Olá mundo.

Infraestrutura Computacional - Módulo II

Paulo Ricardo Lisboa de Almeida

DSBD

UFPR

Professor

Paulo Ricardo Lisboa de Almeida

Doutor em Ciência da Computação, Bel. em Eng. da Computação.

paulorla@ufpr.br

prlalmeida.com.br

www.linkedin.com/in/paulorla

Professor - Pesquisa

Aprendizado de máquina.

Machine Learning para fluxos de dados.

Cidades inteligentes.

DSBD

Efficient Prequential AUC-PR Computation

Artigo - IEEE ICMLA 2023.

Florida, Estados Unidos.

David

Paulo

Grégio

Zanata

Efficient Prequential AUC-PR Computation

David L. Pereira Gomes, André Grégio, Marco A. Zanata Alves, Paulo R. Lisboa de Almeida
Department of Informatics – Federal University of Parana (UFPR) – Curitiba, PR - Brazil
david.gomes@ufpr.br, gregio@ufpr.br, mazalves@ufpr.br, paulorla@ufpr.br

***Abstract*—When dealing with classification problems for data streams, we often need to compute the classification metrics in a prequential manner. The Area Under the Precision-Recall Curve (AUC-PR) metric is extensively used in imbalanced classification scenarios, where the negative class outnumbers the positive one. Despite its advantages, it may be computationally expensive to recompute that metric every time a new test instance becomes available. In this work, we present an efficient algorithm to compute the AUC-PR in a prequential way. Our proposed algorithm uses a self-balancing binary search tree to avoid the need to reorder the data when updating the AUC-PR value with the most recent data. Our experiments take into consideration six well-known, publicly available stream-based datasets. Our experiments show that our approach can be up to 13 times faster and use 12 times less energy than the traditional batch approach when considering a window of size 1,000.**

***Index Terms*—AUC-PR, prequential, stream, metrics**

I. Introduction

The massive amount of data produced by sensors, devices, and users poses a challenge to the application of classification algorithms whose output needs to be provided in real-time (e.g., critical systems, emergency diagnosis, security, threat detection, etc.). Those data are usually temporally-dependent, arriving at the classifier as a data stream.

Metrics for classification problems involving data streams, such as accuracy, F1-score, and Area Under the Precision-Recall Curve (AUC-PR), are often computed in a prequential manner, i.e., every time a new test instance becomes available. The reasoning behind the prequential calculation of those metrics is to allow for the monitoring of the classifier's performance over time, as well as quickly reacting to environmental changes (e.g., concept drifts), which may hinder the classification capability of a decision-support system.

Therefore, the prequential computation of classification metrics often requires computing them using a window W that contains the latest labeled data received. Thus, a metric must be recomputed on each update of this window, which may lead to an overhead that turns the classification of data streams into an overly expensive task, especially if we rely on computationally intensive metrics, such as the AUC-PR. The incurred overhead may increase costs (e.g., more computing power in servers) and the carbon footprint associated with this type of classification system.

In this work, we introduce an efficient algorithm to compute the AUC-PR for streams in a prequential manner (assuming a stream of instances, in which samples arrive for classification one at a time). The AUC-PR metric belongs to a family of metrics focused on imbalanced scenarios. To the best of our knowledge, this is the first algorithm to reduce the time complexity from $O(m \log m)$ to $O(m)$ when computing the AUC-PR metric for streams. In our experiments, the proposed algorithm was 13 times faster and used 12 times less energy when compared to the batch approach (i.e., recomputing the metric from scratch every time the window W is updated), often used when a prequential algorithm is unavailable. The main contributions of this paper are:

- An algorithm to calculate the AUC-PR for stream settings in a prequential way, focusing on its efficiency;
- The evaluation of our proposed algorithm and comparison with a widely used implementation of the metric that considers batch settings.

II. Background and Related Work

In this Section, we introduce concepts needed for properly understanding our proposed method, such as the prequential computation of metrics and the definition of the Precision-Recall Curve. We also present the related state-of-the-art work.

A. Batch versus Prequential Metrics

Data classification can be divided into two settings: batch (or static) learning or stream. In the former, we consider that the available data is limited to a "snapshot" of a certain period of time, whereas in the latter, we have to handle unlimited data continuously arriving at potentially high rates [1].

Under a static setting, we may create a classifier using a train set S_c, and test its performance using some metric in a test set S_t, where $S_c \cap S_t = \emptyset$. This approach is known as holdout or batch testing [2]. On the other hand, under a stream scenario, new instances arrive over time, making it impossible to have a fixed test set to assess the classifier's performance—especially under conditions where the problem may evolve.

In a stream scenario, it is common to define a window W containing the m latest instances received and compute the performance metrics using this window. Every time a new test instance arrives, this window is moved to accommodate the new instance, and the metrics are updated. This approach is known as the prequential computation of the metric [2].

For example, let's consider the stream at times t and $t+1$ in Figure 1, in which the metrics are computed within a window that contains the $m = 5$ latest instances. When a test instance $\mathbf{x}_{t+1}$ arrives at time $t+1$, the window is moved to accommodate this new instance, and the oldest instance in the window ($\mathbf{x}_{t-4}$) is removed from the window.

When the window moves, it is updated, and we may recompute the performance metrics using the entire window (i.e., the whole window is considered a batch). Besides being simple,

ICMLA 2023.

Pesquisa - Concept Drifts e streams

Como criar modelos que se adaptam às mudanças de ambiente?

Como acompanhar as mudanças?

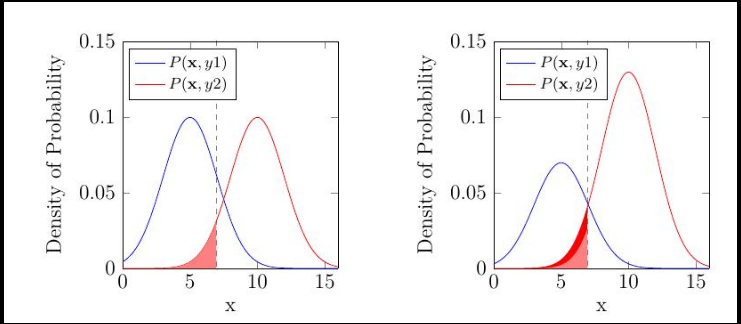
Quando a informação mudou?

Qual informação continua sendo útil?

t

t+1

Pesquisa
Dynse Framework




Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

ELSEVIER

Adapting dynamic classifier selection for concept drift

Paulo R.L. Almeida[a,*], Luiz S. Oliveira[a], Alceu S. Britto Jr.[b,c], Robert Sabourin[d]

[a] Federal University of Parana (UFPR), Rua Cel. Francisco H. dos Santos, Curitiba, PR 100-81531-990, Brazil
[b] Ponta Grossa State University, Av. General Carlos Cavalcanti, Ponta Grossa, PR, 4748-84030-900, Brazil
[c] Pontifical Catholic University of Parana (PUCPR), R. Imaculada Conceição, Curitiba, PR 1155-80215-901, Brazil
[d] École de Technologie Supérieure, 1100 Notre-Dame Street West, Montreal, Quebec, Canada

ARTICLE INF

Article history:
Received 20 December 2017
Revised 12 March 2018
Accepted 13 March 2018
Available online 15 March 201

Keywords:
Concept drift
Dynamic classifier selection
Dynamic ensemble selection
Concept diversity

1. Introduction

Naïve Approaches to Deal With Concept Drifts

Paulo R. Lisboa de Almeida, Department of Computer Science, Univ. do Estado de Santa Catarina, Joinville (SC), Brazil, paulo.almeida@udesc.br
Luiz S. Oliveira, Department of Informatics, Univ. Federal do Paraná, Curitiba (PR), Brazil, luiz.oliveira@ufpr.br
Alceu de Souza Britto Jr., Jean Paul Barddal, Graduate Program in Informatics (PPGIa), Pontifícia Universidade Católica do Paraná, Curitiba (PR), Brazil, {alceu, jean.barddal}@ppgia.pucpr.br

Abstract—A common problem in machine learning is to find representative real-world labeled datasets to put the methods to test. When developing approaches to deal with concept drifts, some datasets such as the Forest Covertype and Nebraska Weather are common choices for testing, even though there is

concept drifts, while the second are unable to handle concept drifts. Naïve methods should give us insight about the changes since these may not adapt to drifts, and thus, should not perform well, for instance, with relevant accuracy drops when

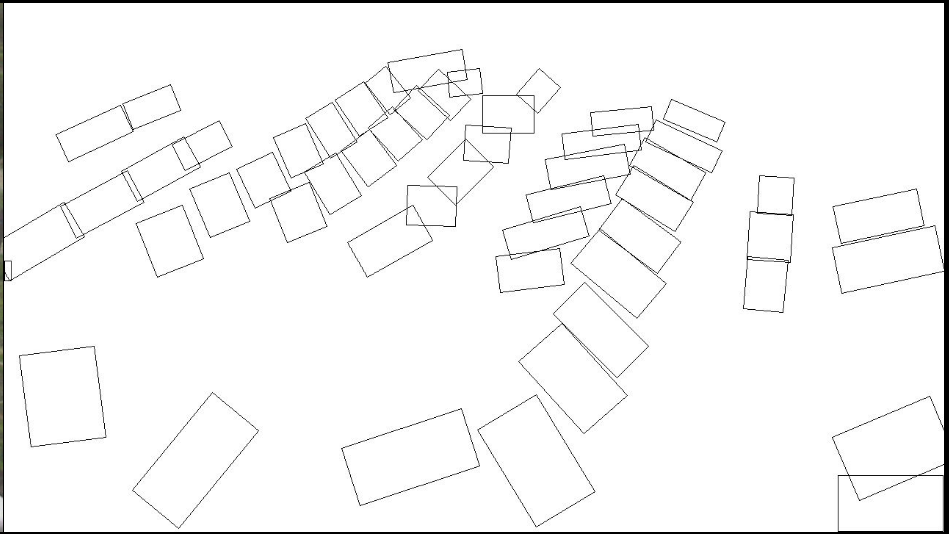
Handling Concept Drifts Using Dynamic Selection of Classifiers

Paulo R. Lisboa de Almeida*, Luiz S. Oliveira*, Alceu de Souza Britto Jr.‡ and § and Robert Sabourin¶
*Universidade Federal do Paraná, DInf, Curitiba, PR, Brazil
Email: {prlalmeida,lesoliveira}@inf.ufpr.br
‡Universidade Estadual de Ponta Grossa, Deinfo, Ponta Grossa, PR, Brazil
§Pontifícia Universidade Católica do Paraná, PPGIa, Curitiba, PR, Brazil
Email: alceu@ppgia.pucpr.br
¶École de Technologie Supérieure, Montreal, QC, Canada
Email: robert.sabourin@etsmtl.ca

Distance Functions and Normalization Under Stream Scenarios

Eduardo V. L. Barboza*, Paulo R. Lisboa de Almeida*, Alceu de Souza Britto Jr.†‡ and Rafael M. O. C
*Department of Informatics, Universidade Federal do Paraná, Curitiba (PR), Brazil
Email: {eduardo.barboza, paulorla}@ufpr.br
†Graduate Program in Informatics, Pontífica Universidade Católica do Paraná, Curitiba (PR), Brazil
‡Universidade Estadual de Ponta Grossa, Ponta Grossa (PR), Brazil
Email: alceu@ppgia.puc.br
§École de Technologie Supérieure, Université du Québec, Montréal (QC), Canada

Pesquisa - Parking Lots



Professor - Pesquisa

Grupo DSBD.

- dsbd.inf.ufpr.br

Nosso Hardware

5 servidores:

- 328 processadores.
- 3,3 TB de DRAM.
- GPUs para criação de modelos de IA
    - 4 GPUs NVidia A5000.
    - 1 GPU NVIdia A6000
    - 43.520 CUDA Cores.
    - 144 GB de memória de vídeo.

Paulo	Grégio	Zanata
		
DSBD +	Secret +	Hipes

Comece agora mesmo

Temos vagas para:

- Doutorado.
- Mestrado.

Infraestrutura Computacional

Vamos aos dados da disciplina de Infraestrutura Computacional.

Objetivos

- Compreender os conceitos de redes de computadores, Internet, Web e nuvem.
- Utilizar ferramentas básicas de gestão e manipulação de redes.
- Identificar as configurações de rede de máquinas locais.
- Acessar recursos em computadores remotos.
- Desenvolver serviços baseados na Web e na nuvem.

Avaliação

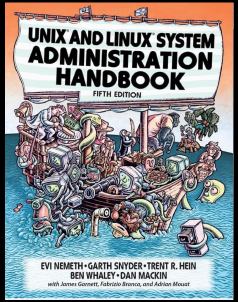
3 quizzes: 6% cada

1 Projeto: 12%

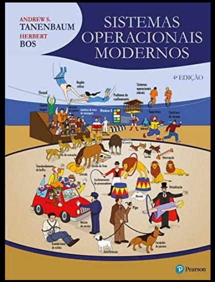
Presença nas aulas: 70%

Bibliografia

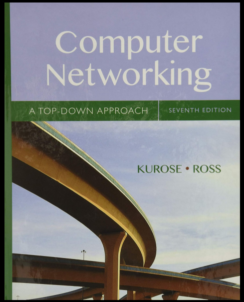
Snyder et al. UNIX and Linux System Administration Handbook. 5a ed. 2017.



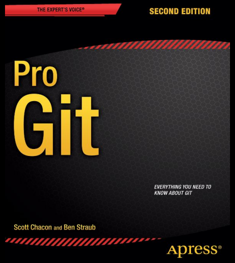
Tanenbaum, Bos. Sistemas operacionais modernos. 4a ed. 2016.



Kurose, Ross. Redes de computadores e a internet: uma abordagem top-down. 2013.



Chacon, Straub. Pro Git. 2a ed. 2014. Disponível em git-scm.com/book/en/v2



Bibliografia - Internet

Cuidado!

A internet é uma fonte importante de informações.

Bibliografia - Internet

Cuidado!

A internet é uma fonte importante de informações.

E uma fonte inesgotável de bobagens e pseudoespecialistas!

Seja criterioso ao pesquisar algum conceito na internet.

Na dúvida entre em contato com o professor.

Pergunta

O que você entende por “Internet”?

Pergunta

O que você entende por “Internet”?

O que é algo que está sendo executado na nuvem?

É isso...

Perguntas?

Licença