

Universidade de Lisboa - Instituto Superior Técnico  
Licenciatura em Engenharia Informática e de Computadores  
Inteligência Artificial

## 2º Projeto - Grupo 22

Gonçalo Marques, 84719

Manuel Sousa, 84740

### P1

Para resolver este problema, começámos por elaborar um conjunto de features a aplicar sobre as palavras. De início contruímos features básicas que verificassem o numero de vogais e consoantes de uma palavra, o numero e acentos, etc. O primeiro objetivo passava apenas por estudar o comportamento do avaliador, e durante este processo, facilmente concluímos que quanto maior fosse a unicidade da feature, isto é, quanto mais unico fosse o output da feature em relação à palavra recebida, menor seria o erro. De seguida elaborámos uma função que somava o ASCII dos caracteres constituintes da palavra, e desta maneira, reduzimos o erro substancialmente, visto que o output dado pela feature será sempre unico, menos quando palavras diferentes fossem constituídas pelos mesmos caracteres (daria o mesmo resultado). Posto isto, criámos uma função que desse um output unico para cada palavra recebida, do genero de uma função de hash. Existe um mecanismo em python que o permite fazer, e com a utilização de uma feature assim conseguimos 0% de erro!

Tabela 1: Analise dos erros com a evolução das fetures

	Teste 1	Teste 2
N Acentos + N Vogais + N Consoantes	0.259615384615	0.259615384615
N Consoantes + Caracteres para ASCII	0.0697115384615	0.0697115384615
N Consoantes + Caracteres para ASCII + Hash	0.0	0.0

Pela analise da tabela concluímos que todos os testes escolhidos têm uma percentagem de erro aceitavel. No entanto a ultima opção é a mais viável porque garante uma percentagem de erro de 0% graças à função de gera um numero unico para cada string.

### P2

Texto.

## P3

Trajetória do ambiente 1:

Tabela 2: Trajetória 1

5	0	6	0
6	0	6	1
6	0	6	1
6	0	6	1

Trajetória do ambiente 2:

Tabela 3: Trajetória 2

5	0	6	0
6	0	1	1
1	0	2	0
2	1	1	0

A função de recompensa é a função de Q-Learning (inserir função Q-Learning), com uma learning rate (alpha), ou a importância de nova informação face à informação já aprendida, de 0.14, e um discount factor (gamma), ou a importância de recompensas atuais face a futuras, de 0.9.

O agente move-se sempre para o estado que tem uma maior recompensa, de acordo com a implementação da função Q2pol.