados em dois cenários distintos.

Cenário 1: Desempenho com Features Mínimas (Baseline)

Neste cenário, os modelos foram treinados utilizando apenas os dados brutos agregados por dia (TotalPrice), sem contexto temporal adicional.

Modelo	RMSE	MAE	R^2	Avaliação Qualitativa
Árvore de Decisão	20,392.10	15,608.36	-0.118	Ruim (pior que a média)
Random Forest	20,568.71	16,349.11	-0.138	Ruim (pior que a média)
XGBoost	19,316.41	15,203.28	-0.003	Ruim (pior que a média)
Prophet	19,144.06	15,155.03	0.014	Razoável

LSTM Simples	16,415.53	13,790.31	0.275	Razoável
Bi-LSTM + Attention	**12,156.75**	**10,004.84**	**0.603**	**Muito Boa**

Diagnóstico do Cenário 1: Mesmo sem features contextuais, a arquitetura Bi-LSTM + Attention demonstrou uma capacidade inata de capturar dependências sequenciais, superando todos os outros modelos com folga. Os modelos tabulares clássicos (Árvore, Random Forest, XGBoost) falharam completamente, com R^2 negativo, indicando que suas previsões foram piores do que simplesmente usar a média histórica das vendas.

Cenário 2: Desempenho com Features Enriquecidas

Neste cenário, as features de calendário (day_of_week, week_of_year, is_holiday) foram adicionadas ao conjunto de dados.

Modelo	**RMSE**	**MAE**	**R^2**	**Avaliação Qualitativa**
Árvore de Decisão	17,758.25	13,332.91	0.152	Razoável
Random Forest	11,774.30	8,622.40	0.627	Muito Boa
XGBoost	11,677.68	9,114.66	0.633	Muito Boa
Prophet	22,972.32	18,323.13	-0.419	Ruim (pior que a média)
LSTM Simples	16,415.53	13,790.31	0.275	Razoável
Bi-LSTM + Attention	**3,635.27**	**2,683.01**	**0.964**	**Muito Boa**

Diagnóstico do Cenário 2: A introdução de features contextuais causou uma transformação radical nos resultados. O modelo Bi-LSTM + Attention teve um salto quântico em performance, atingindo um R^2 de 0.964. Notavelmente, os modelos Random Forest e XGBoost, que antes eram ineficazes, tornaram-se competitivos, explicando mais de 62% da variância. Isso evidencia que, embora a arquitetura seja importante, a engenharia de features foi o verdadeiro catalisador do sucesso. A anomalia do Prophet, que piorou seu desempenho, sugere uma incapacidade do

algoritmo de integrar corretamente as features externas fornecidas neste caso.

5. Arquitetura Final: Especificação Técnica (Bi-LSTM + Attention)

O modelo de melhor performance foi uma rede Bi-LSTM com um mecanismo de atenção.

- **Preparação dos Dados**:
  - **Normalização**: Todas as features foram normalizadas para um intervalo [0, 1] com MinMaxScaler, passo crucial para a estabilidade do treinamento.
  - **Criação de Sequências**: Uma função deslizou uma "janela" de 60 dias sobre a série temporal. Para cada janela, os 60 dias de dados multivariados (TotalPrice, day_of_week, week_of_year, is_holiday) se tornaram a entrada (X), e o valor de vendas do 61º dia se tornou o alvo (y).
- **Arquitetura da Rede Neural**:
  1. **Input**: Camada de entrada com formato (None, 60, 4), representando lotes de sequências de 60 dias com 4 features cada.
  2. **Encoder Bi-LSTM**: Uma camada LSTM Bidirecional processa a sequência em ambas as direções (passado-futuro e futuro-passado), criando uma representação contextual rica dos dados.
  3. **Mecanismo de Atenção**: Uma camada de Atenção recebe as saídas do encoder e calcula pesos de importância para cada um dos 60 dias passados. Ela gera um "vetor de contexto", que é uma soma ponderada das saídas do encoder, permitindo que o modelo foque dinamicamente nos dias mais influentes para a previsão.
  4. **Decodificador (MLP)**: O vetor de contexto é concatenado com os estados finais do encoder e alimentado em uma rede densa com ativação ReLU e Dropout, que mapeia a representação complexa para a previsão final.
  5. **Output**: Um único neurônio com ativação linear que produz o valor de venda previsto (normalizado).
- **Treinamento e Avaliação**:
  - **Compilação**: Otimizador Adam e função de perda mean_squared_error.
  - **Early Stopping**: O treinamento foi monitorado para evitar overfitting, parando caso a perda na validação não melhorasse por 20 épocas consecutivas.
  - **Reversão da Escala**: As previsões foram convertidas de volta à sua escala monetária original (£) para interpretação e cálculo das métricas finais.

6. Conclusão e Aplicabilidades

Este projeto demonstra conclusivamente que a previsão de séries temporais de varejo com alta volatilidade atinge um nível de excelência através da **sinergia entre uma arquitetura de Deep Learning avançada e uma engenharia de features**

criteriosa. A jornada iterativa, validada por uma análise comparativa rigorosa, foi crucial para elevar o desempenho de um R^2 negativo em modelos básicos para um **R^2 de 0.964**, transformando dados brutos em inteligência acionável com um erro de previsão de apenas **£3,635**.

Aplicabilidades de Negócio:

- Otimização de inventário e planejamento logístico com alta precisão.
- Direcionamento tático de campanhas de marketing em períodos de alta ou baixa demanda prevista.
- Alocação estratégica de recursos humanos e financeiros em datas sazonais e feriados.

O trabalho foi conduzido com rigor técnico, provando como a ciência de dados moderna pode gerar valor tangível e uma vantagem competitiva significativa para a tomada de decisões.