

Trabalho Prático 3
Previsão de tráfego de rede com Redes Neurais Recorrentes
MC833 – Programação em Redes de Computadores – UNICAMP

Introdução

A previsão de volume de tráfego em redes de computadores é fundamental para planejamento de capacidade, detecção de anomalias e balanceamento de carga. Este projeto propõe que você trabalhe com um arquivo público (PCAP) da base MAWI/CAIDA, construa uma série temporal segundo a segundo com base nos dados de tráfego e utilize uma abordagem de previsão baseada em redes neurais.

LSTM (Redes Neurais Recorrentes)

Modelos do tipo LSTM (Long Short-Term Memory) pertencem à classe de redes neurais recorrentes (RNNs) e são especialmente eficazes para tratar problemas com dados sequenciais, como séries temporais de tráfego de rede. As LSTMs são capazes de aprender padrões temporais complexos, mantendo uma "memória" interna que considera tanto tendências de curto quanto de longo prazo.

Neste trabalho, a LSTM será usada para prever o volume de tráfego por segundo, utilizando como entrada uma janela deslizante de valores anteriores (por exemplo, os últimos 10, 20 ou 30 segundos). A rede deverá prever o próximo valor da série temporal construída a partir do arquivo PCAP.

Como utilizar a LSTM:

A implementação deverá ser feita com TensorFlow/Keras (ou alternativamente PyTorch).

A arquitetura recomendada é enxuta: uma camada LSTM seguida de uma camada densa (fully connected).

A entrada será uma sequência de bytes_per_second, gerada a partir da série temporal.

A saída será o valor previsto para o segundo seguinte.

Ajustes de hiperparâmetros como epochs, batch_size e look_back devem ser considerados e registrados.

Além de treinar e avaliar o modelo, você deverá realizar uma análise exploratória dos dados (EDA), com foco em identificar padrões no tráfego, como picos, comportamentos periódicos e a distribuição de protocolos (por exemplo, TCP e UDP).

Detalhes da tarefa

1. Aquisição e preparação dos dados

- O que fazer:

- Download do arquivo PCAP:

Um arquivo pré-processado já está disponível para download no seguinte link:

👉 [\[link\]](#)

Ele contém a série temporal de tráfego em formato compacto, facilitando a análise e treinamento do modelo.

- Opção alternativa:

Os alunos que desejarem podem baixar diretamente um arquivo PCAP bruto da base pública MAWI (<https://mawi.wide.ad.jp/mawi/>) especificamente o seguinte [\[link\]](#), e processá-lo por conta própria. No entanto, atenção: o processamento de arquivos PCAP grandes pode ser demorado e exigir mais recursos computacionais.

- Códigos de apoio:
 - Na pasta compartilhada do projeto, estão disponíveis dois exemplos de código em Python:
 - Um script para ler arquivos em formato Parquet contendo os dados pré-processados. Um script alternativo que lê e processa diretamente arquivos PCAP, gerando a série temporal `bytes_per_second`.
 △ Este segundo script pode levar mais tempo para rodar, especialmente com arquivos grandes.
2. Análise exploratória de dados (EDA)
- O que fazer:
 - Gerar estatísticas descritivas: média, desvio-padrão, quartis, valores máximos e mínimos.
 - Produzir visualizações: histogramas de distribuição de bytes para diferentes protocolos (TCP, UDP)
 - Resultado esperado
 Uma seção “Análise exploratória de dados” no relatório, contendo todas as figuras e comentários interpretativos.
3. Modelamento preditivo: LSTM
- O que fazer
 - Preparação da série – Gerar os conjuntos de treino (80%) e teste (20%) a partir da série `bytes_per_second`, extraída do arquivo PCAP. Criar janelas deslizantes (look-back) de 10 segundos como entrada para o modelo LSTM. Em seguida, testar outras janelas (ex: 20 s e 30 s) para avaliar a sensibilidade da rede à quantidade de histórico.
 - Rede neural (LSTM) – Implementar uma arquitetura enxuta com uma camada LSTM seguida de uma camada densa (fully connected) responsável por prever o próximo ponto da série. Ajustar hiperparâmetros essenciais como número de épocas (epochs), tamanho do batch (batch size) e tamanho da janela (look-back). Salvar e registrar o melhor modelo.
 - Treinamento e avaliação – Treinar o modelo nos dados de treino e avaliá-lo no conjunto de teste, utilizando a métrica [MSE](#) (Erro Quadrático Médio) para mensurar o desempenho preditivo.
 - Visualização e análise:
 - Uma tabela com os valores de MSE para diferentes configurações de janela;
 - Um gráfico “valor real × previsto” mostrando a curva prevista pela LSTM em comparação com os dados reais no conjunto de teste;
 - Comentários breves destacando os trechos em que o modelo apresenta boa aderência ou desvios importantes (ex: picos, períodos de estabilidade ou variações rápidas).
 - Resultado esperado
 - Seção do relatório intitulada “Modelamento preditivo” contendo:
 - Descrição da abordagem LSTM adotada;
 - Tabela de MSE por configuração de janela;
 - Gráfico comparativo entre valores reais e previstos;
 - Análise preliminar sobre o desempenho do modelo LSTM.
4. Discussão crítica

- O que fazer
 - Explicar por que o modelo teve o desempenho observado: influência de burst TCP, padrões horários, estacionariedade da série, etc.
 - Relacionar os resultados a fenômenos típicos de redes (picos de tráfego, comportamentos periódicos, possíveis ataques).
 - Resultado esperado
 - Conclusões claras no relatório, apontando vantagens, limitações e sugestões de trabalho futuro

Entregáveis

Envio único: relatório em PDF, via Google Classroom.

Boas práticas essenciais

Fixar sementes (numpy, random, tensorflow ou pytorch) para reprodutibilidade.

Comentar o código.

Não versionar o PCAP; use .gitignore.

Discutir implicações éticas (privacidade dos fluxos).