# Inteligência Artificial nas Redes de Computadores

Ivo Vilas Boas, Henrique Neto, and Daniel Sousa

Universidade do Minho, Departamento de Informática, 4710-057 Braga, Portugal e-mail: {a89492,a89618,a89562}@alunos.uminho.pt

**Resumo** Neste trabalho apresenta-se quatro tópicos diferentes no que consta a *Machine Learning*. Os tópicos são o funcionamento básico de uma *Machine Learning*,previsão e classificação de tráfego e performance e deteção de intrusos.

Em cada um foi feita uma abordagem generalizada com alguns exemplos concretos, pretendendo assim mostrar um pouco de cada tema.

Na parte do funcionamento foi abordado um modelo que consiste em seis etapas, Formulação do problema, Recolha de dados, Analise de dados, Construção do modelo,Validação e Lançamento e Inferência.

Na parte de Previsão e Classificação de tráfego, foram abordados essencialmente a previsão e a respetiva classificação do mesmo.

Na parte da Performance foi feita uma introdução geral da performance, especificando um exemplo concreto desta que e o QoE (*"Quality of Experience*).

Na parte de deteção de intrusos foi feita uma abordagem aos sistemas encarregados por este trabalho e as técnicas para detetar os intrusos baseadas em inteligência artificial.

## 1 Introdução

Com o crescimento e o sucesso da Internet na ultima década, cada vez existe mais preocupação com o desenvolvimento e manutenção das redes de computadores visto que cada rede tem as suas próprias necessidades, quer de recursos quer de desempenho, que normalmente variaram ao longo do tempo ou do espaço em que trabalham. Com esta complexidade e diversidade é bastante abundante a construção de novos algoritmos para cada rede desenvolvida, que revela-se uma tarefa difícil e apresenta vários desafios em garantir a eficiência da rede.

Assim ,com os avanços recentes de *machine learning*, tem-se relevado cada vez mais viável o uso destes algoritmos para lidar com a grande demanda atual. Neste ensaio iremos apresentar uma abordagem generalizada deste conceito.

## 2 Funcionamento Básico de uma *Machine Learnig Network*

Semelhante ao trabalho de uma aplicação de *machine learning* tradicional, uma *Machine Learnig Network* (MLN) segue um fluxo de características com varias etapas compostas por trabalhos quer manuais quer algorítmicos.

Sendo assim falaremos do conceito geral de uma MLN e, para esse efeito, consideraremos que o trabalho de implementação é separado em seis fases: formulação do problema, recolha de dados, analise de dados, construção do modelo, validação e por fim lançamento e inferência, como evidenciado na Figura 1.

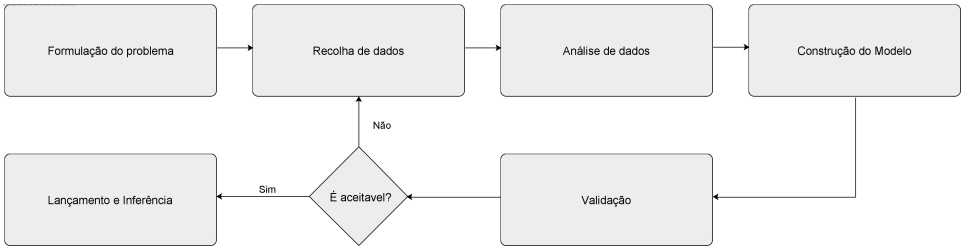


**Figura 1.** Fluxo de trabalho numa M.L.N.

### 2.1 Formulação do problema

Devido aos custos impostos por estes algoritmos, o problema inicial é formulado e categorizado, podendo ser visto depois como um problema de *machine learning* ou seja como um problema de classificação, de agregação, de regressão ou de decisão de problemas. Assim é definido o tipo de informação a ser recolhida e que modelos devem ser estudados. Caso o problema for abstraído de forma impropria o modelo final poderá não se adequar a rede alvo e por sua vez resultará num mau desempenho do algoritmo de aprendizagem. Por exemplo, para melhorar a experiência em um serviço de transmissão direta, será melhor formular um problema de exploração em tempo real em vez de um problema baseado em previsões [4].

### 2.2 Recolha de dados

Nesta fase o objetivo é armazenar uma grande quantidade de informação sem redundâncias ou padrões. Com isto vários dados (como registos de sessões ou de tráfego) são registados de vários pontos da rede de acordo com as necessidades do problema.

Neste contexto de *machine learning*, geralmente os dados são reunidos em duas fases, uma online e outra offline. Na fase offline, são recolhidos e armazenados vários registos e históricos para uma futura analise. Na fase online, informações do estado do desempenho da rede são fornecidas em tempo real ao modelo atual de aprendizagem. Adicionalmente estas informações também são usadas para atualizar os dados já presentes na adaptação atual do modelo.

### 2.3 Análise de dados

Uma rede de computares esta sempre a ser sujeita a vários fatores com origens diferentes porém, devido ao contexto do problema apenas alguns fatores provocaram diferenças significativas na eficiência e rapidez do sistema final. Por exemplo o tempo de ida e volta de uma mensagem na fase de reconhecimento é crucial na escolha da melhor implementação de um protocolo TCP[3], enquanto que noutros protocolos pode não ser tão relevante.

Tendo isto em conta, para atenuar os custos de tempo, torna-se necessário e importante processar e limpar a informação original, a por processos tais como normalização , calculo de valores em falta, etc. Porém em alguns casos é necessário conhecimento do domínio da rede e do problema inicial [4],o que dificulta muito o processo de analise e atrasa o processo de modulação, e que incentivam bastante o *machine learning* para esta etapa.

### 2.4 Construção do Modelo

Nesta fase é escolhido um algoritmo ou modelo inteligente, a partir de fatores como a quantidade de informação recolhida, a categoria do problema inicial, as condições físicas

da rede alvo, entre outros. Posteriormente os dados provenientes da fase offline da recolha de dados, serão usados para treinar o modelo por uma técnica chamada *hyper-parameter tunning*. Desta forma o modelo da rede evolui e aprende os melhores métodos para trabalhar na rede em que será implementado.

### 2.5 Validação

Nesta fase, o modelo é posto sobre vários testes de validação a partir da informação em registo, para testar a sua precisão, e porventura concluir se este se encontra sobreajustado ou subajustado as características analisadas anteriormente. A partir disto é possível otimizar o modelo, aumentar o volume de dados e ,caso este esteja sobreajustado, reduzir a complexidade original.

Também são analisadas amostras de dados erradas para determinar se o modelo e suas características se adequam ao tráfego que será exposto e também para confirmar se a informação testada é representativa o suficiente do sistema real. Caso o o modelo não seja adequado, as etapas anteriores terão de ser refeitas e sujeitos a novos testes e novos dados.

### 2.6 Lançamento e Inferência

Após ser validado, o modelo é implementado no ambiente da rede operacional e posteriormente existe a possibilidade de surgirem problemas no modelo. Isto deve-se ao facto de poderem existir limitações computacionais, energéticas devido as trocas feitas entre precisão da rede e gestão de congestionamento que influenciam o desempenho do sistema final. Ainda mais, *machine learning* trabalha para obter o modelo ótimo do problema e assim assegura melhorias na performance.

## 3 Previsão e Classificação de tráfego

As primeiras implementações das *machine learning networks* forem feitas como algoritmos de previsão e classificação de tráfego, devido ao facto destas questões serem fulcrais para o desempenho de vários elementos nas redes e estão sempre sujeitas a novos estudos.

### 3.1 Previsão de tráfego

Numa rede de computadores, a previsão do tráfego e do débito de informação permite que haja uma melhor gestão de congestionamento, alocação de recursos, encaminhamento de dados, e em alguns casos melhor qualidade de serviços que decorrem em tempo real. Sendo assim é possível explorar esta técnica de duas formas que são distinguidas por ou realizarem observações diretas ou não. Tendo em conta que as medições diretas tem custos bastante elevados, especialmente em situações de grande transito e debito de informação, a eficiência destes métodos varia conforme a rede destino e as implementações feitas.

Sendo assim, este problema pode ser abordado de duas maneiras diferentes. Uma maneira exige que um ser humano invista bastante tempo de trabalho a implementar um sistema de algoritmos sofisticado, estudando o domínio da rede á procura de padrões que ainda não estão a ser contabilizados. Outra maneira é usar *machine learning* para fazer esse trabalho pesado. Por exemplo, existem modelos que tentam prever o debito de informação futuro a partir da quantidade e do tamanho do tráfego recente[5]. Outra abordagem envolvem apenas o estudo entre os fins da rede. Desta forma podemos simplesmente basear-nos no modelo da rede e no cabeçalho de cada mensagem para conseguir previsões do tráfego[6].

### 3.2 Classificação de tráfego

A classificação de tráfego é um fator critico na gestão de rede e de sistemas de segurança, visto que, é responsável pela correspondência de aplicações e protocolos com o fluxo de tráfego. Tradicionalmente é implementado por métodos baseados em portas ou na carga útil. Ora, o método baseado de portas pode nem sempre ser eficiente devido erros provenientes da reutilização de atribuições, enquanto que o método baseado em carga útil sofre com problemas de privacidade, que por vezes provocam falhas em alguns sistemas.

Tendo isto em conta, vários métodos de *machine learning* baseado em características estatísticas tem sido bastante estudadas nos anos recentes, especialmente nos domínios de segurança. Porém não é fácil considerar um algoritmo de *machine learning* como uma solução omnipotente por isso ainda é raro encontrar estes procedimentos no mundo atual. Por exemplo, ao contrario de *machine learning* tradicional, neste contexto de segurança de rede, a classificação de uma imagem como uma pessoa poderia causar muitos atrasos e custos caso a categorização estivesse errada.

## 4 Performance

Como vimos anteriormente, a resolução de problemas com elevado nível de complexidade é uma das principais vantagens de uma *Machine Learning Network*, uma vez que esta exibe níveis de performance semelhantes ou até superiores ao dos ser-humano.

De modo a obter a esses resultados as decisões tomadas pela maquina são influenciadas pelas mais diversas técnicas. Exemplos de aplicações destas técnicas São no QoE de um vídeo, o melhor canal wireless,entre outras.

### 4.1 QoS("quality of system") e do QoE("quality of experience")

A experiência do utilizador(QoE), ao contrario da experiência do sistema, tem um vasto numero de parâmetros que influenciam a qualidade recebida pelas utilizadores. Enquanto a QoS pode ser medida através do bit rate, latência e o erro dos servidores multimédia e a QoE não pode. A razão é simples, por exemplo, se estiver a ser feita uma conferencia através da rede, o propósito da transmissão de dados é para obter a informação do áudio e do vídeo no momento em que se usa o serviço. Devido à maneira como nos recebemos essas duas informações esta não e a mesma como se fosse num ficheiro transferido desse mesmo serviço.

Desse modo a condição necessária para uma entrega de uma rede multimédia de alta qualidade é perceber como a qualidade é recebida pelos utilizadores os parâmetros que a afetam e quanto a afetam. E é aqui que entra o uso de *Machine Learning* em termos da performance da rede e redução de custos.

Existem várias técnicas que podem ser utilizadas bem como vários algoritmos, como por exemplo , MOS (*Mean Opinion Score*),PSNR(*Peak Signal-To-Noise Ratio*). A técnica MOS baseia-se apenas na opinião do utilizador. Após a atribuição de uma pontuação por parte do utilizador, o sistema, com os dados atribuídos cria uma função matemática e gera um novo modelo baseada na função. O Modelo PSNR é parecido mas em vez de usar a opinião dos utilizadores utiliza a relação entre a energia máxima de um sinal e o ruído que afeta a sua transmissão fidedigna. Para este método e usada uma escala logarítmica como medição e,da mesma forma que o MOS, gera um novo modelo de streaming baseado na escala obtida.

Outros métodos usados são aqueles que prevêem mudanças na QoS em vez de só mudarem após a transmissão do serviço,ou seja, estes são a nível da rede.

Neste caso o *Machine Learning* e usado para prever eventuais mudanças na rede e para selecionar a resposta de adaptação mais adequada.Como exemplo destes modelos temos o CS2P("Cross Session Stateful Predictor"), o CFA ("Critical Feature Analytics").

O CS2P usa um outro modelo para modelar a evolução da taxa de transferência, e outro por grupos de parâmetros comuns.

Este modelo é treinado de maneira offline e o modelo é alterado conforme a taxa de transferência da rede. Dados revelam que este modelo aumenta em media 3,2 % do QoE.

Já o CFA usa um modelo que e determinado por poucos parâmetros críticos dos quais a qualidade do vídeo é definida. Assim o modelo aprende automaticamente estes parâmetros para diferentes sessões de vídeo num curto período de tempo sendo as alterações do modelo quase em tempo real.

# 5 Deteção de intrusos

Com a expansão exponencial das redes de computadores nos últimos anos, ataques a estas têm se tornada cada vez mais um tópico de discussão e como tal, uma solução que surgiu para aumentar a segurança foi a implementação de Inteligências Artificiais nas redes com o âmbito de detectar ataques e/ou intrusos nestas redes.

## 5.1 Sistemas de Deteção de Intrusos

Um Sistema de Deteção de intrusos (SDI) é definido como "um sistema eficaz de segurança capaz de detetar, prever e possivelmente reagir a ataques". O principal objetivo de um SDI é detetar todos os intrusos numa rede especifica de forma eficiente. A implementação de um sistema destes permite que os administradores da rede na qual foi implantada assegurem a manutenção de protocolos de segurança, tais como impedir o acesso de atacantes externos a informações críticas guardadas na rede ou até mesmo impedir que os usuários "dignos" da rede tenham acesso aos recursos da mesma. Assim, um SDI baseado numa AI é capaz de manter a segurança da rede através do uso de uma AI capaz de realizar a recolha de dados da rede, processamento e filtragem destes dados, e no caso de deteção de uma anomalia nestes, sinalizar os administradores da rede, ou até mesmo proceder à aplicação de contra-medidas em resposta a estes atacantes externos.[7]

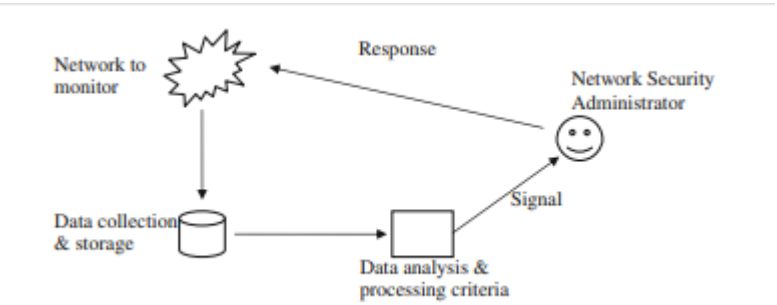


**Figura 2.** Arquitetura de um SDI

## 5.2 Técnicas para a deteção de intrusos baseadas em IA

Um SDI não necessita obrigatoriamente de ser baseado numa IA, por exemplo, existem sistemas baseados em métodos estatísticos e também no conhecimento da rede. No entanto, existe aqueles que defendem que os sistemas baseados em IA mostram-se vantajosos em relação a outros métodos devido a certas vantagens que estas apresentam, tais como: uma

grande flexibilidade e adaptabilidade comparada com outros métodos, reconhecimento de padrões e deteção de novos padrões de dados nada rede e uma grande velocidade na computação, analise e tratamento dos dados (uma velocidade bastante superior à de um ser humano).

Existem diversas técnicas baseadas em IA e na tabela 3 podemos visualizar algumas dessas técnicas, bem como o critério de processamento de dados que estas usam, a fonte dos dados "base" da rede e o tipo de resposta efetuados por estas técnicas.

| Name of system | Processing criteria | Source of audit data | Type of response |
|---|---|---|---|
| NSM (Heberlein et al. 1990) | Hybrid | N/w | Passive |
| Bro (Paxson 1998) | Signature | N/w | Passive |
| MIDAS (Sebring et al. 1988) | Hybrid | Host | Passive |
| Haystack (Smaha 1988) | Hybrid | Host | Passive |
| IDES (Lunt et al. 1992) | Anomaly | Host | Passive |
| W & S (Vaccaro and Liepins 1989) | Anomaly | Host | Passive |
| Comp Watch (Dowell et al. 1990) | Anomaly | Host | Passive |
| ASAX (Habra et al. 1992) | Signature | Host | Passive |
| USTAT (Ilgun et al. 1995) | Signature | Host | Passive |
| IDIOT (Crosbie et al. 1996) | Signature | Host | Passive |
| GrIDS (Chen et al. 1996) | Hybrid | Hybrid | Passive |
| NIDES (Anderson et al. 1995) | Hybrid | Host | Passive |
| EMERLARD (Porras and Neumann 1997) | Hybrid | Hybrid | Active |
| Janus (Goldberg et al. 1996) | Signature | Host | Active |
| Tripwire (Kim and Spafford 1997) | Signature | Host | Passive |
| OSSEC HIDS (Hay et al. 2008) | Hybrid | Host | Active |
| Snort (Beale et al. 2004) | Hybrid | N/w | Active |
| AAFID (Spafford and Zamboni 2000) | Anomaly | Host | Active |
| NAIDR (Hochberg et al. 1993) | Anomaly | N/w | Passive |
| OSSEC HIDS (Hay et al. 2008) | Hybrid | Host | Active |
| RealSecure (Internet Security Systems (ISS) 2010) | Signature | Hybrid | Active |

**Figura 3.** Alguns exemplos de técnicas baseadas em AIs

Podemos ver três diferentes critérios de processamento, estes sendo *Hybrid*, *Signature* e *Anomaly*. *Anomaly* refere-se a sistemas construídos com base em dados que são considerados normais numa dada rede, e qualquer derivação destes será considerado um ataque ou anomalia. No entanto nem sempre uma anomalia nos dados é causada por um ataque externo e dessa noção surgiram os sistemas com critérios de processamento *Hybrid*, no qual para alem da procura de anomalias nos dados, são também utilizadas outras técnicas para tentar descartar anomalias não causadas por ataques. Por fim temos os sistemas baseados em *Signature* nos quais é observada a assinatura dos dados analisados na procura por assinaturas estrangeiras à rede que possam implicar um ataque à mesma.[8]

Nos tipos de resposta dos diferentes sistemas, *passive* refere-se a uma resposta no qual o sistema apenas avisa um administrador da rede da possibilidade de um intruso na mesma, e *active* refere-se a uma resposta ativa do sistema, nos quais o próprio sistema toma medidas de resposta face aos intrusos.

## 6 Conclusões

Neste ensaio, compreendemos que com a dimensão e diversidade das redes de computadores atuais e concordamos que o uso de técnicas de *machine learning* é imperativo para crescimento e para desenvolvimento de novas tecnologias de comunicação. Porém também nos foi evidente que não é fácil implementar estes sistemas e que ainda é necessária bastante investigação para viabilizar estes modelos.

## Referências


1. Mowei Wang, Yong Cui, Xin Wang, Shihan Xiao, and Junchen Jiang : "Machine Learning for Networking: Workflow, Advances and Opportunities", IEEE Network
2. Raouf Boutaba, Mohammad A. Salahuddin, Noura Limam, Sara Ayoubi, Nashid Shahriar, Felipe Estrada-Solano, and Oscar M. Caicedo, "A comprehensive survey on machine learning for networking: evolution, applications and research opportunities"
3. K. Winstein and H. Balakrishnan, “TCP Ex Machina: Computer- Generated Congestion Control,” Proc. ACM SIGCOMM Computer Commun. Rev.
4. J. Jiang et al., “CFA: A Practical Prediction System for Video QoE Optimization,” Proc. NSDI 2016
5. Z. Chen, J. Wen, and Y. Geng, “Predicting Future Traffic Using Hidden Markov Models,” Proc. IEEE 24th Int’l. Conf. Network Protocols (ICNP) 2016
6. P. Poupart et al., “Online Flow Size Prediction for Improved Network Routing,” Int’l. Conf. Network Protocols (ICNP)
7. Boutaba et al. Journal of Internet Services and Applications, "A comprehensive survey on machine learning for networking: evolution, applications and research opportunities", Raouf Boutaba1, Mohammad A. Salahuddin, Noura Limam, Sara Ayoubi, Nashid Shahriar, Felipe Estrada-Solano and Oscar M. Caicedo
8. Djabeur Mohamed Seifeddine Zekrifa, "Hybrid Intrusion Detection System", Computer Science.