Exploração de Técnicas de Subamostragem para Otimizar Sistemas de Aprendizagem Automática na Nuvem

Pedro Gonçalo Mendes

Orientadores: Prof. Paolo Romano e Prof. João Nuno de Oliveira e Silva

Conteúdos

- Motivação
- Trabalho Relacionado
- Nephele
- Fabulinus
- Conclusão

TÉCNICO LISBOA

inesc id lisboa

Motivação

- A computação de aplicações de aprendizagem automática na nuvem é cada vez mais popular, devido à possibilidade de processar grandes quantidades de dados através da aquisição de recursos na nuvem;
- O preço a pagar pelo utilizador depende do tipo, da quantidade e do período de utilização dos recursos.

Motivação

Custo Estimado para treinar um modelo

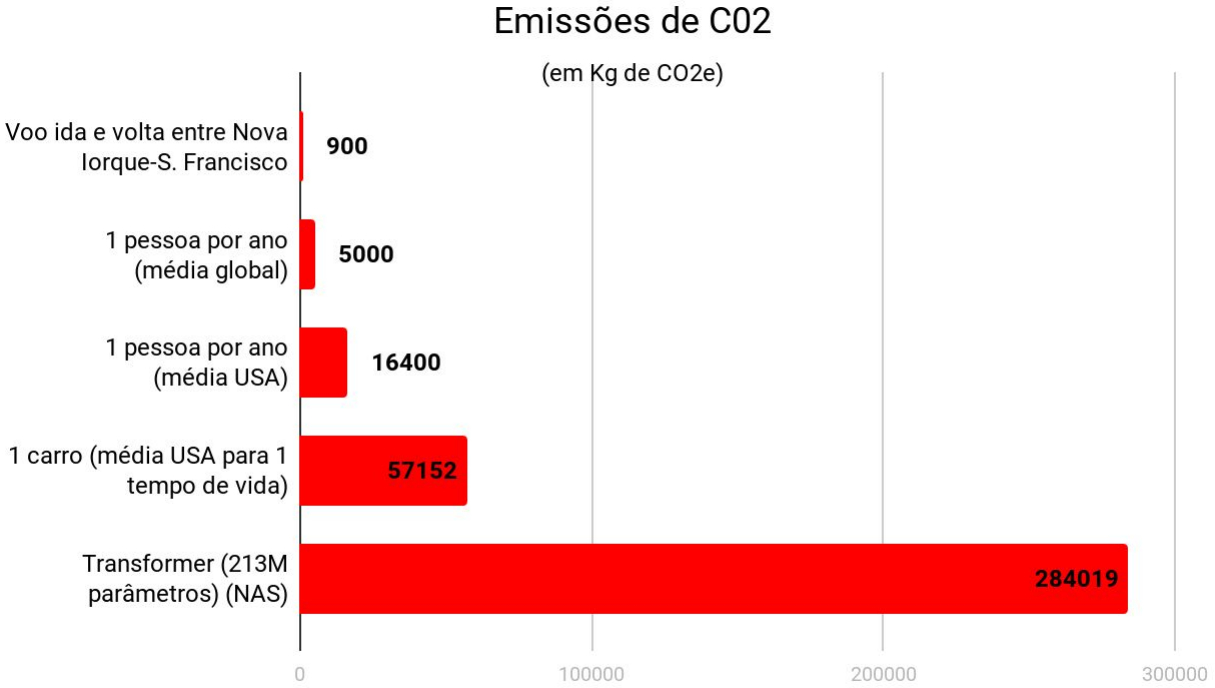
	Energia Consumida (kWh)	Emissões de CO2 (Kg de CO2e)	Custo de computação na nuvem (USD)
Transformer (65M parâmetros) (Jun,17)	27	12	$41-$140
Transformer (213M parâmetros) (Jun,17)	201	87	$289-$981
ELMo (Fev,18)	275	119	$433-$1,472
BERT (110M parameters) (Oct,18)	1507	652	$3,751-$12,571
Transformer (213M parâmetros) (NAS) (Jan, 19)	656347	284019	$942,973-$3,201,722
GPT-2 (Fev, 19)	-	-	$12,902-$43,008

Fonte: Energy and Policy Considerations for Deep Learning in NLP, Strubell et al. (2019)

TÉCNICO LISBOA

inesc id lisboa

Motivação



Fonte: Energy and Policy Considerations for Deep Learning in NLP, Strubell et al. (2019)

TÉCNICO LISBOA

inesc id lisboa

Motivação

- É fundamental a correta seleção dos recursos na nuvem para treinar modelos de aprendizagem automática;

- Exemplo de problemas de otimização:
  - Minimizar o custo de treino sujeito a restrições da qualidade do serviço;
  - Maximizar a precisão do modelo sujeito a uma restrição no custo de treino.

Exemplo de restrições da qualidade do serviço: limite máximo de tempo de treino, limite mínimo na precisão do modelo, limite no orçamento.

Motivação

- Existem 2 conjuntos principais de parâmetros que podem ser ajustados para aumentar a eficiência:

Parâmetros da nuvem

Parâmetros da aplicação

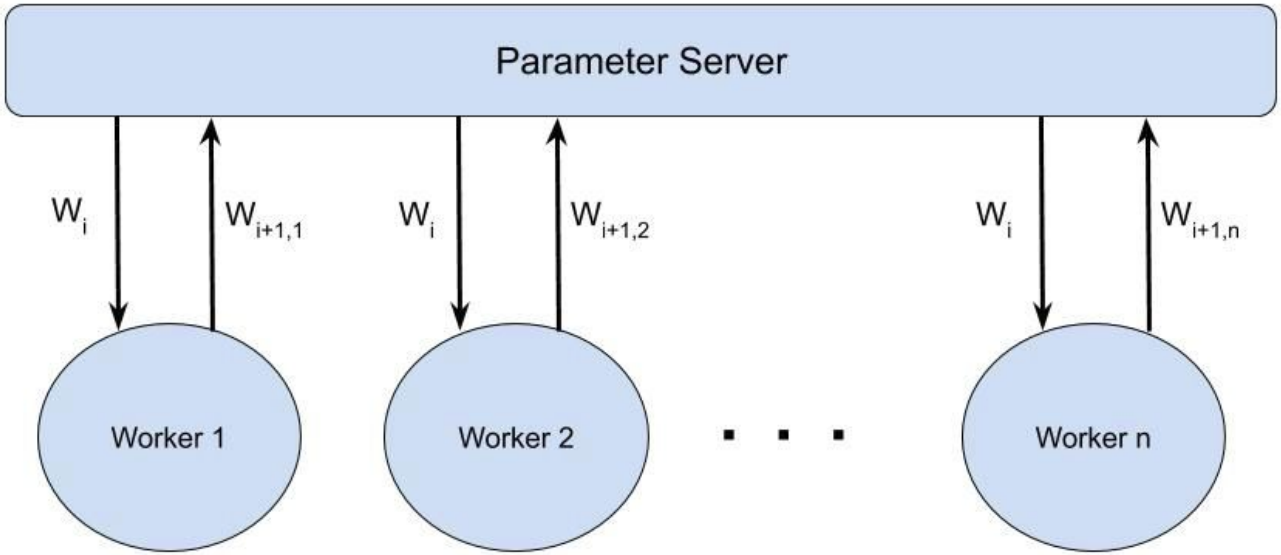
Motivação

- Parâmetros da nuvem
  - Poucas máquinas mas grandes (muitos vCPUs);
  - Muitas máquinas mas pequenas.

- Parâmetros da aplicação: Treino distribuído de Redes Neuronais

Motivação

Treino distribuído de Redes Neuronais



Motivação

- Parâmetros da aplicação: Treino distribuído de Redes Neuronais
  - Parâmetros do *Parameter Server*:
    - Modo de treino (sincronismo);
    - *Batch Size*;
    - *Learning Rate.*

Motivação

- É necessário a otimização conjunta destes dois grupos de parâmetros;

Ex:

  - Usar o modo de treino assíncrono em grandes *clusters* pode diminuir o tempo de treino quando comparado com o modo de treino síncrono em grandes *clusters*.

Trabalho Relacionado

	Parâm. App	Parâm. nuvem	Objetivo	Subamostragem	Restrições
Hyperband, 17	✓	✗	Maximizar precisão do modelo	✗	✗
BOHB, 18	✓	✗	Maximizar precisão do modelo	✗	✗
Fabolas, 16	✓	✗	Maximizar precisão do modelo	✓	✗
CherryPick, 17	✗	✓	Minimizar custo de treino	✗	Tempo
Lynceus, 18	✓	✓	Minimizar custo de treino	✗	Tempo, Precisão
Paris, 17	✗	✓	Maximizar precisão do modelo	✗	Custo
Scout, 18	✗	✓	Maximizar desempenho	✗	Custo
Nephele	✓	✓	Minimizar custo de treino	✓	Tempo, Precisão
Fabulinus	✓	✓	Maximizar precisão do modelo	✓	Custo

TÉCNICO LISBOA

inesc id lisboa

Objetivos

Otimizar a alocação de recursos na nuvem e os parâmetros da aplicação, explorando técnicas de subamostragem de forma a:

1. Reduzir o custo de treino assegurando níveis satisfatórios do tempo de execução e da qualidade do modelo;

2. Maximizar a precisão do modelo satisfazendo a restrição no custo máximo de treino.

TÉCNICO LISBOA

inesc id lisboa

Nephele

A redução do custo de treino é alcançada através dum *trade-off* entre a precisão do modelo e o tempo de treino:

-Subamostragem é usada para a reduzir custo de treino de forma controlada garantindo uma precisão mínima do modelo.

Nephele

Problema de Otimização:

$$\underset{x,s}{\text{minimizar}}\ C(x,s)$$

$$\text{sujeito a}\quad A(x,s) \geq A_{\min}$$

$$T(x,s) \leq T_{\max}$$

$$\sum_{(x_k,s_k)\in\mathcal{S}} C(x_k,s_k) \leq B$$

Custo

Configuração

Taxa de subamostragem

Precisão do modelo

Orçamento

Tempo de treino

Configurações testadas

TÉCNICO LISBOA

inesc id lisboa

Nephele

- Não assume nenhum conhecimento *a priori* sobre o modelo nem sobre a plataforma de treino:
  - Através da exploração de técnicas de Otimização Bayesiana (BO);

Otimização Bayesiana

- Método de procura do ótimo de uma função *black-box*;
- Constrói um modelo da função;
- Procura baseada num modelo da função;
- Avalia configurações de forma a construir e melhorar o modelo;
- Escolhe as configurações a avaliar de acordo com uma função de aquisição:
  - Melhoria Esperada (*Expected Improvement* EI);
  - Melhoria Esperada Restringida (*constrained Expected Improvement* EIc).

TÉCNICO LISBOA

inesc id lisboa

Nephele

- Dois modelos: Precisão e Custo;
- Explora N amostras aleatórias para inicialização dos modelos;
- Os modelos são atualizados e melhorados através da exploração de novas configurações;
- Seleção da nova configuração a avaliar baseado no rácio:

  Melhoria Esperada Restringida (EIc) / Custo;

Nephele

Avaliação:

- Sistemas:
  - Nephele;
  - Random Search;

  } Usando subamostragem

  - Lynceus: BO com EIc/custo;
  - CherryPick: BO com EIc;

  } Sem usar subamostragem

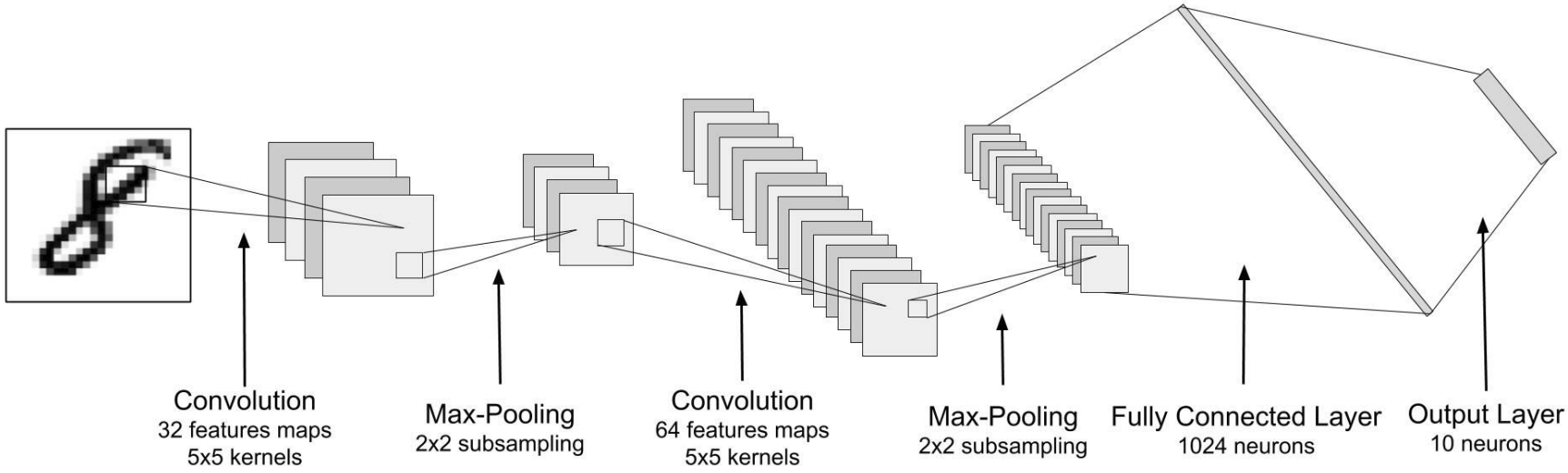
Conjunto de avaliação:

- 3 conjuntos de dados relativos ao treino de redes neuronais.

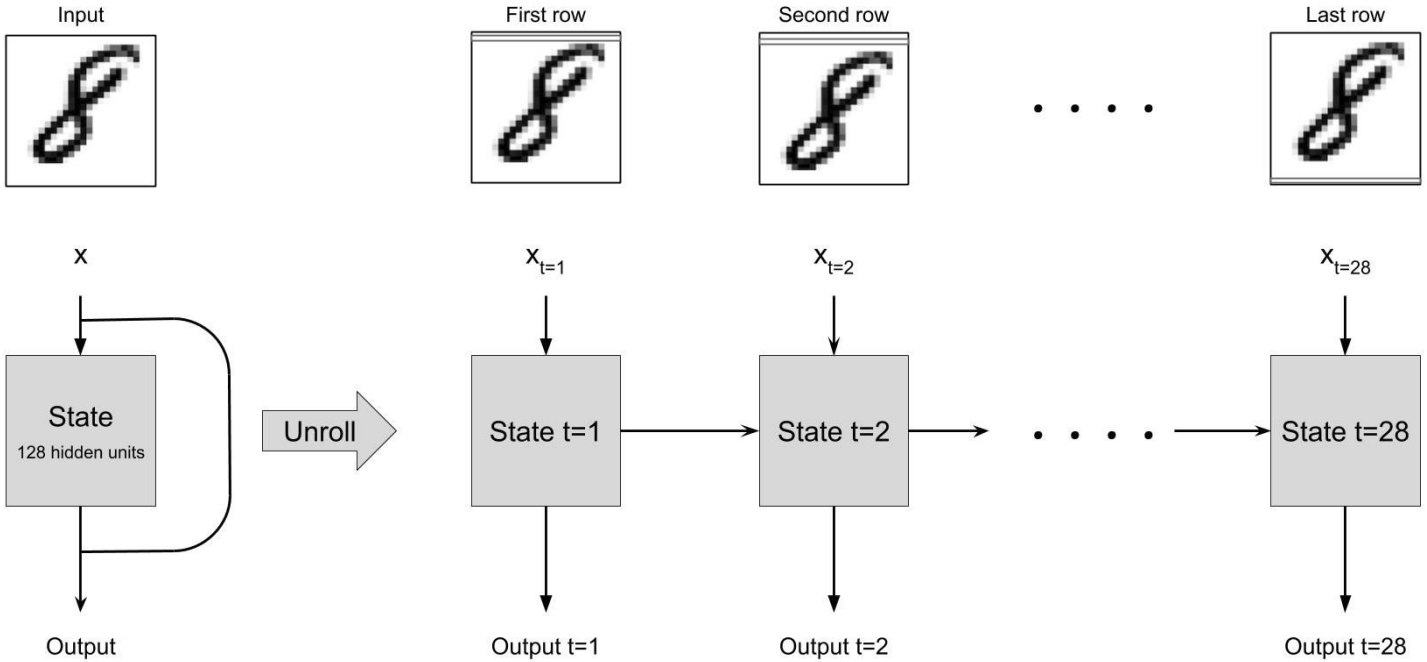
TÉCNICO LISBOA

inesc id lisboa

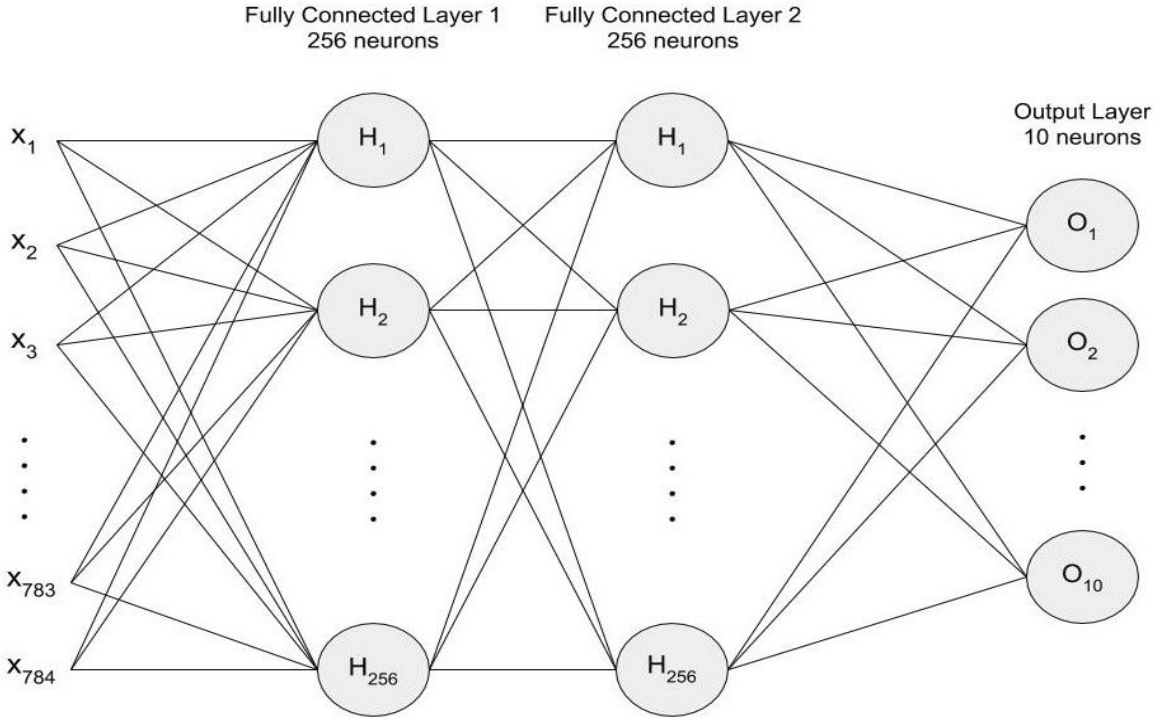
Rede Neuronal Convolucional



Rede Neuronal Recorrente



Rede Neuronal *Multilayer*



TÉCNICO LISBOA

inesc id lisboa

Casos de estudo usados para a avaliação

Configuração					
x					
Parâmetros da nuvem		Parâmetros do Parameter Server			
Tipo Vm	Número de Cores	Learning Rate	Batch Size	Mode de t	
t2.small t2.medium t2.xlarge t2.2xlarge	8 16 32 48 64 80	0.001 0.0001 0.00001	16 256	Síncrono Assíncrono	0.25 0.5 1

Total de 1440 configurações no espaço de procura

Nephele

- Cada execução é repetida 100 vezes;

- Métrica de avaliação:

Custo normalizado pelo ótimo

$$\frac{\text{Custo}}{\text{Custo}_{\text{ótimo}}}$$

Nephele

Custo de Produção: Treino de uma rede neuronal convolucional (CNN)

- A configuração ótima encontra-se no conjunto de dados $s = 0.1$

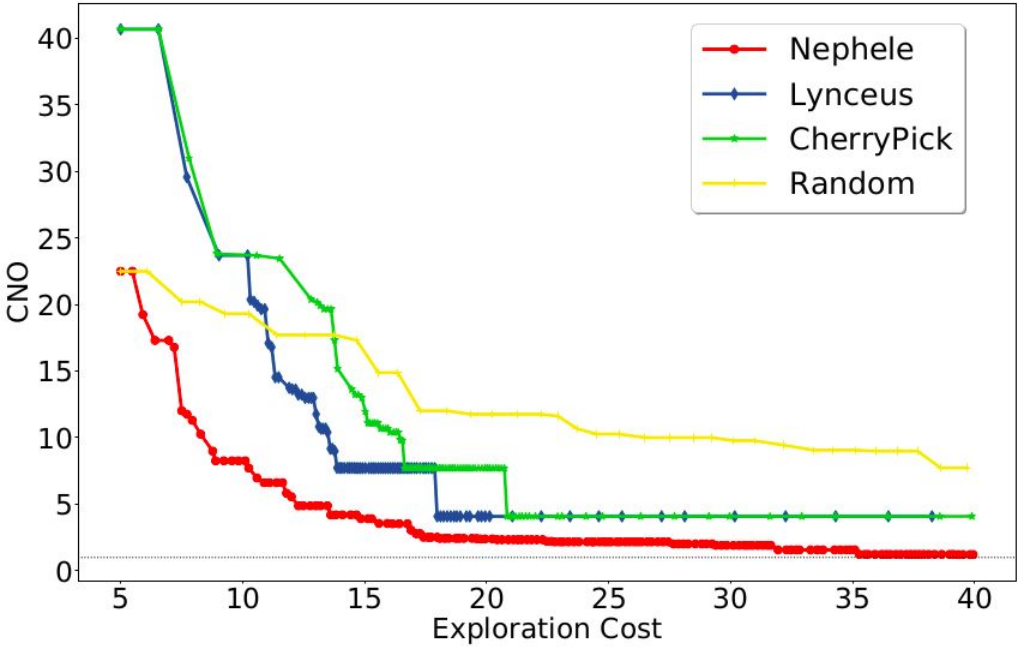
Possível usar um conjunto de dados subamostrados para minimizar o custo de treino assegurando a qualidade do serviço

TÉCNICO LISBOA

inesc id lisboa

Nephele

Custo de Otimização: Treino de uma rede neuronal convolucional (CNN)



Nephele **reduz o custo até 76%** comparado o Lynceus e CherryPick

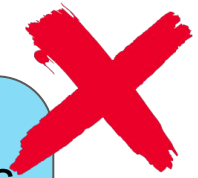
TÉCNICO LISBOA

inesc id lisboa

Nephele

Custo de Produção: Treino de uma rede neuronal *Multilayer*

- A configuração ótima encontra-se no conjunto de dados $s = 1$



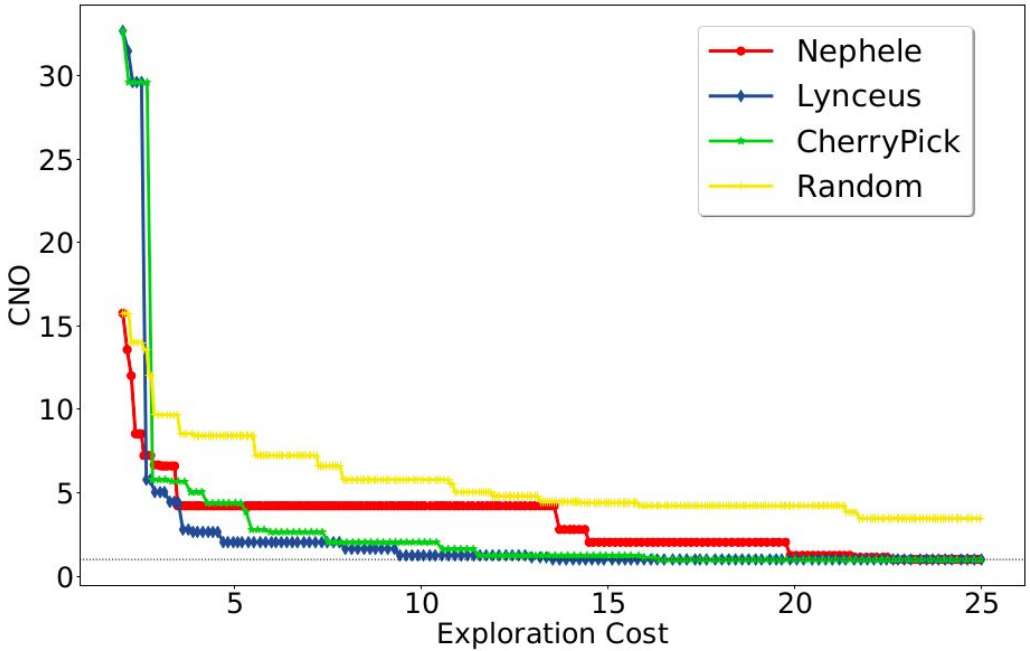
Não é possível usar um conjuntos de dados subamostrados para minimizar o custo de treino assegurando a qualidade do serviço

TÉCNICO LISBOA

inesc id lisboa

Nephele

Custo de Otimização: Treino de uma rede neuronal *Multilayer*



TECNICO LISBOA

inesc id lisboa

Objetivos

Otimizar a alocação de recursos na nuvem e os parâmetros da aplicação, explorando técnicas de subamostragem de forma a:

1. Reduzir o custo de treino assegurando níveis satisfatórios do tempo de execução e da qualidade do modelo;

2. Maximizar a precisão do modelo satisfazendo a restrição no custo máximo de treino.

Fabulinus

●Treina do modelo usando um conjunto de dados subamostrados para reduzir o custo de treino;

●O conhecimento proveniente destes treinos é transferido de forma a maximizar a precisão do modelo assegurando as restrições no custo usando o conjunto de dados completo.

Fabulinus

Problema de Otimização:

$$\begin{aligned} \underset{x}{\text{maximizar}} \quad & A(x, s = 1) \\ \text{sujeito a} \quad & C(x, s = 1) \leq C_{\max} \end{aligned}$$

TÉCNICO LISBOA

inesc id lisboa

Fabulinus

- Estende um sistema chamado Fabolas que considera um problema de otimização similar com duas principais exceções:

1.Não considera nenhuma restrição (na nuvem é desejável o controlo e/ou limite no custo);

2.Não otimiza a alocação de recursos na nuvem, mas apenas os parâmetros do modelo.

Fabulinus

- Estende uma função de aquisição chamada Entropy Search, que seleciona a configuração a testar que maximiza o ganho de informação sobre o ótimo usando o conjunto de dados completo;

$$a_F(x,s) = \frac{1}{C(x,s) + C_{overhead}} \mathbb{E}_{p(y|x,s,\mathcal{S})} \left[\int p^{s=1}_{max}(x'|\mathcal{S} \cup \{(x,s,y)\}) \cdot \log \frac{p^{s=1}_{max}(x'|\mathcal{S} \cup \{(x,s,y)\})}{u(x')} dx' \right]$$

TÉCNICO LISBOA

inesc id lisboa

Fabulinus

- Função de aquisição:

  Pesquisa por Entropia Restringida (Constrained Entropy Search (ESc))

$$\frac{\text{Ganho de Informação do ótimo}}{\text{Custo de treino + Custo de otimização}} \times \text{Precisão Esperada Restringida (CEA)}$$

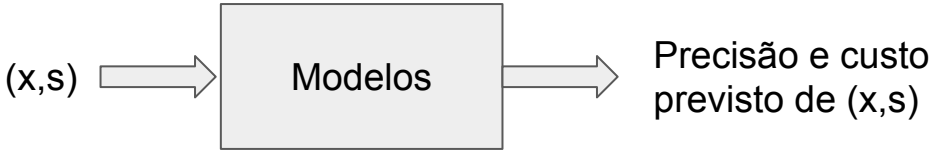
Fabulinus

- Precisão Esperada Restringida (CEA) mede a probabilidade da configuração **ótima prevista após esta exploração** cumprir as restrições;

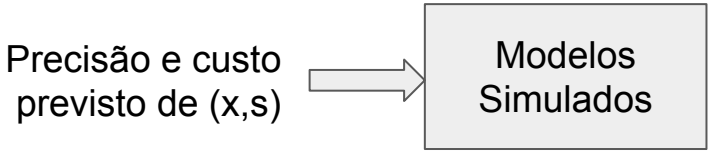
Precisão da configuração ótima prevista x Probabilidade do custo da configuração ótima prevista respeitar a restrição

Fabulinus

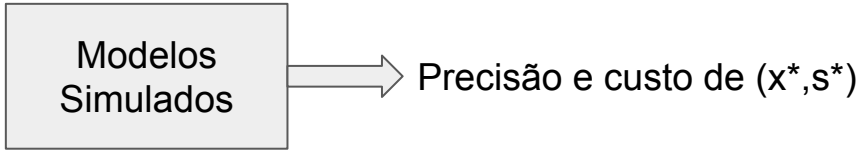
1. Previsão do custo e precisão;



2. Atualização dos modelos;



3. Cálculo da configuração ótima (x*,s*) e da respectiva precisão e custo;



TÉCNICO LISBOA

inesc id lisboa

Fabulinus

Avaliação:

- Sistemas:
  - Fabulinus;
  - Fabolas;
  - BO com EI;
  - BO com EIc.

Conjunto de avaliação:

- 3 conjuntos de dados relativos ao treino de redes neuronais.

Fabulinus

- Cada execução é repetida 10 vezes;

- Métrica de avaliação:

Precisão Restringida ($Accuracy_C$)

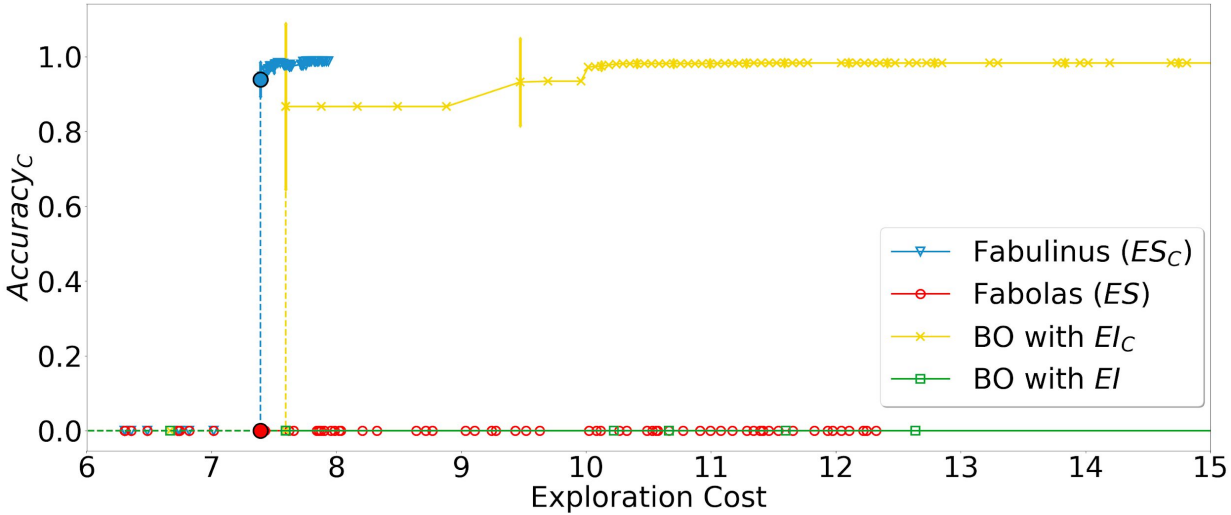
$$Accuracy_C\ (x,s) = \begin{cases} A(x,s), & \text{se } (x,s) \text{ é viável} \\ 0, & \text{caso contrário} \end{cases}$$

TÉCNICO LISBOA

inesc id lisboa

Fabulinus

Custo de Otimização: Treino de uma rede neuronal convolucional (CNN)



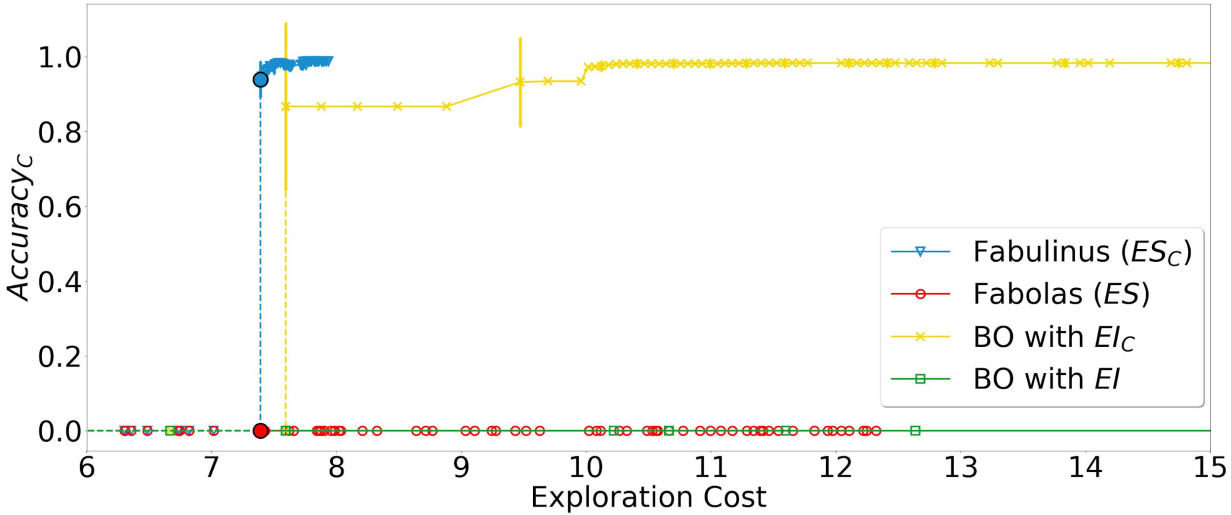
Fabulinus **reduz até 27% o custo** de otimização (comparando com BO usando subamostragem)

TÉCNICO LISBOA

inesc id lisboa

Fabulinus

Custo de Otimização: Treino de uma rede neuronal convolucional (CNN)



Fabulinus recomenda configurações que maximizam a precisão do modelo e respeitam a restrição de custo

TÉCNICO LISBOA

inesc id lisboa

Conclusão

- A utilização que técnicas de subamostragem permite reduzir o custo de treino e de otimização através de um compromisso entre a precisão do modelo requerida e o tempo de treino.

Conclusão

- Nephele consegue reduzir o custo até 76% em comparação com sistemas do estado de arte (para a qualidade do serviço requerida nas experiências);

- Fabulinus pode reduzir até 27% o custo de otimização para recomendar configurações que maximizam a precisão do modelo quando comparado com sistemas de otimização bayesiana sem subamostragem, ao mesmo tempo que assegura as restrições no custo.

TÉCNICO LISBOA

inesc id lisboa

Trabalho em Curso

- Desenvolvimento de novas heurísticas de forma a reduzir:
  - a complexidade computacional da função de aquisição proposta no sistema Fabulinus;
  - o custo de otimização (experiência adicionais usando diferentes orçamentos iniciais).