A nível do conteúdo, a secção do *resumo* encontra-se muito genérica, sem focar os aspetos principais do artigo, nomeadamente os temas que vão ser abordados e de que forma o artigo se estrutura e desenrola sobre os mesmos.

A abordagem a processos de aprendizagem baseada em casos, apesar de varrer as principais características deste algoritmo, podia ter focado outros aspetos relevantes.

Se de facto este processo pode ser resumido nas fases de procura, reutilização, revisão e aprendizagem, faltou a referência detalhada ao conceito essencial que está por detrás deste processo, o conceito de similaridade.

Ou seja, perante um novo problema, o algoritmo começa por reunir os exemplos passados da biblioteca de casos, similares ao problema atual. Contudo, esta seleção não é feita de forma aleatória ou por um simples patern matching de atributos. São atribuídos pesos e uma certa flexibilidade aos valores que este podem tomar, de forma a reunir os casos similares.

O encontro de soluções similares com o problema atual, também não exige que especificamente se aplique a mesma solução no contexto atual. Uma adaptação dos passos da resolução pode ser realizada, adaptando a solução passada ás características do problema que se quer ver resolvido no presente.

Supondo um exemplo simples onde procuramos resolver um problema de um automóvel, se houver um caso na base de conhecimento cujo problema seja “Luz da frente queimada” e a solução “instalar lâmpada nova na luz frente”, se no presente surgir o problema “Luz da ótica traseira queimada”, o caso da “luz da frente queimada” vai possivelmente apresentar uma forte similaridade. Contudo, a solução aplicada não será mudar a luz da frente, mas sim adaptar essa solução ao contexto e mudara lâmpada da luz traseira.

Outra característica que apesar de presente podia estar mais realçada, é a explicação da forma como o algoritmo realiza aprendizagem. Este processo ocorre quando se verifica que a ação desempenhada como solução, corrigiu de facto o problema e então, o par (problema\_Atual, Solução\_Aplicada) podem ser introduzidos como conhecimento novo na base de conhecimento.

A nível das ferramentas de desenvolvimento e soluções existentes no mercado é apresentado um número aceitável de exemplos reais. No caso das aplicações do algoritmo no contexto real, podia talvez ter sido dado um exemplo especifico numa determinada área da sociedade.

Tendo agora como foco de análise a descrição dos algoritmos genéticos, na abordagem ao processo de codificação de soluções no algoritmos genéticos, é referido que “Cada individuo é caracterizado por um conjunto de propriedades, o seu genótipo, cada um destes é representado por vetores binários onde cada elemento de um vetor denota a presença (1) ou ausência (0) de uma determinada característica”.

Apesar desta descrição não estar tecnicamente incorreta, sem ter sido especificado o seu contexto, é na minha opinião uma visão muito restrita e limitada à forma como atualmente se pode codificar um cromossoma em algoritmos genéticos.

Nas implementações iniciais deste algoritmo, os cromossomas de uma população eram de facto uma sequencia de bits. Contudo, não é necessariamente verdade que cada bit represente a presença ou ausência do uma característica. Basta para isso pensar que se uma característica tiver mais do que duas hipóteses, essa representação seria inútil.

Numa representação binária, um bit ou subconjunto de bits podem representar alternativas sobre um determinado parâmetro da solução.

Em implementações mais recentes, os elementos de uma população podem já ser strings, números inteiros ou até mesmo árvores de decisão.

Ainda dentro das principais etapas do algoritmo, seria relevante mencionar que as fases de codificação e criação da função de fitness são as mais complexas e as únicas que exigem ter em atenção o contexto do problema. Apesar do algoritmo em si ser genético para qualquer contexto de aplicação, tem que se pensar numa forma de representar os atributos de uma solução na forma de um cromossoma e, mais do que isso, criar uma função de fitness capaz de interpretar os valores de um cromossoma.

Esta função deve ser capaz de analisar o desempenho dessa solução, quase que de forma análogo a um processo de reforço, mas sem que a mesma seja efetivamente executada.

De referir ainda que, se de facto podemos olhar para o melhor individuo da geração final e o considerar como a melhor solução, é também verdade que qualquer elemento dessa geração final pode ser visto como outras aproximações à solução ótima.

A referência ás ferramentas de desenvolvimento e soluções implementadas com base em GA foi bem conseguida e detalha casos diversificados.

A introdução ao contexto das redes neuronais fazendo um paralelismo com a complexidade do cérebro humano foi muito bem conseguida, comparativamente com o arranque inicial dos outros dois temas anteriores.

Existe uma referencia aos conceitos gerais destes processos de aprendizagem, ao tipos de aprendizagem que um sistema pode ter e ás principais formas da rede aprender e alterar o peso dos seus neurónios.

Esta seção é assim a mais equilibrada a nível de conteúdo relevante e detalhe nas informações.

No global, tendo em conta que em alguns contextos os artigos científicos estão limitados a um número reduzido de páginas, a preocupação em focar apenas os aspetos principais de cada processo de aprendizagem seria uma mais valia.

Contudo, sendo que neste contexto não existia esta limitação e se procurava uma abordagem com mais detalhe, certos aspetos podiam ter sido abordados com mais enfase.

A nível do formato Lecture notes, não existe nenhum erro ou aspeto a apontar. O artigo segue as regras formais desse template.

**Paper 19**

A nível de conteúdo, a seção resumo apresenta-se bem escrita, com referencia direta aos temas e metodologias usadas para os abordar.

A introdução ao processo de raciocínio baseado em casos apresenta-se concisa, fornecendo logo no inicio e de forma clara os principais conceitos deste tipo de algoritmo: a utilização de casos passados semelhantes para resolver um novo problema e a adição de novos casos resolvidos á biblioteca de casos, como forma de aprendizagem.

Referir que CBR não se trata apenas de um único algoritmo, mas sim de uma designação geral que cobre um conjunto de métodos, permite leitor interpretar as seguintes seções num domínio de aplicação muito superior. A citação destes métodos mais específicos de aprendizagem baseada em similaridade foi assim um bom ponto de partida.

O desenvolvimento das várias fases do algoritmo está bem detalhado e construído, assim como os exemplos de aplicação do algoritmo em situações reais.

Relativamente ao tópico das árvores de Decisão, na secção 4.1, é descrito de que forma se realiza a indução neste tipo de algoritmos. Contudo, a primeira frase da referida seção pode ser vista como um pouco subjetiva para alguém que não esteja familiarizado com estes métodos de aprendizagem.

Uma abordagem mais direta, como por exemplo “A base para a indução parte da análise de um dataset de dados, onde cada entrada possui valores para os diferentes atributos e, conforme esses valores, a entrada recai sobre uma das classes do problema.” Depois desta referência mais simples a um conjunto de dados inicial, o resto da secção apresenta-se muito bem construída para explicar como se forma uma árvore de decisão.

Ainda neste tema, podia ter ser referido o problema de *overffiting*. Pelo facto da aprendizagem ser supervisionada, o modelo de decisão gerado pela iteração do algoritmo pode ajustar-se bastante bem ao conjunto de dados inicial, que usou para treino e criação da árvore, mas pode ser menos apto para modelar novos casos futuros.

Na secção 5.1, relativamente ao funcionamento das Support Vector Machines (SVM), é afirmado que o conjunto de dados inicial é projetado para uma representação cartesiana de N dimensões, onde N representa o número de atributos dos dados.

Contudo, é limitar a aplicação do algoritmo afirmar que os pontos são catalogados como pertencentes a apenas uma de duas classes, dado que o algoritmo pode ser adaptado a mais do que duas classes.

Relativamente ao método do kernel, podia ter sido referido de forma mais direta que este método se baseia num aumento nas dimensões do espaço sobre o qual os dados são representados.

Apesar de na perceção humana, imaginar visualmente representações acima de um espaço a 3 dimensões seja impossível, para o algoritmo SVM, a representação dos dados numa dimensão acima do original torna o seu processo mais simples, ajudando o algoritmo a encontrar as melhores fronteiras entre as classes e encontrar assim hiperplanos que as separem.

Nos exemplos de aplicação dos algoritmos em soluções do dia a dia, podiam talvez ter sido dadas referências aos estudos ou artigos relativos aos projetos baseados em SVMs.

Numa análise global, em nenhum dos três sistemas de aprendizagem foram referidos exemplos ou métodos que permitam utilizar estes processos de aprendizagem. Por exemplo, no caso das árvores de decisão, o algoritmo C4.5 está incorporado na ferramenta Weka, sobre a designação J48. O original desenvolvido na linguagem C e o segundo, desenvolvido na linguagem java sobre a que se baseia o weka.

A conclusão aborda de forma geral as principais características de cada processo de

aprendizagem abordado, citando alguns dos aspetos positivos de cada algoritmos.

Teria sido interessante neste segmento dar um maior relevo entre as eventuais

vantagens de cada processo, mas citando também as suas limitações de representação e

desvantagens .