

**Dados, Informação, Conhecimento,  
Ensino de Programação, a Inteligência Empresarial  
e as Suas Motivações**

RELATÓRIO DE MESTRADO

**António Paulo Dias Pestana**  
MESTRADO EM ENGENHARIA INFORMÁTICA

**Dados, Informação, Conhecimento,  
Ensino de Programação, a Inteligência Empresarial  
e as Suas Motivações**

RELATÓRIO DE MESTRADO

**António Paulo Dias Pestana**

MESTRADO EM ENGENHARIA INFORMÁTICA

ORIENTAÇÃO  
Eduardo Leopoldo Fermé

**Orientador:**

Professor Doutor Eduardo Leopoldo Fermé

*Professor Associado no Centro de Ciências Exatas e da Engenharia da  
Universidade da Madeira*

## **Título da dissertação**

Dados, Informação, Conhecimento, Ensino de Programação, a Inteligência Empresarial e as suas Motivações

## **Copyright**

António Paulo Dias Pestana

A Universidade da Madeira tem o direito, perpétuo e sem limites geográficos, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel, de forma digital, ou por qualquer meio conhecido ou venha a ser inventado, de a divulgar através de repositórios científicos e admitir a sua cópia e distribuição com objetivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.

## **Agradecimentos**

Pretendo, através destas linhas, expressar a minha gratidão não só pela ajuda no desenvolvimento desta dissertação, mas também pela carreira que tenho vindo a construir, a qual considero positiva.

Em primeiro lugar, quero agradecer ao Professor Doutor Eduardo Fermé, por me ter orientado no desenvolvimento desta dissertação, com a incansável ajuda que me deu, não só através de correções, bem como pelas boas ideias que me foi transmitindo, e que, me permitiram atingir um resultado final mais rico.

Agradeço, ainda, aos colegas e amigos que conheci ao longo destes anos de carreira, os quais me permitem hoje, com satisfação, disponibilizar este trabalho. Posso agradecer em especial à CENTRIA, (Centro de Inteligência Artificial da UNL), especialmente ao Professor Doutor Nuno Cavalheiro Marques que contribuiu, inevitavelmente, pela minha evolução científica.

Na Portugal Telecom, ao Duarte Freitas e ao Luís Santos.

Na Escola Secundária de Francisco Franco onde se torna mais difícil agradecer a alguém em especial, pois toda a estrutura funciona como uma família, cuja entreeajuda e amizade é base para todo o resto. Contudo, deixo um agradecimento especial ao meu amigo Marco Olim e aos meus colegas do grupo de Informática.

A todas as empresas ou entidades que me proporcionaram ministrar formação, quer desenvolver projetos aliciantes e interessantes. A todos eles o meu muito obrigado, pois sai enriquecido dos mesmos.

Por fim, à minha família por sempre me ter acompanhado e ajudado ao longo destes anos de profissão.



Nº do aluno: 2086608

Nome: António Paulo Dias Pestana

Título da dissertação: *Dados, Informação, Conhecimento, Ensino de Programação, a Inteligência Empresarial e as suas motivações*

Palavras-Chave:

- Inteligência
- Text Mining
- Integração
- Etiquetas
- Redes Neurais
- Suporte técnico
- Ensino de Programação

Keywords:

- Intelligence
- Text Mining
- Integration
- Labels
- Neural Networks
- Technical support
- Programming Education



## Resumo

---

O objetivo deste relatório é dar a conhecer um possível percurso de carreira para um aluno que, à entrada no mundo profissional, se interessou por sistemas de suporte à decisão e mais tarde enveredou por uma carreira de docente. Descrevi a minha experiência profissional desde a entrada no curso de Engenharia Informática e as opções que fui tomando durante e depois do curso, demonstrativas do interesse e tendências para a área de suporte à decisão dentro das *TI*, *help desk*, assim como pela área do ensino.

Assim, o facto de ter trabalhado em diversas áreas e em diversas entidades, colaborei destacadamente na CENTRIA, Portugal Telecom (*System Care*) e Escola Secundária de Francisco Franco. Nestas entidades fiz desenvolvimento aplicacional em *Text Mining*, na definição de requisitos, na qualidade e integração de dados e na transmissão de ensinamentos. Esta multiplicidade de contextos permitiu a minha evolução profissional e humana.

O curso na FCT-UNL capacitou-me para ser tolerante à frustração, devido aos inúmeros obstáculos com que me fui deparando ao longo do curso, transmitindo-me, assim, uma capacidade de adaptação ao nível das mais diversas tecnologias e metodologias. Ao longo da minha carreira, e graças a todos os ensinamentos assimilados, tenho conseguido ultrapassar sempre as tarefas difíceis a nível técnico, funcional e de gestão que me foram surgindo.

Espero transmitir claramente como funcionam os meus projetos, as suas componentes, dificuldades e particularidades.

---

## Summary

---

The purpose of this report is to describe a possible career path of a student who, after finishing university and entering the professional world, became interested in Decision Support Systems and later on embarked on a teaching career. I've described my professional experience, starting from my Computer Engineering degree and going through the choices I made along the way. I believe those choices demonstrate my personal interest in the Decision Support area within IT, as well as Help Desk and teaching.

After getting the Computer Engineering degree, I worked in several areas and collaborated with a number of Entities, of which I should point out CENTRIA, Portugal Telecom (System Care) and Escola Secundária Francisco Franco. This collaboration extended from application development in Text Mining, to quality and requirement definition, as well as data integration and teaching. This multidisciplinary experience has made me grow, not only professionally, but also as a human being.

As a result of the number of challenges and obstacles I encountered throughout the University course, I became more adaptable and resilient, and those characteristics have allowed me to successfully work with several technologies and methodologies. I strongly believe that, thanks to the teachings and experiences I've accumulated throughout my career, I've been able to fulfil and materialize the necessary tasks for my progression, not only technically, but also from a functional and management point of view.

I hope this report clearly conveys how my projects work and illustrates all the components, specificities and difficulties I've encountered along the way.

---



“Onde há uma empresa de Sucesso,  
alguém tomou alguma vez uma decisão  
Valente”

(Peter Drucker)

“O mais importante ingrediente na  
fórmula do sucesso é saber como lidar  
com as Pessoas”

(Theodore Roosevelt)

"Feliz aquele que transfere o que sabe e  
aprende o que ensina"

(Cora Coralina)



# Índice

---

<b>Resumo</b> .....	<b>9</b>
<b>Summary</b> .....	<b>10</b>
<b>Índice</b> .....	<b>14</b>
<b>Índice de Imagens</b> .....	<b>17</b>
<b>Índice de Fotografias</b> .....	<b>18</b>
<b>1. Atividade Profissional</b> .....	<b>19</b>
1.1. CENTRIA .....	22
1.2. System Care .....	22
1.3. Escola Secundária de Francisco Franco.....	23
1.4. Outra experiência profissional .....	23
<b>2. Projetos mais relevantes</b> .....	<b>25</b>
2.1. Extrator de informação para anúncios de conferências (“Call for Papers”) .	25
2.1.1. Contexto .....	25
2.1.2. Enquadramento Teórico .....	25
2.1.2.1. Rede Neuronal FeedForward.....	26
2.1.2.2. Modelo de Trigramas.....	27
2.1.2.3. Método de Cross - Validation.....	27
2.1.3. Requisitos e Motivação .....	27
2.1.4. Pesquisa prévia / Análise do problema .....	28
2.1.5. Solução Técnica e Execução do Projeto .....	28
2.1.5.1. Base de Dados .....	28
2.1.5.1.1. Análise da Estrutura das Conferências .....	29
2.1.5.1.2. Conceção da Base de Dados.....	30
2.1.5.1.3. Descrição da Base de Dados .....	31
2.1.5.1.3.1. Diagrama de Tabelas .....	31
2.1.5.2. Etiketador.....	32
2.1.5.2.1. Especificação.....	32
2.1.5.2.2. Descrição de Scripts .....	33

2.1.5.2.3.	Definição de Etiquetas.....	35
2.1.5.2.4.	Dicionário Interno de Palavras .....	37
2.1.5.2.5.	Rede Neuronal.....	37
2.1.5.2.6.	Treino e Testes .....	38
2.1.5.2.6.1.	Metodologia .....	40
2.1.5.3.	Parser de Dados de Conferências .....	43
2.1.5.4.	Interface Web .....	43
2.1.5.4.1.	Descrição da Interface.....	43
2.1.5.4.2.	Implementação .....	48
2.1.6.	Análise crítica dos resultados.....	48
2.1.7.	Contribuições / Responsabilidades no Projeto .....	50
2.2.	Apoio de Microinformática a clientes PT .....	50
2.2.1.	Contexto .....	50
2.2.2.	Requisitos e Motivação .....	51
2.2.3.	Pesquisa prévia / Análise do problema .....	51
2.2.4.	Solução Técnica e Execução do Projeto .....	51
2.2.4.1.	Aplicação Genesis .....	51
2.2.4.2.	Aplicação GSU .....	52
2.2.4.3.	Aplicação RADIUS .....	52
2.2.4.4.	Despiste técnico.....	53
2.2.4.4.1.	Aplicação Assistente virtual NetCabo.....	53
2.2.5.	Análise Crítica dos Resultados.....	54
2.2.6.	Contribuições / Responsabilidades no Projeto .....	56
2.3.	Montagem e Programação de Robôs .....	56
2.3.1.	Enquadramento Teórico .....	56
2.3.1.1.	Teoria do Construtivismo .....	56
2.3.1.2.	Teoria do Construcionismo .....	57
2.3.2.	Contexto .....	57
2.3.3.	Requisitos e Motivação .....	58
2.3.4.	Pesquisa prévia / Análise do problema .....	59
2.3.5.	Solução Técnica e Execução do Projeto .....	60
2.3.5.1.	Descrição do Kit Lego® Mindstorms™ .....	60

2.3.5.1.1.	Constituição e Funcionamento .....	60
2.3.5.1.1.1.	O RCX.....	60
2.3.5.1.1.2.	Os Motores .....	61
2.3.5.1.1.3.	Os Sensores .....	61
2.3.5.1.1.3.1.	Sensores de Toque.....	62
2.3.5.1.1.3.2.	Sensores de Luz.....	62
2.3.5.1.1.4.	Torre de Transmissão .....	62
2.3.5.1.1.5.	O Firmware .....	63
2.3.5.1.1.5.1.	Lego.....	63
2.3.5.1.1.5.2.	BrickOS.....	63
2.3.5.1.2.	Ferramentas de Programação .....	64
2.3.5.1.2.1.	Robotics Invention System.....	64
2.3.5.1.2.2.	RCX Command Center.....	64
2.3.5.1.2.3.	Bricx Command Center.....	65
2.3.5.1.3.	Linguagens de Programação .....	65
2.3.5.1.3.1.	Linguagem de Programação NQC.....	65
2.3.5.1.3.1.1.	Casos de Estudo.....	66
2.3.5.1.3.1.1.1.	Problema 1.....	66
2.3.5.1.3.1.1.2.	Problema 2.....	68
2.3.5.1.3.1.1.3.	Problema 3.....	69
2.3.5.1.3.1.1.4.	Problema 4.....	70
2.3.5.1.3.1.1.5.	Problema 5.....	71
2.3.5.1.3.1.1.6.	Problema 6.....	72
2.3.5.1.3.1.1.7.	Problema 7.....	73
2.3.6.	Análise Crítica dos Resultados.....	74
2.3.7.	Contribuições / Responsabilidades no Projeto .....	75
	<b>Projetos em Desenvolvimento .....</b>	<b>78</b>
	<b>Conclusão.....</b>	<b>80</b>
	<b>Bibliografia .....</b>	<b>82</b>

## Índice de Imagens

---

Figura 1 - Cadeiras efetuadas durante a licenciatura da secção de SSDI .....	20
Figura 2 - Percurso Profissional .....	22
Figura 3 - Exemplo de uma Pequena Rede Neuronal .....	26
Figura 4 - Modelo ER .....	31
Figura 5 – Exemplo de Classificação de Texto utilizando Etiquetas.....	36
Figura 6 - Rede Neuronal utilizada durante o Treino .....	38
Figura 7 - Exemplo de um gráfico típico das taxas de erro do conjunto de treino e do conjunto de validação durante um treino .....	39
Figura 8 - Configuração de treino .....	42
Figura 9 - Gráfico obtido durante o treino .....	42
Figura 10 - Interface Inicial .....	44
Figura 11 - Interface depois de efetuado o Login .....	44
Figura 12 - Interface com um Exemplo de uma consulta de eventos .....	45
Figura 13 - Exemplo de uma Interface com detalhes dessa conferência .....	46
Figura 14 - Interface com um Exemplo de uma operação sobre uma conferência .....	47
Figura 15 - Exemplo de uma Interface com a opção editar conferência .....	47
Figura 16 – Aplicação Genesis .....	52
Figura 17 – Aplicação Radius.....	53
Figura 18 - Robotics Invention System 2.0 .....	60
Figura 19 - O RCX da Lego MindStorms.....	61
Figura 20 - Motor ligado a saída do RCX.....	61
Figura 21 - Sensor de Toque.....	62
Figura 22 - Sensor de Luz.....	62
Figura 23 - Comunicação entre o computador e o RCX.....	63
Figura 24 – Torre de Infravermelhos .....	63
Figura 25 - Ambiente de programação do RIS .....	64
Figura 26 - Ambiente de programação do RcxCC.....	64
Figura 27 - Ambiente de programação do BricxCC .....	65
Figura 28 - Estrutura de uma tarefa na linguagem NQC .....	65
Figura 29 - Portal Web Good Luck Tours .....	79

## **Índice de Fotografias**

---

Fotografia 1 - Alunos do grupo I .....	76
Fotografia 2 - Alunos do grupo II .....	76
Fotografia 3 - Alunos na fase de programação do Robô “Todo o Terreno” .....	76
Fotografia 4 - Alunos na fase em que passavam via sensor para o Robô “Todo o Terreno”, o programa por eles feito .....	77

## 1. Atividade Profissional

Iniciei o meu percurso no mundo laboral e na área da Engenharia Informática ainda durante a faculdade, tendo na altura tentado aproveitar todos os conhecimentos que já trazia e os que fui adquirindo numa perspetiva de utilizador. Ainda assim, este início não teve qualquer relação com aquela que acabou por vir a ser a minha grande experiência na área das *TI*, o *Business Intelligence* e posteriormente com o ensino.

Da minha aprendizagem e investigação sobre o processo de obtenção de informação de qualidade a partir de texto, foi possível verificar que, de facto, o *Text Mining* melhora o *Business Intelligence*, porque os processos de *Text Mining* abrem as portas à utilização de novas fontes de conhecimento. Com a adoção destes processos, torna-se viável utilizar fontes de informação externa à organização, a maior parte dela gratuita ou com baixos custos de obtenção, para alimentação de processos de inteligência competitiva e *marketing intelligence*.

O *Text Mining* é considerado uma especialização dos processos de *data mining* confinada à informação textual, na medida em que esta metodologia visa igualmente a descoberta de conhecimento “escondido” nas bases de dados. Um dos principais, senão mesmo o principal fator de diferenciação entre *data mining* e *texto mining*, é a tipologia da informação utilizada para alimentar os respetivos processos de análise.

No caso do *text mining* a fonte de dados é o texto na forma de expressão natural. O interesse crescente por processos de *text mining* também se justifica pela possibilidade de aproveitamento da informação existente nas bases de dados corporativas. Efetivamente, através destes processos torna-se viável utilizar informação que foi produzida pelos diversos sistemas das empresas com intuítos meramente operacionais e sobre os quais não era expectável, à priori, ser usada para outros fins, para extrair conhecimento e, por conseguinte, gerar valor para a organização.

Considero importante mencionar que, de alguma forma, existiu uma influência clara nas opções que efetuei durante a licenciatura na escolha do meu percurso. Relativamente às

disciplinas opcionais que então elegi, demonstrei apetência para a secção central que refiro na figura 1, a SSDI, (Sistemas Simbólicos, de Decisão e de Informação).

A figura 1 dá a conhecer as cadeiras que efetuei da secção de SSDI. Nos hexágonos azuis estão referidas as que eram obrigatórias no plano curricular, nos hexágonos verdes as que escolhi como opcionais. O estágio final está num hexágono de cor castanha, pois embora seja obrigatório dentro do plano curricular, o seu conteúdo acaba por ser opcional.



**Figura 1 - Cadeiras efetuadas durante a licenciatura da secção de SSDI  
(Sistemas Simbólicos, de Decisão e de Informação)**

Seguidamente, apresento um pequeno resumo do meu percurso dentro da Engenharia Informática.

Iniciei a minha atividade, ainda durante o curso, no apoio a várias empresas e particulares presentes em vários setores de atividades, realizando as seguintes tarefas, entre outras:

- Montagem de máquinas e storage;
- Instalação e configuração de sistemas operativos MS-DOS, Microsoft Windows e Linux;
- Instalação e configuração de inúmero *software* necessário na área de atividade de cada cliente desde SW de restauração, contabilidade e arquitetura;
- Desenho de redes e instalação das mesmas, desde a passagem de cablagem entre divisões; acoplagem de fichas *Ethernet* RJ45. Instalação de *Switch* e *Routers* e sua posterior configuração;
- Gestão de sistemas distribuídos de *storage* e impressão em redes Microsoft e Novel;
- Reparação de máquinas.

Este processo prévio ao término do curso de Engenharia Informática deu-me uma noção muito importante do funcionamento das máquinas e reconhecimento dos problemas de instabilidade no seu comportamento, do cuidado a ter com os equipamentos e com os quais é efetivamente necessário ser previdente e assim evitar problemas diretamente associados ao *HW*. Para além disso, conheci vários SO e também a forma de interagir com o seu *core*, o que me ajuda ainda hoje na forma como abordo um sistema que me é desconhecido.

Durante o ano de 2004, quando estava na fase final do curso na FCT-UNL, procurei fazer o estágio numa instituição da área das *TI*. Isto deveu-se ao facto do meu objetivo ter sido sempre trabalhar no meio empresarial e poder ter acesso a inúmeros tipos de atividades e negócios e aprender um pouco sobre cada um deles. Assim, quando surgiu como hipótese na lista de estágios propostos, “entrar” na CENTRIA, um centro de investigação de renome, resolvi propor-me para o mesmo.

Após uma entrevista e analisadas as candidaturas, fui aceite e iniciei assim o meu Estágio. O mesmo foi fundamental para perceber como poderia intervir concretamente a nível do negócio de uma grande organização.

Após cumprir o estágio final de curso, fui convidado a manter-me em funções na CENTRIA, onde iniciei o desenvolvimento do projeto “*call for papers*”. Daqui em diante consegui dividir a minha carreira em 3 fases, que coincidiram com os momentos de transição para novas entidades empregadoras, dando origem a um novo conjunto de desafios e responsabilidades.

Assim, o meu percurso passou pelas seguintes fases:



Figura 2 - Percurso Profissional

### 1.1. CENTRIA

Na CENTRIA assumi desde o início o papel de *Analist & Developer*. Isto permitiu-me, devido à complexidade e versatilidade exigidas, desenvolver fortemente a componente técnica nas áreas de Modelação de dados, desenvolvimento de processos ETL, ou seja, Etiquetagem, extração e carga de dados para padronização dos mesmos.

A experiência nesta área permite uma facilidade na compreensão dos problemas e requisitos apresentados, assim como a necessidade de se obterem soluções de qualidade, sempre com pensamento na evolução dos sistemas.

### 1.2. System Care

Após a saída da CENTRIA seguiu, em 2006, uma experiência na área do “*Help desk*”. Nessa altura, a Portugal Telecom funda a APCC – Associação Portuguesa de Contact Centers, criando posteriormente a marca System Care. O suporte técnico System Care consistia em apoio de Micro-informática a clientes PT. O apoio aos clientes era efetuado, nomeadamente, através do acesso remoto ao PC do cliente. Esta ferramenta era exclusivamente dedicada a apoio ao cliente na área da micro-informática e era prestada por operadores da *System Care*.

Objetivos:

- Garantia de suporte a questões de microinformática que, não estando diretamente relacionadas com o serviço Internet, podem ser relevantes para a qualidade do serviço prestado.
- Complemento ao Suporte TMN, Sapo, Telepac e Netcabo.

### **1.3. Escola Secundária de Francisco Franco**

Em 2007, ingressei no ensino, assumi as funções de Professor na escola Secundária de Francisco Franco, na qual me mantenho em funções até hoje. Durante os anos de 2009 e 2010 ingressei na Universidade da Madeira e realizei a Profissionalização em Serviço, no grupo 550, de Informática.

O meu contributo focou-se não só no ensino das diversas disciplinas de informática, mas também em impulsionar o núcleo de Informática. Um dos objetivos que me foi proposto fazer foi criar, junto dos alunos, uma dinâmica de análise, compreensão e resolução de problemas, utilizando robots e analisar a atividade informática dos alunos aquando da utilização dos mesmos.

Para além da função lúdica, os robots são utilizados como função didática, pois aborda a teoria e a prática de conteúdos direcionados para a introdução à programação, permitindo o desenvolvimento de projetos de pequeno e médio porte, estimulando a criatividade e a solução de problemas do quotidiano por parte dos alunos.

### **1.4. Outra experiência profissional**

Ao longo destes anos prestei apoio microinformática, elaboração de portais web e formação nas seguintes empresas:

- Magna Voce - Formação e Consultadoria, Lda.
- ForMar XXI, Formação e Consultoria, Lda.
- Qi Forma - Formação Profissional, Lda.
- Good LuckTours – Viagens, Lda.

- Qualificar F.P. – Formação Profissional, Educação e Serviços
- Escola A.S.A.S. - Escola Profissional de Agentes de Serviço e Apoio Social.
- Netcriações – Publicidade e Internet, Lda.

## **2. Projetos mais relevantes**

### **2.1. Extrator de informação para anúncios de conferências (“Call for Papers”)**

#### **2.1.1. Contexto**

O *TextMining* trata da extração de conhecimento partindo de documentos. O excesso de informação nesses documentos pode tornar-se, dependendo das situações, pouco vantajosas. Por exemplo, quando se pretende detetar e selecionar informações específicas nos textos.

No âmbito deste projeto, pretende-se generalizar o sistema de forma a suportar a definição de diversos tipos de filtros para distintos domínios de aplicação, nomeadamente, a integração de um módulo para obter informação a partir de e-mails. O objetivo específico é a configuração do sistema para tratamento de e-mails, contendo chamadas de trabalhos para conferências (*call for papers*). Estas possuem um conjunto de dados interessantes, tais como o nome da conferência, as datas de realização, os tópicos da conferência assim como o país, a cidade e o local da sua realização.

#### **2.1.2. Enquadramento Teórico**

Para a definição de filtros é utilizado um etiquetador que baseia-se numa rede neuronal. Convém, portanto, explicar de uma forma breve o que são redes neuronais. As redes neuronais artificiais são modelos computacionais inspirados nas redes neurológicas biológicas que constituem o cérebro humano. Uma das principais características das redes neuronais artificiais é a possibilidade de serem ajustadas de forma a ficarem treinadas para resolver um determinado problema.

As redes neuronais são compostas por unidades neuronais, que apresentam um modelo simplificado do comportamento dos neurónios do cérebro humano. As unidades neuronais podem ser de vários tipos: entrada, saída ou escondida. Tal como os neurónios, as unidades neuronais ligam-se através de conexões. As conexões neste caso são normalmente unidirecionais e têm um peso associado. Cada unidade possui uma função

de ativação que especifica a forma como as ligações de entrada numa unidade são avaliadas.

O funcionamento base de uma rede neuronal pode resumir-se a uma sucessão de troca de valores entre as unidades neuronais através das suas conexões. As unidades de entrada recebem valores lidos diretamente de ficheiros de padrões. Caso existam unidades escondidas numa camada intermédia os valores de saída das unidades de entrada são enviados para as respetivas unidades da camada intermédia. Depois das unidades da camada intermédia processarem os valores, estes são enviados para as unidades de saída. As unidades de saída processam os valores e enviam o resultado para um ficheiro de *output*.

### 2.1.2.1. Rede Neuronal FeedForward

Diferentes topologias de redes neuronais podem ser consideradas, sempre com o objetivo de uma otimização da taxa de erro. Pode-se ter uma rede neuronal simples *feedforward*, que possua apenas duas camadas de neurónios: entrada e saída. Numa rede desse género todos os neurónios de entrada conectados aos neurónios de saída. Pode-se ter redes mais complexas que introduzem novas camadas de neurónios, denominadas de camadas intermédias. Nesse género de redes todos os neurónios de entrada estão ligados aos da camada intermédia e estes por sua vez estão ligados a todos os de saída. Em seguida, temos um pequeno exemplo.

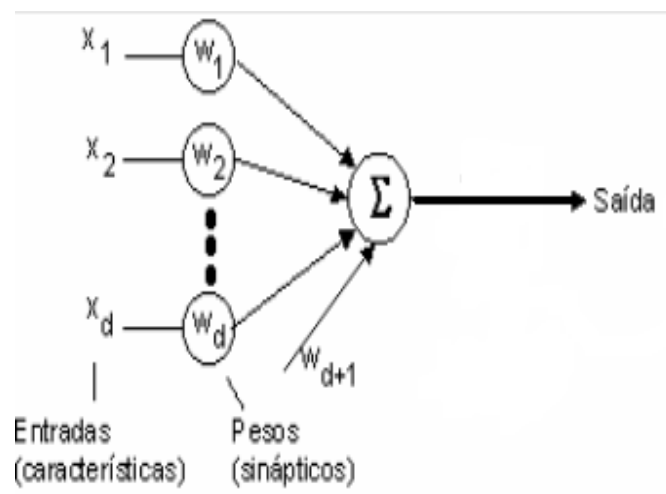


Figura 3 - Exemplo de uma Pequena Rede Neuronal

### **2.1.2.2. Modelo de Trigramas**

Neste projeto o etiquetador a utilizar deve ser capaz de classificar as palavras segundo o contexto, para isso é necessário definir um modelo de contexto. O modelo que vai ser utilizado é o modelo de trigramas. Este considera que o contexto de uma palavra é constituído pelo vizinho anterior e posterior da palavra em questão: **[palavra i-1, palavra i, palavra i+1]**

### **2.1.2.3. Método de Cross - Validation**

O método de *Cross – Validation* é uma técnica utilizada para avaliar a capacidade de generalização de um modelo, a partir de um conjunto de dados. Esta técnica é amplamente utilizada nos problemas onde o objetivo de modelagem é estimar o seu desempenho para um novo conjunto de dados.

O conceito base das técnicas de validação cruzada é o particionamento do conjunto de dados em subconjuntos exclusivos, e posteriormente, utilizar alguns destes subconjuntos para a estimação dos parâmetros do modelo (dados de treino) e o restante dos subconjuntos (dados de validação) são utilizados para a validação do modelo.

### **2.1.3. Requisitos e Motivação**

O objetivo principal deste projeto visa desenvolver um sistema de tratamento de e-mails, contendo anúncios de conferências, normalmente designados por “*call for papers*”. Uma vez identificados e extraídos os dados mais importantes das conferências, passamos à tarefa seguinte, ou seja, guardar essa informação numa base de dados, cujo produto final é disponibilizar a informação da base de dados numa *interface web*, onde se pode não só consultar como também alterar os seus dados. Assim sendo uma base de dados que contenha este género de informação pode ser muito valiosa. Apesar de encontramos nos e-mails, e de forma bastante acessível, informações relativas às conferências, não quer dizer que a sua extração automática, a partir de um e-mail, seja considerada tarefa fácil, daí a motivação e expectativas para este projeto serem elevadas.

#### **2.1.4. Pesquisa prévia / Análise do problema**

Efetuei um levantamento dos requisitos necessários, quer para a base de dados, quer para a interface do sistema. No caso da base de dados, o levantamento de requisitos efetuado foi o necessário para se idealizar o *layout* da interface do sistema e os campos da base de dados.

Depois de apresentada e discutida uma solução sobre a base de dados e a interface para o sistema, passei à sua implementação. De referir que, para a implementação da base de dados, utilizei o *PostgresSQL* na sua versão 4.0 com chaves externas, e para a interface do sistema utilizei *Php* na sua versão 4 para linguagem de interpretação e para verificação de dados para inserção e para obtenção de valores como, por exemplo, validar o preenchimento dos campos de texto.

Na fase das redes neuronais e uma vez que os conhecimentos sobre as mesmas eram nulos, despendi algum tempo a pesquisar informação sobre o assunto e experimentar soluções encontradas.

Finalmente, na implementação do Sistema de receção de texto e sua inserção na base de dados utilizei o *Php* na sua versão 4 para a implementação do mesmo. Esta fase não foi implementada na totalidade.

#### **2.1.5. Solução Técnica e Execução do Projeto**

##### **2.1.5.1. Base de Dados**

As bases de dados relacionais são, geralmente, o suporte mais utilizado para guardar e relacionar grandes quantidades de informação. No âmbito deste projeto, a grande quantidade de informação que se pretende armazenar, consultar, inserir, corrigir diz respeito a e-mails com anúncios de conferências. Antes de desenhar esta nova base de dados procedi à realização de um estudo sobre a estrutura das conferências. Este estudo visa observar as várias componentes que fazem parte da estrutura de uma conferência e decidir quais as que devem ser utilizadas como campos nas diversas tabelas que constituirão a nova base de dados. Digamos que este estudo serviu de “guia” para o desenho e conceção da base de dados.

### 2.1.5.1.1. Análise da Estrutura das Conferências

Geralmente, uma conferência apresenta diversas componentes, das quais algumas são obrigatórias e outras opcionais. As componentes obrigatórias dizem respeito ao nome da conferência, à data da sua realização, à cidade e ao país onde decorre. As componentes opcionais estão relacionadas, por exemplo, com a identificação do alojamento, com os membros da conferência, com possíveis datas associadas à conferência, etc. Para uma conferência ser considerada válida deve ter todas as componentes obrigatórias atrás mencionadas. São as componentes opcionais que originam maior dificuldade e ambiguidade no processo de deteção de conferências. Aqui, surge não só a dificuldade em saber qual o formato do identificador, como também a dificuldade em descobrir o tipo de componentes correspondem os dados de conferências semelhantes. Vejamos, por exemplo, os seguintes dados de conferências:

“João Gama, University of Lisboa, Portugal”

“Cholula 1996 (México), Lisboa 1998 (Portugal) ”

A palavra “Lisboa” representa efetivamente entidades distintas numa conferência, ou seja, pode representar o nome de uma cidade ou o nome de uma instituição. As ambiguidades na apresentação de uma conferência acontecem essencialmente a nível de identificadores. Isto deve-se ao facto de não existirem regras muito rígidas para a sua apresentação. Para melhor compreender as variantes possíveis para a estrutura de uma conferência convém analisar as suas componentes básicas:

- *Datas*

Especifica o tipo de data. Aparece associado a uma conferência, tais como data de início/fim de conferência, data limite de submissão de “*papers*”, data de notificação de aceitação, etc.

Exemplo:

“Submission deadline: June 14, 2004”

Notification of acceptance: July 5, 2004

Camera ready copies due: July 12, 2004”

- *Evento*

Especifica o nome do evento. Esse evento aparece associado à edição de conferência. Este não é mais do que o nome da conferência.

Exemplo:

“The 15th European Conference on Machine Learning (ECML)

The 8th European Conference on Principles and Practice of Knowledge Discovery”

- *Edição Conferência*

Especifica o número da edição de conferência. Essa edição de conferência aparece associada ao evento. A edição de conferência refere a quantidade de vezes que se realiza um determinado evento. Evento esse que pode realizar-se uma única vez, ou ser anual, bianual, trimestral, mensal, etc.

Exemplo:

“The 15th European Conference on Machine Learning (ECML)

The 8th European Conference on Principles and Practice of Knowledge Discovery”

#### **2.1.5.1.2. Conceção da Base de Dados**

Como referi anteriormente, a conceção da base de dados é o resultado da análise pormenorizada das características das conferências. Nessa altura decidi qual o servidor de base de dados que deveria ser utilizado para o repositório dos dados das conferências retiradas. O servidor escolhido foi *PostgresSQL*, este é compatível com o restante sistema e é de fácil e simples aplicação.

A base de dados criada é do tipo relacional por isso deu-se importância às relações entre os vários componentes que constituem a estrutura da conferência. Após a análise desse estudo surge a necessidade de criar duas tabelas distintas, uma para eventos e outra para edição de conferência. Outra conclusão que se pode tirar foi a possibilidade de um mapeamento entre as componentes das conferências e os campos das respetivas tabelas.

### 2.1.5.1.3. Descrição da Base de Dados

Deste modo, apresentamos o diagrama de tabelas constituintes da base de dados de conferências.

#### 2.1.5.1.3.1. Diagrama de Tabelas

[1,1]

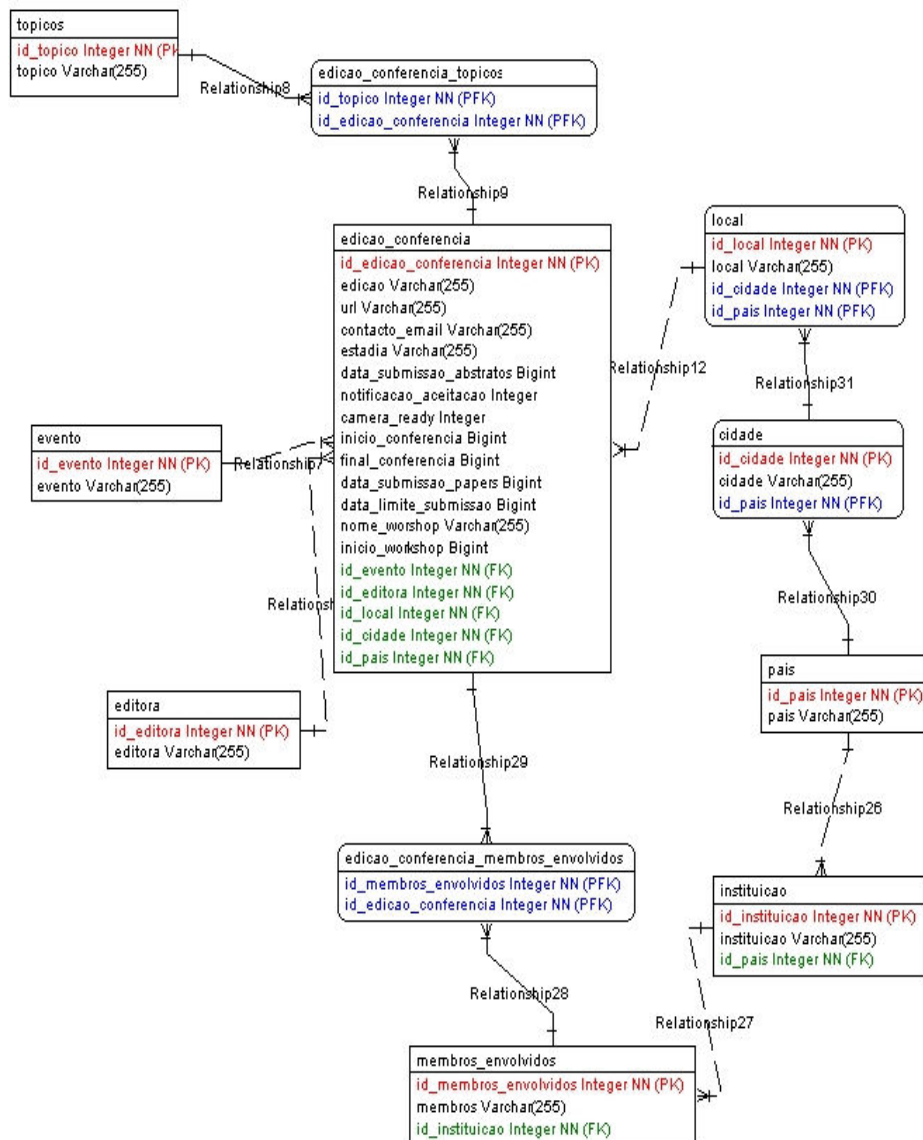


Figura 4 - Modelo ER

### 2.1.5.2. Etiquetador

Identificar a categoria sintática de cada palavra de um texto sem ambiguidades é a função de um etiquetador morfossintático. No entanto, se se associar cada palavra à sua possível categoria no dicionário lexical verifica-se que este dicionário lexical não é capaz de classificar com exatidão todo o tipo de palavras que podem eventualmente surgir num determinado texto. Isto acontece porque é o contexto onde as palavras estão inseridas que determina a categoria da palavra. Por exemplo, as palavras homónimas embora se escrevam da mesma forma podem pertencer a categorias gramaticais diferentes, dependendo do contexto em que estão inseridas. A palavra “the”, por exemplo, tanto pode ser classificada como artigo ou como preposição.

O modelo utilizado é o modelo de trigramas. Este considera que o contexto de uma palavra é constituído pelo vizinho anterior e posterior da palavra em questão: **[palavra i-1, palavra i, palavra i+1]**

Para melhor perceber o modelo de trigramas e a importância do contexto no processo de classificação das palavras, vejamos, por exemplo, a classificação da palavra “to” na seguinte frase: “*went to london*”. O etiquetador tem em conta os elementos “vizinhos”, ou seja, as palavras *went* e *london*. Melhor dizendo, o etiquetador considera o contexto que rodeia a palavra “to”, pois essas palavras vão determinar a classe a que pertence “to”. Deste modo, o etiquetador constata que a palavra “went” pode ser verbo e que “london” pode ser um nome próprio. De entre as possíveis classes para registar a palavra “to” o etiquetador escolherá a melhor, ou seja, aquela em que é precedida de verbo e sucedida por um nome próprio.

#### 2.1.5.2.1. Especificação

Neste projeto adaptou-se o etiquetador já utilizado sobre o corpus *susanne* [MarquesLopes2001], ou seja, o etiquetador que classifica texto em língua inglesa. O etiquetador original estava preparado para classificar as palavras segundo a gramática de língua inglesa através do dicionário interno de palavras onde estão armazenadas as informações das ocorrências dos vários tipos de palavras num texto. Foi necessário treinar a rede neuronal para esta conseguir classificar as palavras das conferências, ou seja, o dicionário foi construído de raiz, com base na marcação do corpus, e então depois adaptou-se o dicionário interno das palavras originais para a realidade das conferências.

O etiquetador utiliza dicionários lexicais a fim de guardar informação sobre as ocorrências relativas das palavras num texto. Estas ocorrências, mais tarde, transformar-se-ão em frequências relativas que serão apresentadas sob o formato de ficheiros de padrões.

Cada palavra do ficheiro de texto a classificar possui uma entrada no ficheiro de padrões. Cada entrada corresponde a um conjunto de frequências relativas associadas a cada possibilidade de classificação do elemento. Estas frequências relativas de classificação devem ser apresentadas de uma forma legível pela rede neuronal. Cada elemento de entrada e saída do ficheiro de padrões corresponde a um neurónio de entrada ou saída na rede neuronal, respetivamente.

Para que um etiquetador consiga classificar um determinado tipo de elemento, a sua rede neuronal deve ser treinada com ficheiros de padrões que contenham esses mesmos elementos. No caso deste projeto, a rede neuronal do etiquetador deve ter elementos de conferências. Assim que a rede neuronal estiver apta a detetar dados de conferências o etiquetador pode começar a marcação do texto.

O etiquetador tem como *input* o ficheiro de texto que pretende marcar, tendo este uma palavra por linha. O etiquetador transforma o ficheiro de texto num ficheiro de padrões, contendo as frequências relativas de cada palavra. Estas são calculadas com base nas ocorrências de palavras definidas pelo dicionário lexical. O ficheiro de padrões é passado para a rede neuronal que o processa e no final devolve uma combinação de valores nos seus neurónios de saída. Após a análise desses valores e determinada qual a etiqueta possível para cada palavra.

A chave do sistema de deteção de conferências é o etiquetador, pois se ele conseguir marcar corretamente todos os componentes de uma conferência, existirá então uma grande possibilidade dessa morada ser futuramente detetada com sucesso.

#### **2.1.5.2.2. Descrição de Scripts**

Os scripts UNIX são a base de implementação do etiquetador original de texto em língua Inglesa. Este tipo de programação também foi usado para adaptação do etiquetador deste projeto. Digamos que a ciência deste projeto constitui precisamente na adaptação do etiquetador original de modo a que conseguisse classificar os componentes das conferências.

Os scripts UNIX possuem um conjunto de utensílios muito importantes para fazer o tratamento e processamento de corpora de texto. Estes utensílios contêm pequenos programas baseados em filtros e *pipes*, que originam um processamento concorrential entre si. O *gawk*, *sed*, *sort*, *tr* e *uniq*, são exemplos destes programas. As ferramentas atrás referidas foram utilizadas na adaptação do etiquetador original a este projeto.

O etiquetador original possui muitos *scripts*, a maioria deles foi utilizado sem alterações.

O principal *script* do etiquetador é o *marcar.x*. Este *script* é responsável pelo desencadear dos restantes sub-scripts relativos à classificação das palavras de um texto. O objetivo do *script* *marcar.x* é etiquetar um corpus de texto contendo um *token* por linha.

```
#      stdin : corpus a marcar (1 palavra por linha).
#      stdout: Texto marcado
#      $1 : Dicionario a utilizar.
NTAGSin=`gawk '($0 ~ /^W*max_f\W*=/) {print $3}'
init_tags.awk`
NTAGSout=`gawk '($0 ~ /^W*max_f_out\W*=/) {print
$3}'init_tags.awk`
if(exit ${ $#==1}) then
    grep "^#" marcar.x |cut -b3-
    exit
fi
class.x $1 |
gawk '
BEGIN {print "<ini>"}
{print}
END{print "<end>"}'|
extract_f.x --words --noinfo |
ctrigram.x |
nntagger $NTAGSout $NTAGSin |
get_tags.x
```

### 2.1.5.2.3. Definição de Etiquetas

As etiquetas servem para classificar. Elas devem ser definidas na fase inicial da classificação e devem manter-se iguais até ao fim de todo o processo. Definir bem as etiquetas é fundamental para a resolução do problema. No caso deste projeto, o problema é a deteção de conferências. Deste modo, as etiquetas devem ser definidas de acordo com os componentes que constam da base de dados [MarquesGonçalves2004]. Assim, em função da informação que pretendemos inserir na base de dados, definiram-se 25 etiquetas:

- NC → Nome Cidade
- NP → Nome País
- NI → Nome Instituição
- NE → Nome Editora
- MES → Nome Mês
- DIA → Nome Dia
- ANO → Ano
- EC → Edição Conferência
- TOPIC → Tópico
- NM → Nome Membro
- LOCAL → Local Conferência
- NWORSP → Nome Workshop
- DATEABS → Data Submissão Abstratos
- DATEPASS → Data Submissão Papers
- DATENOT → Data Notificação de Aceitação
- DATESUB → Data Limite de Submissão
- DATECAR → Data CameraReady

- DATEWKS → Data de Início de Workshop
- DATEINC → Data Início de Conferência
- DATEFIC → Data Final de Conferência
- TC → Título Conferência
- PREP → Preposição
- CONJ → Conjunção
- ARD → Artigo Definido
- OTHER → Texto Irrelevante

Grande parte destas etiquetas corresponde a um mapeamento quase direto dos campos na Base de Dados. No entanto, há etiquetas que não estão associadas a nenhum campo em particular, mas são ainda assim úteis para o reconhecimento de expressões mais complexas. É o caso de etiquetas como *PREP*, *CONJ* ou *ARD*. Criou-se uma outra etiqueta chamada *OTHER*, que é utilizada para todas as palavras que não são relevantes para a informação que se pretende extrair do texto para a base de dados.

Veja-se a classificação da seguinte conferência para melhor assimilar a relação das etiquetas com as componentes das conferências:

```

9th_EC
European_TC
Conference_TC
on_PREP
Principles_TC
and_CONJ
Practice_TC
of_PREP
Knowledge_TC
Discovery_TC
in_PREP
Databases_TC
PKDD _TC

```

**Figura 5 – Exemplo de Classificação de Texto utilizando Etiquetas**

Analisando o exemplo verifica-se que a etiqueta TC identifica a conferência em questão. Já a etiqueta *CONJ* assinala a palavra “*and*”, o que significa, neste caso, a junção de dois componentes do título da conferência.

#### **2.1.5.2.4. Dicionário Interno de Palavras**

O etiquetador possui um dicionário interno de palavras, cujo objetivo é registrar as várias vezes que as palavras ocorrem num determinado texto. Esta informação não só é importante para o processo de classificação das palavras, mas também porque é a partir desta mesma informação, que se calcula as frequências relativas que determinam a etiqueta final de uma palavra. Através de cada linha do dicionário interno de palavras acedemos às ocorrências relativas de uma determinada palavra.

Podemos compreender melhor esta situação, através do exemplo seguinte:

**The \_ART:45 \_PREP:17 \_NP:4 200**

A linha apresentada resume as frequências observadas num texto de treino. Em 200 observações a palavra “*The*” foi classificada como artigo em 45 casos, como preposição em 17 casos e como nome próprio somente em quatro casos. Este tipo de informação é posteriormente utilizado sob a forma de frequência relativa em ficheiros de padrões que são lidos pela rede neuronal associada ao etiquetador.

Neste projeto, foi necessário fazer um estudo sobre os possíveis conjuntos de palavras que melhor representariam quer qualitativa, quer quantitativamente as palavras que ocorrem nas conferências. Só depois de concluído este estudo, deu-se um novo passo, a construção do novo dicionário de palavras.

#### **2.1.5.2.5. Rede Neuronal**

Para a deteção de conferências foi criada uma rede *feed-forward* com uma camada de entrada, com duas camadas intermédias e outra de saída. A camada de entrada vai ler valores diretamente aos ficheiros de padrões, pelo que tem de estar corretamente calibrada segundo o número de entradas de cada padrão. Como existem 25 etiquetas disponíveis para classificação e foi usado um modelo de contexto baseado em trigramas, isto é, com uma vizinhança de 3 elementos, então

cada padrão tem sempre 75 elementos de entrada. Foi por isso criada uma camada de entrada na rede com 75 unidades neuronais. Existe uma camada intermédia, com duas camadas ocultas. Uma primeira camada oculta com 50 unidades neuronais e uma segunda camada oculta com 15 unidades neuronais. A camada de saída deve refletir as possibilidades de classificação de um elemento. Como foram definidas 25 etiquetas para classificação, então foi criada uma camada de saída com 25 unidades neuronais

Para a criação da rede neuronal referida anteriormente foi usado o *Stuttgart Neural Network Simulator* [SNNS]. A rede neuronal pode ser descrita pela *figura* abaixo:

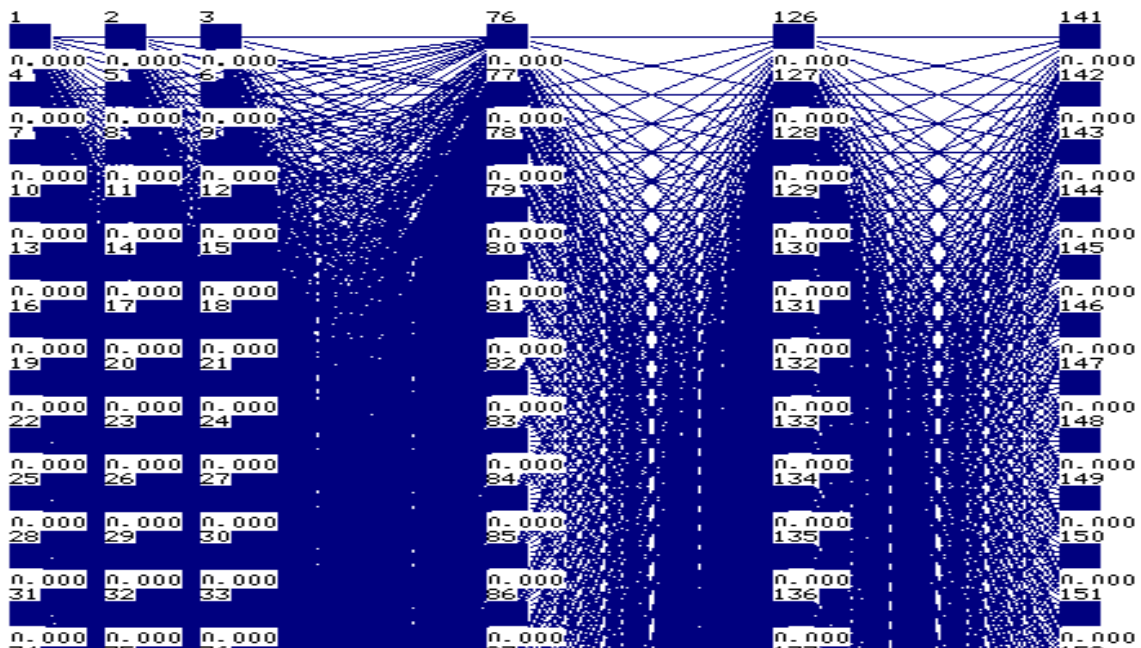


Figura 6 - Rede Neuronal utilizada durante o Treino

#### 2.1.5.2.6. Treino e Testes

A rede neuronal utilizada pelo etiquetador tem de ser treinada, caso contrário a classificação das conferências feita pelo etiquetador não será possível, pois os parâmetros (pesos) do modelo neuronal ainda não foram definidos. Deste treino depende portanto o sucesso do sistema e o sucesso da classificação das palavras das conferências. Uma vez efetuado o treino da rede neuronal segue-se o seu teste. Há assim 2 tipos de conjuntos, o de treino e o de validação.

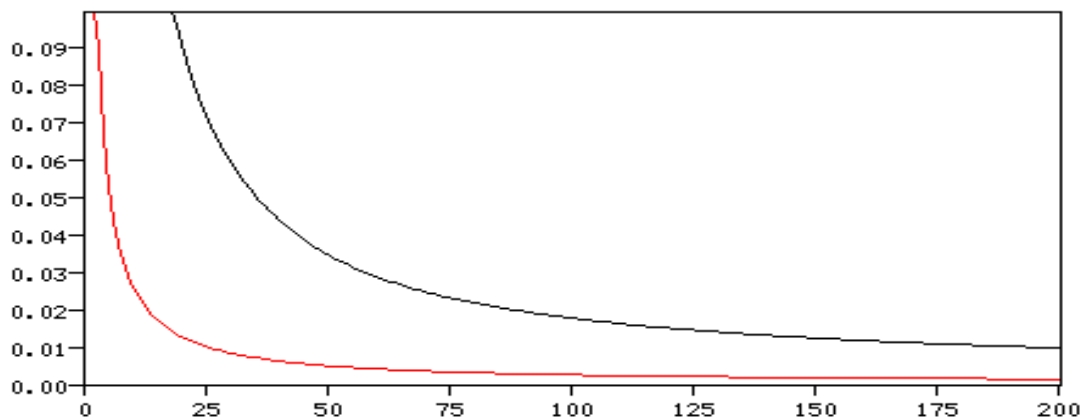
O primeiro (o conjunto de teste) deve conter elementos representativos do problema que se quer classificar, nesta situação é a utilização das várias etiquetas para classificação das várias palavras

num anúncio de conferência. Este treino deve conter pelo menos um número razoável de elementos representativos das várias etiquetas utilizadas.

O conjunto de validação é uma espécie de complemento do conjunto de treino durante a fase de treino do etiquetador. Os elementos utilizados no conjunto de validação não são os que foram utilizados no conjunto de treino do etiquetador. O conjunto de validação tem como objetivo avaliar o número de iterações ótimas que a rede neuronal esteve sujeita durante o treino.

Durante o treino, o etiquetador aprende a melhor classificar os elementos e a probabilidade de erro torna-se mais baixa. Numa determinada altura do treino, verifica-se que o fator de aprendizagem estagnou. Quando isso acontece, quer dizer que o número de iterações corresponde a uma aprendizagem ótima. Isto nem sempre é assim dado que também podem ocorrer mínimos locais.

A determinação desse número de iterações foi feita através da análise do gráfico do conjunto de validação. A figura abaixo representada mostra os gráficos típicos das taxas de erro do conjunto de treino (linha preta) e do conjunto de validação (linha vermelha) durante um treino [SNNS].



**Figura 7 - Exemplo de um gráfico típico das taxas de erro do conjunto de treino e do conjunto de validação durante um treino**

Uma vez treinada a rede neuronal, resta determinar a sua precisão, mas a nível de classificação. Ou seja, determinar a sua taxa de erro efetiva. Para analisar a taxa de erro do etiquetador realizaram-se novos treinos chamados de conjunto de testes.

Este conjunto de testes servem basicamente para realçar as diferenças entre o conjunto de teste e o conjunto de teste marcado pelo etiquetador. O facto de um etiquetador hipoteticamente poder

vir a apresentar um erro mais baixo pode não significar que este é o melhor, pois um erro mais baixo poderá dizer que o etiquetador está demasiado especializado nos dados das conferências do conjunto de teste e não na generalidade dos dados das conferências.

#### 2.1.5.2.6.1. Metodologia

O método de Cross – *Validation* define que em cada treino é omitido um conjunto de casos, sendo o etiquetador treinado com os restantes. Tendo em conta esta definição escolheu-se este algoritmo para o treino e validação do etiquetador. No entanto, o etiquetador posteriormente também vai ser validado com os casos que foram deixados de lado.

Geralmente o corpus de texto original é dividido em 10 subconjuntos, deixando um de fora e treinando com os restantes 9 em cada fase. Isto significa que haverá 10 combinações possíveis de treino e teste. Para as fases de treino e teste, esses 10 subconjuntos são agrupados em conjuntos de treino, validação e teste. O conjunto de treino, como o nome indica, é o escolhido para treinar a rede neuronal do etiquetador. A diferença entre conjunto de validação e de teste é que o primeiro é usado para determinação do ponto ideal de aprendizagem e o segundo para a determinação da taxa de erro final do etiquetador.

Para a realização das fases de treino e teste foi usado um corpus de conferências marcadas, contendo 10 *emails* de conferências. Esse corpus marcado foi depois dividido em conjunto de teste, treino e validação.

Depois de analisado o corpus marcado, verificou-se que era importante a remoção de algumas etiquetas *OTHER*. Constatou-se que a percentagem de acerto geral do *tagger* era de 80%.

Sendo assim foi preciso efetuar a normalização do corpus. Retirou-se aleatoriamente as etiquetas *OTHER* com a seguinte instrução:

```
cat valid.txt |tr -d '\r'|gawk '($NF == "_OTHER")
{if(rand(>0.8) print $0 ; next} {print}' > valid2.txt
```

Para a versão que tem em conta os trigramas. O Ficheiro corpus\_rem\_norm.txt deve ser utilizado para construir o dicionário.

```
$ cat corpus_cfpF1.txt|./trigram.x |gawk '{s=$2 $4 $6}
(s=="_OTHER_OTHER_OTHER" ) {if(rand())>0.80) { print $0} else
{print $0 >> "tmp.txt"} ; next} (s=="_TOPIC_TOPIC_TOPIC")
{if(rand())>0.5) {print $0} else {print $0 >> "tmp.txt"} ;
next} {print}'|gawk '{print $1 " " $2}'>corpus_cfp_norm.txt
```

```
$ cat tmp.txt |gawk '{print $1 " " $2}'>corpus_rem_norm.txt
```

A sequência completa das fases de treino e teste, a nível de execução de *scripts*, é descrita da seguinte maneira:

1) Criação dos ficheiros de padrões para os conjuntos de treino, validação e teste através do *script* *gerar\_pat.x*

```
gerar_pat.x valid.txt dicionario.txt valid.pat
gerar_pat.x valid_n.txt dicionario_norm.txt valid_n.pat
gerar_pat.x train.txt dicionario.txt train.pat
gerar_pat.x train_n.txt dicionario_norm.txt train_n.pat
```

2) Uso do programa Stuttgart Neural Network Simulator (SNNS) para treino da rede neuronal:

- Abertura do ficheiro SNNS\_NET.75.50.15.25.ini.net representa a rede neuronal.
- Abertura dos ficheiros de padrões *valid\_n.pat* e *train\_n.pat* correspondentes aos conjuntos de validação e treino respetivamente.
- Foi usada a configuração de treino apresentada pela figura abaixo:

STEPS	1	STEP	0	JOG	INIT	RESET	ERROR	INFO	
CYCLES	1500	SINGLE	ALL	STOP	TEST	SHUFFLE	EDITORS	act	
PATTERN	1	DELETE	MOD	NEW	GOTO	◀	▶	SUB PAT	
		DEL SET	USE	cv_2a9_cfp					
VALID	1	VALID	USE	cv_1_cfp					
LEARN	0.7	0.0							
UPDATE								SEL. FUNC	
INIT	1.0	-1.0							SEL. FUNC
REMAP								SEL. FUNC	
DONE Learning func: Std_Backpropagation									

Figura 8 - Configuração de treino

- Para exemplificar melhor esta fase, analise-se o gráfico abaixo resultante do treino da rede neuronal com o conjunto de treino train\_n.pate validada pelo conjunto de validação valid\_n.pat.

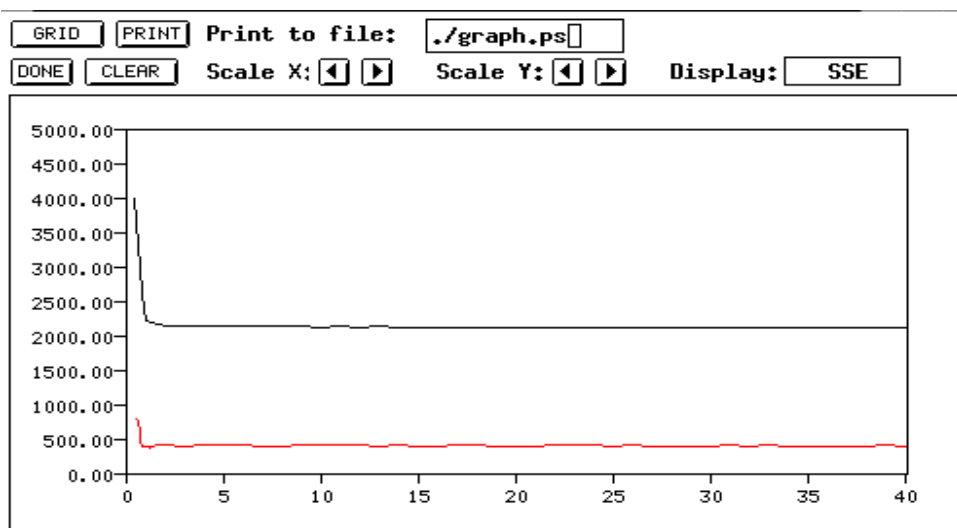


Figura 9 - Gráfico obtido durante o treino

- Gravação da rede neuronal treinada no ficheiro SNNS\_NET.75.50.15.25.fim.net
- Transformação da rede neuronal treinada no ficheiro executável referente ao etiquetador (nntagger) através da sequencia de comandos:
  - ❖ `ssns2.c SNNS_NET.75.50.15.25.ini.net etiquetador.c`
  - ❖ `gcc nntagger.o etiquetador.c -o nntagger -lm`

Consegue-se verificar que devido às etiquetas *OTHER* serem em excesso, e procedendo a uma remoção substancial da mesma do texto, obteve-se uns resultados melhores, isto é, a

aprendizagem foi melhor e deixou de haver tanta confusão de classificação. Logo a quantificação total dos erros de classificação dos elementos é menor e por consequência a sua taxa de erro também.

### **2.1.5.3. Parser de Dados de Conferências**

O *parser* das conferências tem como objetivo e tendo em conta as regras de formação das conferências, conseguir detetá-las a partir de um texto. Depois de as palavras estarem classificadas por etiquetas, e tendo no texto uma palavra por linha pode-se dar início á extração das conferências. O *parser* das conferências verifica se as etiquetas que identificam as componentes de uma conferência ocorre no texto. Depois de extraídas e filtradas, as conferências deverão ser inseridas na base de dados das conferências. Para inserir esses dados foi escolhido o PHP como meio de transmissão de dados.

O próximo passo consistia em guardar a informação resultante da extração de conferências numa base de dados. Isto é, “alimentar” a base de dados através da inserção dos dados, mas devido ao facto de eu ter transitado para uma outra entidade patronal não foi possível a sua conclusão.

### **2.1.5.4. Interface Web**

A informação resultante da extração de *emails* com anúncios de conferências é guardada numa base dados criada para o efeito. Foi criada uma interface que possibilitasse a consulta e gestão da informação, facilitando o acesso a essa mesma informação.

#### **2.1.5.4.1. Descrição da Interface**

A interface Web implementada para o sistema tem como finalidade ser responsável pela gestão de toda a informação contida na base de dados. A interface Web disponibilize todas as funcionalidades essenciais, tais como pesquisa, inserção e remoção de dados. A interface que apesar de ser bastante simples, permite uma gestão bastante intuitiva da informação contida na base de dados. O aspeto global da Interface é apresentado pela figura abaixo. Trata-se da interface inicial, ou seja, é o ponto de partida para as pesquisas disponibilizadas na *frame* à esquerda, ou para quem tiver permissões de administrador, efetuar login na área de administração.

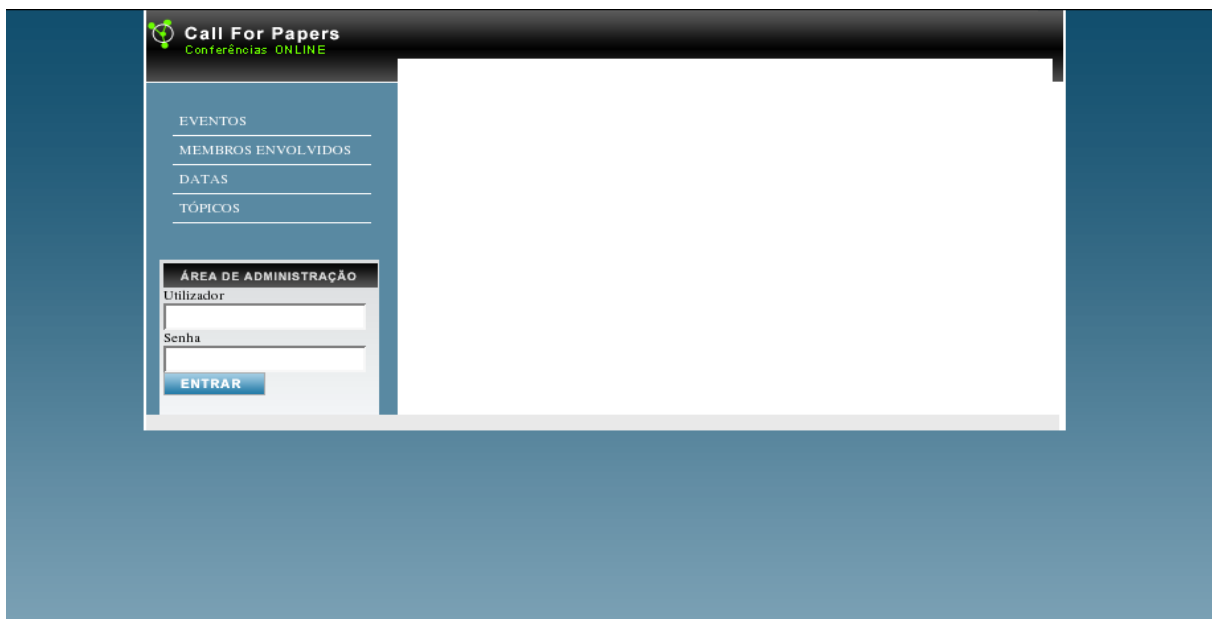


Figura 10 - Interface Inicial

Na *frame* esquerda da interface Web para além de se encontrar um menu com as principais funcionalidades é apresentado um formulário que permite ao utilizador *fazer login*. A ação de *login* permite ao utilizador passar para modo de administrador e assim desfrutar de privilégios especiais. Esses privilégios especiais consistem na permissão para efetuar operações de inserção, remoção e alterações de dados. Após o login essas operações passam a estar disponíveis nas várias páginas que constituem a interface Web. Por exemplo após o utilizador fazer *login* e entrar em modo administrador a página anteriormente apresentada ficaria com o aspeto seguinte:



Figura 11 - Interface depois de efetuado o Login

As funcionalidades foram divididas em grupos segundo a sua operacionalidade. Em modo de utilizador estão disponíveis no menu os grupos de operações de pesquisa relacionados com a consulta de dados:

- Eventos
- Membros Envolvidos
- Datas
- Tópicos

Exemplo de uma consulta de eventos:

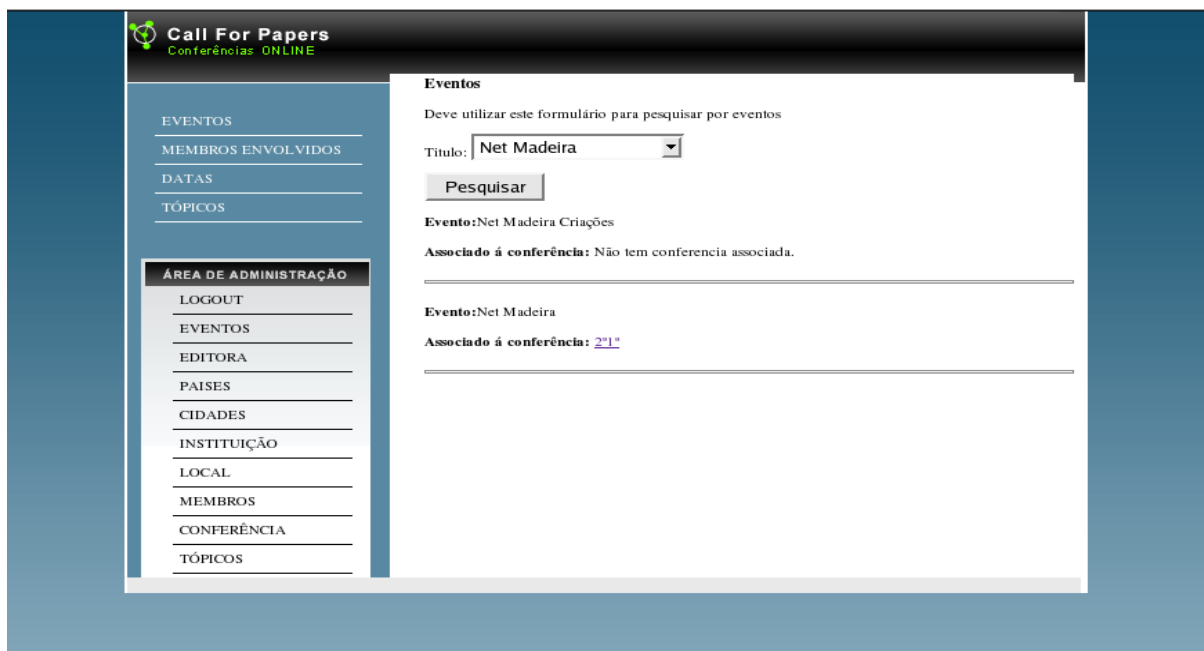


Figura 12 - Interface com um Exemplo de uma consulta de eventos

Pode-se verificar que associado ao evento “Net Madeira” tem duas edições de conferências. Escolhendo por exemplo a 1ª edição, podemos ter acesso aos detalhes dessa conferência, como se pode ver no exemplo abaixo disponibilizado:

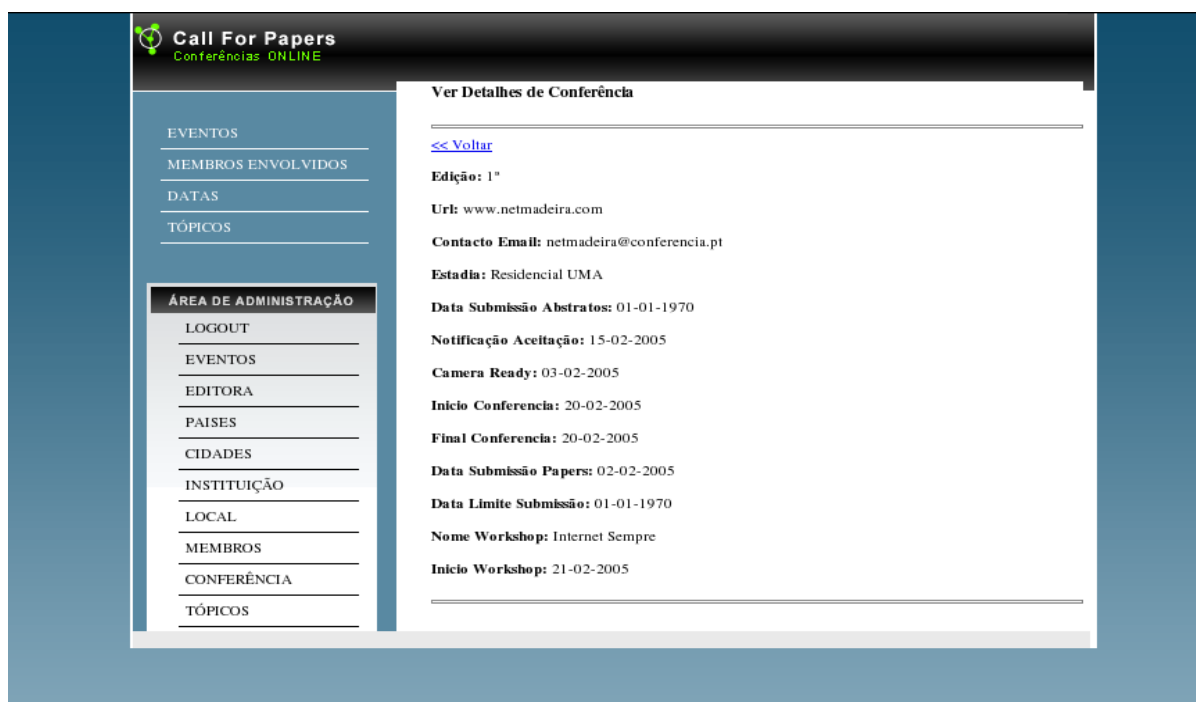


Figura 13 - Exemplo de uma Interface com detalhes dessa conferência

Em modo de administrador é disponibilizado adicionalmente o grupo de operações, relativo à inserção de dados, atualização de dados e remoção de dados:

- Logout
- Eventos
- Editora
- Países
- Cidades
- Instituição
- Local
- Membros
- Conferência
- Tópicos

Exemplo de uma operação sobre uma conferência:

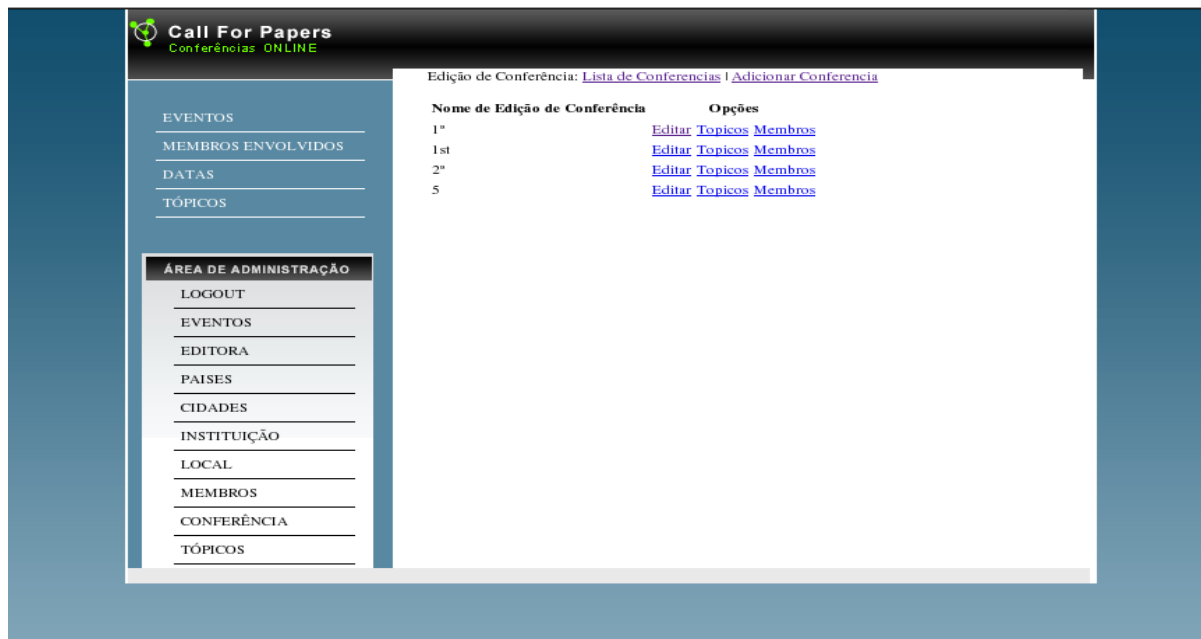


Figura 14 - Interface com um Exemplo de uma operação sobre uma conferência

Nesta interface conferência, encontra-se disponibilizada uma listagem com todas as edições de conferências existentes até ao momento. Associado a cada edição temos 3 operações possíveis, que são editar conferência, associar/remover tópicos à conferência e associar/remover membros à conferência. É possível também adicionar uma nova conferência.

Escolheu-se como exemplo a opção editar conferência, como mostra a figura abaixo:



Figura 15 - Exemplo de uma Interface com a opção editar conferência

Nesta interface é possível editar os dados já gravados e proceder a alterações se o administrador assim o entender.

#### **2.1.5.4.2. Implementação**

A interface Web foi implementada com recurso à linguagem PHP. O PHP é uma linguagem que permite criar páginas Web dinâmicas. Uma outra funcionalidade da linguagem PHP é o seu suporte para estabelecer ligações com vários tipos de base de dados, entre os quais o *PostgresSQL*.

Para estabelecer conexão com a base de dados em *PostgresSQL* o PHP disponibiliza várias alternativas, as conexões do tipo *PostgresSQL* são uma extensão do PHP e são exclusivas para ligações com base dadas *PostgresSQL*.

#### **2.1.6. Análise crítica dos resultados**

Apesar de algumas dificuldades sentidas em algumas fases deste projeto, penso que o sistema desenvolvido cumpriu quase a totalidade dos requisitos definidos inicialmente.

Para garantir um nível elevado de qualidade de dados foram definidas algumas limitações a nível do tipo de conferências a detetar. Algumas dessas limitações foram impostas aquando da definição das estruturas das bases de dados. Outras limitações foram, no entanto, impostas com o objetivo de melhorar a qualidade das conferências detetadas.

A fase mais complicada ao longo deste grande e longo projeto foi treinar a rede neuronal para esta conseguir classificar as palavras das conferências, pois o meu conhecimento sobre redes neuronais, assim como as suas metodologias eram muito poucos.

O etiquetador original estava preparado para classificar as palavras segundo a gramática de língua inglesa através do dicionário interno de palavras onde está armazenada as informações das ocorrências dos vários tipos de palavras num texto. Foi necessário construir um dicionário de raiz, com base na marcação do corpus e, então, depois adaptou-se o dicionário interno das palavras originais para a realidade das conferências, uma tarefa que se verificou complicada. O modelo de trigramas utilizado pelo etiquetador foi fundamental na resolução, pois o mesmo

baseia-se no contexto de uma palavra, onde a mesma é constituído pelo vizinho anterior e posterior da palavra em questão.

O etiquetador transforma o ficheiro de texto num ficheiro de padrões e o mesmo é passado para a rede neuronal *feedforward*, que possui apenas duas camadas de neurónios: entrada e saída. A rede neuronal no final devolve uma combinação de valores nos seus neurónios de saída. De realçar que nem sempre foi determinada a etiqueta correta associada a cada palavra, o que demonstra a complexidade de todo este processo.

Treinar esse mesmo dicionário foi um processo algo demorado e complicado, pois o corpus de texto original foi dividido em 10 subconjuntos, deixando um de fora e treinando com os restantes 9 em cada fase. Isto significa que haverá 10 combinações possíveis de treino e teste. Nesta fase constatou-se que não foi nada fácil atingir uma percentagem de acerto geral de etiquetas na ordem dos 80%, como inicialmente estipulado, pois foi necessário um importante processo de remoção de algumas etiquetas (Ex: "OTHER"), nos diferentes 10 subconjuntos e os devidos treinos e testes até se atingir o resultado pretendido, isto é, a aprendizagem foi melhor e deixou de haver tanta confusão de classificação. Logo a quantificação total dos erros de classificação dos elementos é menor e por consequência a sua taxa de erro também. O método de Cross – Validation, utilizado no treino dos subconjuntos foi importante, pois o mesmo define que em cada treino é omitido um conjunto de casos, sendo o etiquetador treinado com os restantes.

A chave do sistema de deteção de conferências é o etiquetador, pois se ele conseguir marcar corretamente todos os componentes de uma conferência, existirá então uma grande possibilidade de a mesma ser futuramente detetada com sucesso.

Foi com enorme satisfação que todas estas dificuldades foram ultrapassadas e o resultado final foi um sistema a funcionar e a preencher os requisitos propostos.

O sistema tem a tendência a crescer. Se o sistema fosse menos rígido, seria possível verificar que, a nível do formato das conferências, é provável que o número de conferências detetadas fosse maior. No entanto, é igualmente provável que fossem detetadas mais conferências inválidas. Sendo assim, seria interessante desenvolver um sistema mais "genérico" que abrangesse outros formatos de conferências.

### **2.1.7. Contribuições / Responsabilidades no Projeto**

Neste projeto e dado que o mesmo não envolvia uma equipa, mas apenas uma única pessoa que era eu, a minha contribuição foi total no mesmo. Ou seja, eu era o líder do projeto, delineava as áreas de conhecimento específico, tecnológico e funcional que seriam vitais para a implementação do projeto e também era eu que definia as informações necessárias e onde as mesmas ficariam armazenadas.

O desenvolvimento do projeto permitiu aprofundar conhecimentos de plano do mesmo como, por exemplo, limitações do mesmo, situações de estagnação, obstáculos encontrados e avaliação de desempenho. Aprendi mesmo muito ao nível da implementação de projeto, pois quando a implementação se inicia, é preciso que haja relatórios de progresso regulares agendados para determinar o quanto já foi feito e/ou o que ainda precisa ser realizado. Foi possível evoluir em redes neuronais e utilizar algumas ferramentas UNIX para tratamento de texto. Permitiu ainda cimentar alguns conhecimentos em *PostgreSQL*, utilizado para gestão de bases de dados.

Desta forma, posso concluir que, apesar da maioria dos conceitos abordados neste projeto serem inicialmente desconhecidos para mim, razão pela qual tive alguma dificuldade em algumas fases do projeto, a minha contribuição no desenvolvimento do mesmo foi bastante positivo e capacitou-me de conhecimentos ao nível de gestão de projetos.

## **2.2. Apoio de Microinformática a clientes PT**

### **2.2.1. Contexto**

Em 2005 o grupo Portugal Telecom cria a marca *System Care*. O suporte técnico *System Care* consistia em apoio de Microinformática a clientes PT. A criação desta marca tinha como finalidade desenvolver o conceito de "*Outsourcing*", ou seja, dar uma nova dimensão à estratégia de subcontratação na medida em que significava entregar determinadas tarefas a alguém que as faz melhor, mais rápido e a custos mais baixos, porque se especializou nelas, reservando mais tempo, recursos e energias da sua empresa para o que é o seu "*core business*".

No âmbito deste projeto assente numa vasta rede de diferentes áreas de "*Helpdesk*", fiquei inserido numa equipa cuja finalidade consistia em apoio de Microinformática a clientes PT: TMN, Telepac, Sapo e Netcabo.

### **2.2.2. Requisitos e Motivação**

O objetivo principal deste projeto visa o apoio ao cliente PT, através da disponibilização e articulação dos vários canais (telefone, e-mail, etc.), bem como das bases tecnológicas mais modernas, nomeadamente da ferramenta “assistente netcabo virtual”, que permite acesso remoto ao PC do cliente. Assim sendo, uma ferramenta destas pode ser muito valiosa e motivadora de trabalhar, pois tornou o serviço de assistência informática, numa referência no mercado nacional.

### **2.2.3. Pesquisa prévia / Análise do problema**

Sempre que o cliente solicite apoio a um dos seguintes exemplos, eis as soluções disponíveis:

- Configuração de redes sem fios
- Detecção e eliminação de vírus, *spyware* e *adware*
- Detecção e resolução de anomalias no sistema operativo e noutras plataformas informáticas utilizadas pelo Cliente
- Instalação, configuração, atualização e personalização de sistema operativo
- Otimização, instalação e atualização de drivers e componentes de *hardware*
- Detecção rápida de constrangimentos associados ao *hardware* instalado no computador.
- Instalação de equipamentos periféricos

Depois de apresentado pelo cliente o problema é proposto pelo operador que o mesmo proceda a determinados “passos” que permitam efetuar o despiste técnico tendo em vista a solução do constrangimento. De realçar que o operador tem de estar bem preparado para dar resposta a situações novas e de difíceis resoluções, o que por vezes torna muito complicado a análise prévia do problema.

### **2.2.4. Solução Técnica e Execução do Projeto**

#### **2.2.4.1. Aplicação Genesis**

Geralmente a cada contacto de um cliente, e antes de iniciar o despiste técnico, o operador tem obrigatoriamente de efetuar o registo da interação com o cliente, efetuando o registo da chamada telefónica na aplicação “*Genesis*”. Nesse registo é sempre colocado a informação referente ao motivo de contacto do cliente.

Por uma questão de segurança era obrigatório o cliente fornecer telefonicamente certos dados pessoais, que eram confirmados pelo operador. De realçar que se os mesmos não fossem validados pelo operador, o despiste técnico não era efetuada e o contacto telefónico terminado.

The screenshot shows the 'Tenant: milcaf - Milan Cafe' window. The left pane shows a tree view with '6401' selected under 'milcaf - Milan Cafe'. The main area has three tabs: 'Tenant Info', 'Other Info', and 'Lease Info'. The 'Lease Info' tab is active, showing 'Lessee Information' with fields for Name (Antonio Pillara), Company (Milan Cafe), Address (6401 Humbolt Avenue), City-State-Zip (Santa Barbara CA 93105), Net Sqft (3,350.0), Gross Sqft (3,500.0), % Sqft (10.7692), and \$ / Sqft (0.9018). There are also checkboxes for 'Management company pays rent', 'Charge base rent separately from increases', and 'Use base relative index increases'. A 'Data Fields' section contains a table with fields like Bank, Business, Insurance, T.I., Option 1, and Option 2, each with corresponding values and dates.

Bank	Account#	Rating
Santa Barbara B & T	193-047-209	9
Business	Employees	DateEst
	12	1984
Insurance	Coverage	Expires
Continental	1,500,000	12/31/09
T.I.	Cost	\$/sqft
Option 1	Term	Opt.Date
Option 2	Term	Opt.Date

Figura 16 – Aplicação Genesis

#### 2.2.4.2. Aplicação GSU

Seguidamente, o operador confirma na aplicação *GSU* (Gestão de Sistemas e Utilizadores) se os serviços contratados estavam ativos e confirma também os dados do cliente. É essencial confirmar os serviços porque muitos clientes se encontram com os mesmos suspensos por falta de pagamento.

#### 2.2.4.3. Aplicação RADIUS

O *RADIUS* (Remote Authentication Dial In User Service) é um protocolo para aplicações para acesso à rede de computadores e mobilidade através de rede IP.

O *phpRADmin* é uma aplicação *Web* implementada para gestão de *RADIUS*. Esta aplicação permite a gestão de utilizadores, verificar o estado do servidor. Possui ainda geração de estatísticas e perfis de utilizadores. A aplicação *RADIUS* servia para verificar se o *username* e *password* do cliente estava autenticada corretamente. Se os dados configurados no *modem/router* do cliente estavam a autenticar e a disponibilizar *ip* para acesso à internet.

Status	Issued	Expires	User's Name	E-mail	Organization	Department	Locality
Valid	06-Jun-12	07-Jun-12	radius2.blyx.com	admin@blyx.com	radius network	systems	Madrid
Valid	06-May-22	07-May-22	asdfsdfasdf	asdfsdfasdf@asdfsdf.com	radius network	systems	Madrid
Valid	06-May-23	08-May-23	asdfsdfasdf	asdfsdfasdf@asdfsdf.com	radius network	systems	Madrid
Valid	06-Jun-12	07-Jun-12	ksskkskisi	ksskkskisi@ksskkskisi.com	radius network	systems	Madrid
Revoked	06-May-22	07-May-22	ksskkskisi	ksskkskisi@ksskkskisi.com	radius network	systems	Madrid
Revoked	06-Jun-03	07-Jun-03	syvic	syvic@sindomin.net	radius network	systems	Madrid
Revoked	06-Jun-03	07-Jun-03	syvic	syvic@sindomin.net	radius network	systems	Madrid
Valid	06-Jun-12	07-Jun-12	xyvic	xyvic@sindomin.net	radius network	systems	Madrid
Revoked	06-May-22	07-May-22	radius.blyx.com	toni@blyx.com	radius network	systems	Madrid
Valid	06-Jun-12	07-Jun-12	radius.blyx.com	toni@blyx.com	radius network	systems	Madrid
Revoked	06-May-22	07-May-22	toniblyx	toniblyx@gmail.com	toniho	red	Madrid
Valid	06-May-29	08-May-29	user-ipsec	user-ipsec@user-ipsec.com	radius network	systems	Madrid
Valid	06-May-29	08-May-29	user1	user1@blyx.com	radius network	systems	Madrid

Figura 17 – Aplicação Radius

#### 2.2.4.4. Despiste técnico

Após estas confirmações dá-se início ao despiste técnico. Como já foi dito anteriormente, o cliente pode solicitar apoio a um vasto leque de possíveis anomalias, que pode ir desde uma simples recuperação de *password* de email até outras mais complicadas como resolver constrangimentos associados ao *hardware/Software* de um computador. Na grande maioria das vezes era solicitado ao cliente que efetuasse determinados “passos” indicados pelo operador com o intuito de resolver o problema. De realçar que se o cliente não se encontrasse ao pé do equipamento, era dada a indicação para voltar a entrar em contacto com o serviço.

##### 2.2.4.4.1. Aplicação Assistente virtual NetCabo

Caso o problema não ficasse solucionado o operador proponha ao cliente uma intervenção remota no computador do mesmo através da aplicação “assistente virtual netcabo”. O programa destina-se a auxiliar o cliente PT a detetar e resolver problemas de acesso ao serviço, bem como facilitar a intervenção remota do operador no despiste de anomalias técnicas, sempre que solicitada pelo cliente através da linha de atendimento.

A aplicação está disponível para *download* e é gratuita. Antes do cliente instalar a aplicação no seu computador, deve ler atentamente os termos de utilização do programa. A instalação e utilização do programa pressupõe a aceitação e conhecimento dos termos e condições pelo que, em caso de não aceitação, não é permitida a instalação e utilização do mesmo.

Esta aplicação tornou-se uma mais valia porque permite ajudar em muito tanto os clientes como os operadores pois otimiza recursos na solução de constrangimentos técnicos. As sessões remotas são encriptadas e as *passwords* são fornecidas aos operadores com a autorização do cliente. O cliente pode monitorizar todas as ações remotas do operador podendo, a qualquer momento, desligar a sessão. No final do despiste técnico o operador coloca na ficha de cliente através da aplicação “Genesis” se o constrangimento foi resolvido com sucesso ou não. Em caso negativo é agendado a deslocação de uma equipa técnica para resolução de anomalias presencialmente.

### **2.2.5. Análise Crítica dos Resultados**

Com estas ferramentas de CRM (Customer Relationship Management) acima descritas, é mais fácil dar uma resposta ao cliente, podendo-se maximizar a eficiência operacional, pois eram definidas um conjunto de metodologias que proporcionam a gestão e o fortalecimento das relações com os clientes.

Os dois principais aspetos fundamentais para considerar a necessidade de dispor de um centro de atendimento como a *System Care*, são a qualidade no processo de resposta ao cliente, valorizado como aspeto fundamental e a eficiência operacional entre recursos utilizados e contactos efetuados. Além disso, são também considerados importantes, aspetos como eficácia, rapidez na resposta, gestão de custos e cobertura do ciclo de relação com o cliente.

Obviamente que todas estas ferramentas de CRM constituem uma mais-valia nos objetivos finais da empresa, pois proporciona maior eficiência nos negócios e otimiza o fluxo de informação entre as organizações e os clientes.

Sobre as possíveis desvantagens, sinceramente não acho que exista, pois não encontro desvantagens em se querer melhorar o processo de uma empresa e atender bem o cliente. Por vezes, o que pode acontecer é se optar por desistir de uma determinada ferramenta de CRM, devido a problemas com a mesma, mas nunca por mau planeamento.

Por exemplo, algumas das ferramentas de CRM usadas na empresa, como a aplicação RADIUS tinha algumas desvantagens, pelo facto de utilizar JavaScript, faz com que esta aplicação apresente alguns problemas de compatibilidade a diferentes browsers, mas também a mesma era limitada em termos de interatividade e a sua adaptabilidade era também inexistente.

O lançamento da aplicação Assistente foi um virar de página e um passo em frente tanto para os clientes como para a *System Care*, pois otimiza recursos na solução de constrangimentos técnicos. A facilidade de configuração do Assistente é um dos principais atributos, porque desta forma melhora o serviço de apoio prestado ao cliente, através da simplificação dos processos.

A sua utilização passa a permitir que, ao invés de ligar imediatamente para o apoio telefónico, o cliente possa certificar-se, a partir do seu computador, do estado da sua ligação, conectividade e outras questões que possam impedir uma utilização bem-sucedida do serviço de Internet.

O *chat* funciona em tempo real e permite que os utilizadores perguntem diretamente a um operador as suas dúvidas acerca das anomalias detetadas evitando a chamada telefónica e usando o interface de assistência online.

A opção para ligações ao *wireless* permite que os utilizadores protejam a sua ligação, evitando a usurpação da rede por outros internautas. Também esta funcionalidade é apresentada ao cliente de forma simples e com uma utilização intuitiva.

Finalmente, e em último caso, era efetuado o acesso remoto ao computador do cliente sempre que não fosse possível resolver o constrangimento e caso o cliente assim o permita. Sem dúvida, esta funcionalidade tornou-se uma mais-valia porque permite ajudar tanto os clientes como os operadores.

O investimento desta plataforma rondou os 250 mil euros, com o desenvolvimento base da versão do assistente netcabo a ser desenvolvido pela empresa norte-americana SupportSoft e customizada, pela equipa técnica da marca System Care do grupo PT. Esta ferramenta de apoio interativa torna-se a única em Portugal a disponibilizar estas soluções. O objetivo da empresa é o aumento global da satisfação dos utilizadores, meta que tem vindo a ser cumprida com uma redução de chamadas mensais para o call center.

Os resultados foram extremamente animadores, pois desde o *hardware* até ao software, somávamos o *know-how* de uma equipa de colaboradores altamente especializados e permanentemente submetidos a formação credenciada, por forma a garantir a satisfação das necessidades efetivas do cliente.

### **2.2.6. Contribuições / Responsabilidades no Projeto**

Trabalhei a tempo inteiro na PT Comunicações entre janeiro de 2006 a abril de 2007 nas funções de operador técnico de 2ª linha, (operadores da 1.º linha técnica resolviam questões técnicas básicas). As minhas responsabilidades eram não só de esclarecimento de serviços aos clientes, mas essencialmente de configuração desses mesmos serviços (já descritos anteriormente).

Acho que disponha dos conhecimentos, da prática e da experiência adequados às tarefas específicas deste projeto. Fui um profissional comprometido com o sucesso da operação em que estava envolvido. Disponha da flexibilidade e da visão necessária para dar resposta às particularidades que caracterizavam a operação destinada a cada cliente, contribuindo assim para um serviço de assistência informática de excelência

## **2.3. Montagem e Programação de Robôs**

### **2.3.1. Enquadramento Teórico**

A evolução tecnológica e as mais recentes teorias cognitivo-pedagógica têm direcionado o conceito de educar no sentido de disponibilizar ferramentas orientadas para o ato de aprender. Pretende-se apresentar um breve olhar sobre as teorias de aprendizagem, assim como as implicações dessas teorias no ato de aprender.

#### **2.3.1.1. Teoria do Construtivismo**

Construtivismo é uma teoria que explica como a inteligência humana se desenvolve partindo do princípio de que o desenvolvimento da inteligência é determinado pelas ações mútuas entre o indivíduo e o meio.

Esta concepção do conhecimento e da aprendizagem que derivam, principalmente, das teorias da epistemologia genética de Jean Piaget e da pesquisa socio-histórica de Lev Vygotsky, parte da ideia de que o homem não nasce inteligente, mas também não é passivo sob a influência do meio, isto é, ele responde aos estímulos externos agindo sobre eles para construir e organizar o seu próprio conhecimento, de forma cada vez mais elaborada.

### **2.3.1.2. Teoria do Construcionismo**

A montagem e programação de robôs sob o ponto de vista das teorias cognitivo-pedagógicas enquadra-se na teoria construcionista proposta pelo psicólogo Seymour Papert. Esta é fundamentada no construtivismo e baseia-se na construção do conhecimento apoiada na realização de uma ação concreta que resulta num produto real, desenvolvido com o recurso do computador. O construcionismo implica uma interação entre o aluno e o objeto, mediado por uma linguagem de programação, integrada com o robô. Nesta teoria, os alunos são encarados como participantes ativos, aprendendo de uma forma que depende do seu estado cognitivo real, ou seja, depende do seu processo de transformação e evolução.

### **2.3.2. Contexto**

Aprender a programar é um processo difícil e ser um bom programador exige vários tipos de aptidões a serem desenvolvidas. A generalidade dos alunos quando começa a estudar programação sente muitas dificuldades, existindo um elevado índice de insucesso nas disciplinas da área. A falta da contextualização da aprendizagem tem sido apontada como uma das razões para este insucesso, pois mais do que perceber algoritmos já feitos, os alunos têm dificuldade em elaborar soluções para novos problemas que lhes são colocados. Os algoritmos e programas computacionais são processos dinâmicos que leva a pensar que talvez os métodos de ensino tradicionais não são os mais adequados às necessidades dos alunos, porque a forma expositiva para ensinar programação dificulta a aprendizagem e não consegue facilmente motivar, a maioria dos alunos, a interessar-se pela mesma.

A natureza abstrata da programação é por si só uma fonte de dificuldade, logo é necessário haver uma consciencialização por parte dos alunos que a programação se aprende fundamentalmente fazendo e não apenas vendo como se faz, exigindo muito treino, persistência e um bom nível de conhecimentos e prática de técnicas de resolução de problemas.

Sendo assim, em 2006 surgiu o núcleo de informática denominado CLINFO, que mais recentemente foi reestruturado e o nome alterado para CyberFF. A grande finalidade do projeto CyberFF é o de cativar os alunos e toda a comunidade escolar para uma área tecnologicamente avançada como é a programação realizando atividades de extremo enriquecimento e de mais-valia para aqueles que nelas participam.

### **2.3.3. Requisitos e Motivação**

O projeto CyberFF visa criar um espaço para que toda a comunidade educativa possa ter um contato mais direto e experimental com a Informática.

Para os alunos, apresentamos as várias facetas da Informática, mostrando como esta pode ser um instrumento facilitador de aprendizagem, deixando para trás a ideia de que é só um mundo de jogos, de entretenimento e de lazer. Para isso realizamos algumas atividades como demonstrações de robótica, exposições de material informática, concursos, entre outras.

De salientar que a Universidade da Madeira efetuou uma parceria com a escola Secundária de Francisco Franco onde disponibilizou diversos problemas, agrupados em atividades, que se destinam não só ao apoio das disciplinas de Bases de Programação, do curso Tecnológico de Informática, mas essencialmente a promover a aprendizagem da algoritmia e a potenciar nos alunos a aquisição do raciocínio lógico.

Para os professores, pretendemos a criação de um espaço físico e virtual (blog), onde possa existir a partilha de experiências e conhecimentos entre os vários participantes. Estes espaços pretendem motivar a utilização das tecnologias de comunicação, dotando os participantes de ferramentas inovadoras, que poderão contribuir para o enriquecimento das suas atividades profissionais e lúdicas.

Para a comunidade escolar, almejamos a participação na revista da escola, a colaboração com os outros clubes, núcleos e projetos e ainda a organização de conferências e exposições.

#### **2.3.4. Pesquisa prévia / Análise do problema**

Pretende-se com estas atividades sobre “Montagem e Programação de Robôs” dar a conhecer aos alunos os conceitos básicos dos sistemas robotizados nas suas vertentes: - robôs articulados - robôs móveis - e conhecer a programação básica de cada um dos diferentes tipos de robôs.

No geral e para todas as atividades propostas os alunos tinham de efetuar uma pesquisa para melhor:

- Noção de Algoritmo e Programa;
- Identificação de variáveis e expressões;
- Escolha e compreensão de estruturas de decisão;
- Escolha e compreensão de estruturas de repetição;

Os alunos eram sempre acompanhados e orientados pelos professores nas soluções por eles idealizadas.

## 2.3.5. Solução Técnica e Execução do Projeto

### 2.3.5.1. Descrição do Kit Lego® Mindstorms™



Figura 18 - Robotics Invention System 2.0

#### 2.3.5.1.1. Constituição e Funcionamento

O kit LEGO® Mindstorms™ é constituído por uma unidade de controlo – o RCX, uma torre de comunicação com o computador por infravermelhos, dois motores, dois sensores de toque e um sensor de luz.

##### 2.3.5.1.1.1. O RCX

A unidade de controlo RCX (Robotics Command Center) é a base onde todos os sensores e motores são ligados. Este é o “cérebro” do kit LEGO® e tem como dimensões: 9,5 cm de comprimento, 6,3 cm de largura e 4 cm de altura.

O RCX possui três entradas (1, 2 e 3) onde podem ser ligados os sensores, assim como três saídas (A, B e C), para os motores. É constituído ainda por um visor LCD, por um módulo de transmissão por infravermelhos, um conjunto de 4 botões, onde o botão vermelho tem como função ligar/desligar o RCX, o botão verde tem como função iniciar/parar um programa, o botão cinza tem como função seleccionar o programa desejado e o botão preto possibilita monitorizar as entradas ou saídas do RCX, conforme mostra a figura em baixo.



Figura 19 - O RCX da Lego MindStorms

#### 2.3.5.1.1.2. Os Motores

Podem ser ligados até 3 motores, nas respectivas nas portas A, B e C do RXC. Para tal é necessário usar um cabo específico, conforme mostra a figura em baixo.



Figura 20 - Motor ligado a saída do RCX

#### 2.3.5.1.1.3. Os Sensores

O kit RCX possui diversos sensores. Os sensores que normalmente vêm com o kit são o sensor de toque e o sensor de luz. O sensor de rotação e o sensor de temperatura, são sensores adicionais ao kit. Estes sensores podem ser adquiridos na empresa LEGO®.

#### 2.3.5.1.1.3.1. Sensores de Toque

O sensor de toque é um sensor simples e funciona basicamente como um interruptor. É constituído por um pequeno bloco de plástico que possui um botão na sua extremidade. Quando o mesmo é pressionado envia dados ao RCX que os analisa através de um dos seus programas.



Figura 21 - Sensor de Toque

#### 2.3.5.1.1.3.2. Sensores de Luz

A componente fundamental do sensor de luz é o transistor Bipolar. Esta componente é atravessada por uma corrente que é proporcional à intensidade luminosa.

Numa perspetiva simplista, o funcionamento do sensor de luz pode resumir-se ao seguinte esquema funcional:

É constituído por um diodo emissor de luz (LED) e um transistor Bipolar, que absorve a luz vinda do ambiente. O mesmo precisa de energia do RCX para poder funcionar. As medidas feitas pelo sensor são armazenadas em percentagem (0 a 100%, sendo mais comuns os valores entre 40 e 60). O RCX utiliza essas medidas para determinar qual a luminosidade do ambiente em que o robô se encontra.



Figura 22 - Sensor de Luz

#### 2.3.5.1.1.4. Torre de Transmissão

Esta torre é responsável por enviar para o recetor de infravermelho existente no RCX toda a informação. Para esse envio de *firmware* ou programas é necessário um computador ao qual se liga, via porta USB, uma torre de infravermelhos. Toda a comunicação entre o computador e o RCX é efetuada via infravermelhos, utilizando o emissor/recetor presente no RCX e na torre.



Figura 23 - Comunicação entre o computador e o RCX



Figura 24 – Torre de Infravermelhos

#### 2.3.5.1.1.5. O Firmware

O *firmware* não é nada mais nada menos que o sistema operativo do RCX que faz funcionar todo o sistema. É o responsável pela interpretação dos programas em *bytecode* e convertê-los em instruções máquina. O *firmware* fica armazenado na RAM do RCX, pelo que deve ser feito o download, via torre de infravermelhos, a cada primeira instalação do sistema.

##### 2.3.5.1.1.5.1. Lego

O *firmware* padrão para o RCX, é um sistema operativo constituído basicamente por uma máquina virtual. Tem como tarefa básica a interpretação do *bytecode* e transformá-los numa série de instruções e dados apropriados aos microprocessadores do RCX. O RIS (Robotics Invention System) é o *IDE's* que gera *bytecode* para o *firmware* padrão.

Uma das vantagens da máquina virtual é que se a arquitetura for alterada o *firmware* pode continuar a ser compatível com os programas escritos antes das alterações.

##### 2.3.5.1.1.5.2. BrickOS

O *BrickOS* é um *firmware* alternativo ao RCX, e suporta as linguagens de programação C, C++ e Pascal. Trata-se de um sistema operativo multitarefa, pois permite a gestão dinâmica da memória, semáforos, secções críticas e um acesso nativo ao LCD, aos botões, à comunicação por infravermelho, aos motores e aos sensores.

### 2.3.5.1.2. Ferramentas de Programação

#### 2.3.5.1.2.1. Robotics Invention System

O *Robotics Invention System* (RIS) é um programa para computador que permite automação e controlo sobre dispositivos robóticos LEGO. É uma aplicação que permite a programação e a construção de robôs, arrastando blocos de código para a área do programa (RIS), conforme mostra a figura em baixo.



Figura 25 - Ambiente de programação do RIS

#### 2.3.5.1.2.2. RCX Command Center

Ao contrário da RIS a ferramenta *RCX Command Center* é indicada para pessoas que têm grandes conhecimentos de programação.

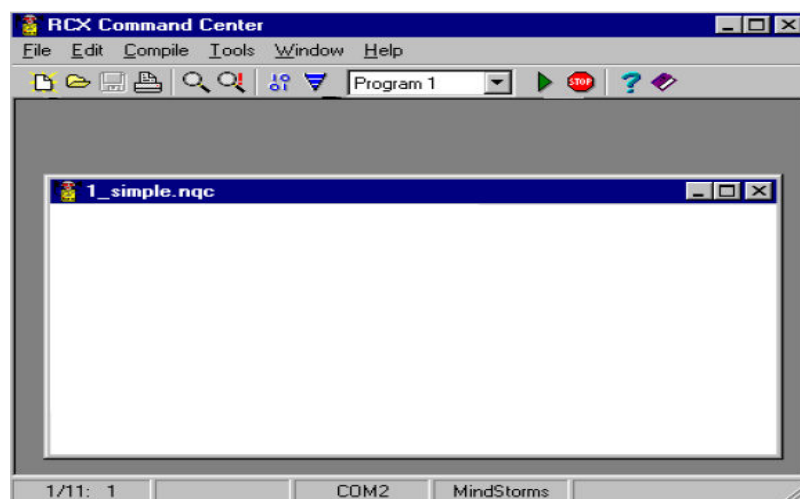


Figura 26 - Ambiente de programação do RcxCC

### 2.3.5.1.2.3. Bricx Command Center

Tal como o RcxCC, a ferramenta BricxCC é indicada para um público-alvo que possui alguns conhecimentos de programação, com a vantagem de suportar linguagem C, C++, Java e Pascal.

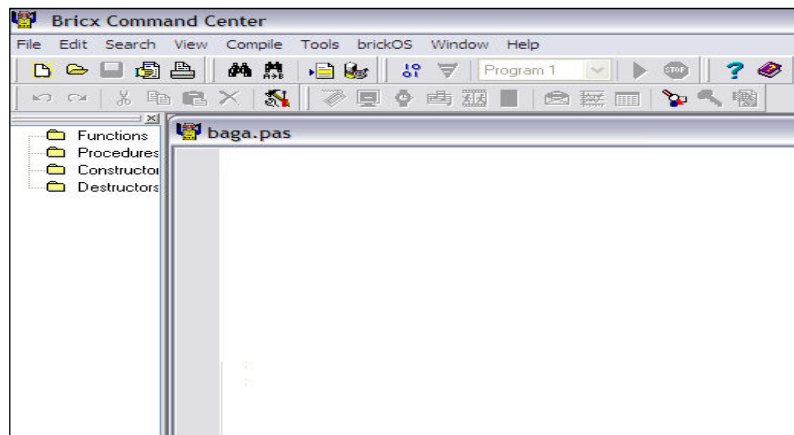


Figura 27 - Ambiente de programação do BricxCC

### 2.3.5.1.3. Linguagens de Programação

#### 2.3.5.1.3.1. Linguagem de Programação NQC

*NQC*, ou *Not Quite C*, é uma linguagem nativa, porque é uma linguagem que “corre” sobre o *firmware* Lego. É uma linguagem simples para programar os robots da Lego *MindStorms*. O compilador do *NQC* traduz o código fonte em *LEGO bytcodes*, que pode ser então executado no RCX sobre o *firmware* LEGO.

Programar em NQC consiste em tarefas (*task*). Cada programa necessita de uma tarefa com o nome de *main* (tarefa principal), que será automaticamente executada pelo robô. Uma tarefa consiste num conjunto de comandos, também denominados de instruções (*statements*). Cada instrução finaliza com um ponto e vírgula. Deste modo, torna-se claro onde uma instrução acaba e onde a próxima começa. De uma maneira geral, uma tarefa tem a seguinte estrutura:

```
task main()           //tarefa principal
{
  statement1;         //primeira instrução
  statement2;         //segunda instrução
  ...
}
```

Figura 28 - Estrutura de uma tarefa na linguagem NQC

### 2.3.5.1.3.1.1. Casos de Estudo

Numa primeira fase é importante que os alunos adquiram os conhecimentos e as competências básicas para trabalhar com os *kits* de Robótica da Lego *MindStorms*. Para todos os problemas propostos esta fase tem duas atividades principais, que são a construção dos robôs e a sua programação.

Após a construção do robô, o mesmo deve ser programado na linguagem NQC para que cumpra corretamente as tarefas propostas nas atividades seguintes. Para tal, deve ser usado o ambiente de programação RcxCC que está disponível no computador e que permite comunicar com o robô. As sessões dedicadas à programação dos robôs incluem a resolução de um conjunto de tarefas de complexidade crescente, segundo um guião pré-definido.

#### 2.3.5.1.3.1.1.1. Problema 1

##### ATIVIDADE 1

Digite as instruções abaixo e crie o seguinte programa:

```
task main()
{
  OnFwd(OUT_A);
  OnFwd(OUT_C);
  Wait(400);
  OnRev(OUT_A+OUT_C);
  Wait(400);
  Off(OUT_A+OUT_C);
}
```

O programa tem seis comandos. Faça uma pequena descrição do que consiste cada um deles:

- OnFwd(OUT\_A);

---

---

- OnFwd(OUT\_C);

---

---

- Wait(400);

---

---

- OnRev(OUT\_A+OUT\_C);

---

---

- Wait(400);

---

---

- Off(OUT\_A+OUT\_C);

---

---

Conseguiram resolver a atividade?

Sim.

Não. Porquê? \_\_\_\_\_  
\_\_\_\_\_

Consideram que a atividade foi:

Muito fácil

Fácil

Mais ou menos

Difícil

Muito Difícil

### 2.3.5.1.3.1.1.2. Problema 2

#### ATIVIDADE 2

Na atividade 1, podemos verificar que o robô se movimenta de forma bastante rápida. Por defeito, o mesmo move-se o mais rápido que pode. Para modificar esta velocidade deverá ser utilizado o comando *SetPower()*. A potência definida por este comando tem de ser um valor entre o número 0 e 7, onde o número 7 representa a velocidade mais rápida e o 0 representa a mais lenta.

Programa uma nova versão do programa construído na atividade 1, na qual o robô deva mover-se de uma forma lenta.

Conseguiram resolver a atividade?

Sim.

Não. Porquê? \_\_\_\_\_

\_\_\_\_\_

Consideram que a atividade foi:

Muito fácil

Fácil

Mais ou menos

Difícil

Muito Difícil

### 2.3.5.1.3.1.1.3. Problema 3

#### ATIVIDADE 3

Vamos tentar com a realização desta atividade 3, fazer algo mais interessante.

Pretende-se programar o robô para fazer curvas. Introduzindo algumas das características da linguagem de programação NQC, programe o robô de maneira a que o mesmo possa movimentar-se um pouco em frente e depois virar com um ângulo de 90 graus.

Nota: Pode ser necessário tentar alguns números próximos do 85 para fazer com que o robô vire precisamente com um ângulo de 90 graus.

Conseguiram resolver a atividade?

Sim.

Não. Porquê? \_\_\_\_\_

\_\_\_\_\_

Consideram que a atividade foi:

Muito fácil

Fácil

Mais ou menos

Difícil

Muito Difícil

#### 2.3.5.1.3.1.1.4. Problema 4

### ATIVIDADE 4

Pretende-se programar o robô de maneira a que o mesmo se desloque por um percurso, cujo formato é o de um quadrado.

Para o mesmo se movimentar nesse quadrado terá de obedecer às seguintes etapas:

1. Mover para a frente
2. Virar 90 graus
3. Mover para a frente de novo
4. Virar 90 graus
5. Etc.

Nota: Pode repetir o código do programa quatro vezes, mas o programa pode ser feito de maneira mais fácil, utilizando a estrutura de repetição “Repeat”.

Conseguiram resolver a atividade?

Sim.

Não. Porquê? \_\_\_\_\_

\_\_\_\_\_

Consideram que a atividade foi:

Muito fácil

Fácil

Mais ou menos

Difícil

Muito Difícil

### 2.3.5.1.3.1.1.5. Problema 5

#### ATIVIDADE 5

Pretende-se programar o robô de maneira a que o mesmo se movimente em linha reta durante algum tempo e depois vire para a esquerda ou para a direita. Utilize a função “*Random*” que gera números aleatórios entre “0 e 1”. Se o número gerado for “0”, o robô vira à direita, caso contrário vira à esquerda.

Nota: O programe pode ser feito utilizando a estrutura de repetição “*While*” e a estrutura de controlo “*If ... Else*”.

Conseguiram resolver a atividade?

Sim.

Não. Porquê? \_\_\_\_\_

\_\_\_\_\_

Consideram que a atividade foi:

Muito fácil

Fácil

Mais ou menos

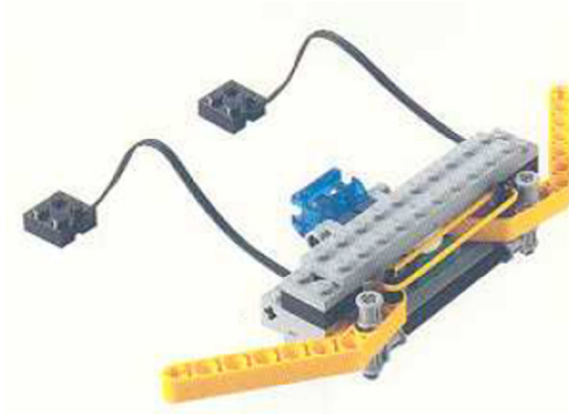
Difícil

Muito Difícil

### 2.3.5.1.3.1.1.6. Problema 6

#### ATIVIDADE 6

Um dos aspetos interessantes dos robôs Lego é que nos mesmos podem ser ligados sensores. Como tal é necessário modificar o robô que temos utilizado nas atividades anteriores e efetuar a montagem do respetivo sensor.



Vamos começar com um programa simples. Programe o robô para que o mesmo se desloque em frente até que bata em algo.

Conseguiram resolver a atividade?

Sim.

Não. Porquê? \_\_\_\_\_  
\_\_\_\_\_

Consideram que a atividade foi:

Muito fácil

Fácil

Mais ou menos

Difícil

Muito Difícil

### 2.3.5.1.3.1.1.7. Problema 7

#### ATIVIDADE 7

Vamos tentar com a realização desta atividade 7, fazer algo mais interessante. Vamos tentar programar o robô de maneira a desviar-se de obstáculos.

Para o mesmo se desviar dos obstáculos deverá obedecer às seguintes etapas:

1. Caso o robô bata num objeto
2. Mover para traz
3. Virar
4. Continuar
5. Caso o robô bata novamente num objeto
6. Mover para traz
7. Virar
8. Continuar
9. Etc.

Nota: O programa pode ser feito utilizando a estrutura de repetição “*While*”, com um *loop* infinito.

Conseguiram resolver a atividade?

Sim.

Não. Porquê? \_\_\_\_\_  
\_\_\_\_\_

Consideram que a atividade foi:

Muito fácil

Fácil

Mais ou menos

Dificil

Muito Dificil

### 2.3.6. Análise Crítica dos Resultados

Após a realização do projeto e fazendo uma reflexão sobre o desenrolar do mesmo considero que os objetivos propostos foram atingidos na sua totalidade. No geral, desenvolveu-se com sucesso todas as seis fases dos objetivos propostos:

<b>Fase 1</b>	Organização dos grupos de trabalho;
<b>Fase 2</b>	Montagem dos robôs em equipa;
<b>Fase 3</b>	Ambientação ao robot RCX e ao <i>software</i> de programação;
<b>Fase 4</b>	Desenvolvimento das atividades;
<b>Fase 5</b>	Apresentação dos trabalhos à comunidade educativa;
<b>Fase 6</b>	Perceções dos alunos sobre a programação e relativamente ao potencial da robótica educativa no desenvolvimento de projetos em trabalho colaborativo.

Desenvolver o raciocínio ou possibilitar situações de resolução de problemas é a razão mais nobre e irrefutável do uso do computador na educação. Quem não quer promover o desenvolvimento do poder de pensamento do aluno? Por exemplo, essa não é uma das razões pelas quais ensinamos programação na escola? No entanto, isso é fácil de ser falado mas não de ser conseguido.

Com este projeto foi possível motivar e despertar a curiosidade dos alunos para a robótica e para o ensino da programação, contribuindo para uma melhor contextualização do conteúdo, maior favorecimento do trabalho em grupo e também para uma melhor compreensão dos conceitos a trabalhar. Porém, considero que é necessário um estudo mais aprofundado sobre os conteúdos da metodologia e as suas orientações pedagógicas no que concerne à relação das montagens e programação dos robôs com os trabalhos nas diferentes disciplinas. Este projeto foi importante para percebermos que a robótica e a sua programação pode ser uma grande contribuição para o

trabalho na construção de um conhecimento partilhado mais contextualizado, pois o potencial da robótica mostra-se ser uma nova ferramenta no processo ensino-aprendizagem.

Com os conhecimentos que agora possuo, mas que antes não tinha, e aplicando a teoria construtivista e construcionista de aprendizagem ao serviço do ensino com a tecnologia, em particular na educação com o computador, julgo que foi dado um passo em frente no sentido do desenvolvimento do aluno, pois considero que aplicando estas teorias conseguirei propor problemas aos alunos que contenham elementos dentro da zona de desenvolvimento real, que é a zona cognitiva onde o mesmo pode trabalhar sozinho, mas também elementos da zona de desenvolvimento proximal, que é a zona cognitiva onde os alunos são capazes de resolver problemas, mas não são capazes de o fazerem sem ajuda, quer a mesma seja através do professor, de um livro, de um colega, de um vídeo ou do próprio computador. O trabalho em grupo permitirá que os alunos ultrapassem mais rapidamente a zona de desenvolvimento proximal e atinjam a zona de desenvolvimento real.

Realço que com a aplicação destas teorias os alunos no futuro irão mostra-se ainda mais motivados e empenhados em aprender. Sendo assim, encontra-se já em estudo mais atividades na mesma área.

### **2.3.7. Contribuições / Responsabilidades no Projeto**

Durante a ação os alunos foram ativos e sempre encorajados por mim a atingirem os objetivos das atividades. Devo Salientar que todos os alunos estavam extremamente motivados e entusiasmados com a atividade e mostraram-se sempre muito interessados e atentos, finalizando com sucesso a grande maioria das tarefas pretendidas.

Julgo que estas atividades foram realmente uma mais-valia para todos os alunos, já que um dos objetivos subjacentes à realização das mesmas era “despertar” nos alunos o gosto pela programação e “transportar” essa mesma motivação para as disciplinas do Curso Tecnológico de Informática, mais especificamente para a disciplina de Bases de Programação, dado que os alunos demonstram algumas dificuldades na assimilação de conhecimentos na mesma. Senti e constatei que após a realização das atividades os alunos ficaram mais motivados e predispostos a aprender a programar, inclusive, alguns alunos fizeram-me saber que gostaram imenso das

atividades e agradeceram-me pela oportunidade que lhes foi dada para “trabalharem” com os robôs. Nada mais motivador também para mim porque “senti” que estava a desempenhar bem o meu papel de professor.

A atividade em imagens:



**Fotografia 1 - Alunos do grupo I**



**Fotografia 2 - Alunos do grupo II**



**Fotografia 3 - Alunos na fase de programação do Robô “Todo o Terreno”**



**Fotografia 4 - Alunos na fase em que passavam via sensor para o Robô “Todo o Terreno”, o programa por eles feito**

Posto isto, considero que este projeto decorreu com sucesso, estando já na forja futuras atividades. A realização do mesmo foi gratificante, pois senti o interesse e o *feed-back* de todos os alunos.

## Projetos em Desenvolvimento

---

Atualmente encontro-me inserido num projecto na área do turismo, mais concretamente na implementação de uma central de reservas *online*.

Esta oportunidade proporcionada pela agência de viagens *Good Luck Tours - Viagens Turismo, Lda*, tem sido uma experiência enriquecedora e com boas perspectivas de novos negócios.

A central de reservas que está a ser desenvolvida consiste na integração do sistema através da tecnologia *Web Service*. O *Web Service* é uma solução utilizada na integração de sistemas e na comunicação entre aplicações diferentes, tornando assim possível que novas aplicações possam interagir com aquelas que já existem e que sistemas desenvolvidos em plataformas diferentes sejam compatíveis. Os *Web Services* são componentes que permitem às aplicações enviar e receber dados em formato *XML*. Cada aplicação pode ter a sua própria "linguagem", que é traduzida para uma linguagem universal, o formato *XML*.

Esta tecnologia está a ser disponibilizada pela *Travelport* <sup>[1]</sup>, através do *Galileo Web Services (GWS)*, que é um serviço que permite à agência de viagens *Good Luck Tours - Viagens Turismo, Lda*, comunicar com o sistema distribuído da Galileu (*GDS*) via *XML*, permitindo assim aceder a todos os segmentos de voos existentes no mesmo. Através do *GWS* a agência de viagens pode integrar o conteúdo do *GDS* Galileo com o seu próprio conteúdo por forma a fornecer soluções de voos integradas.

Com a implementação desta plataforma é possível “ligar” a agência de viagens e compradores de todo o mundo, levando o negócio de viagens consigo e para onde quer ir, preparados para ir ao encontro das expectativas dos compradores.

O sistema tem uma particularidade interessante, pois permite pagamentos *online*. A *Redunicre*<sup>[2]</sup> foi o sistema utilizado para efectuar os pagamentos. O cliente após escolher o voo pretendido,

---

[1] A *Travelport* é uma companhia de serviços de dimensão mundial, fornecedora líder de transacções electrónicas, fornecendo soluções para empresas que operam na indústria global de turismo. A multinacional de turismo engloba os sistemas de distribuição global (*GDS*) Galileo.

efectua o pagamento da reserva e automaticamente é-lhe enviado por *e-mail* o bilhete eletrónico, facilitando assim a vida das pessoas.

O portal web da agência de viagens em imagem:

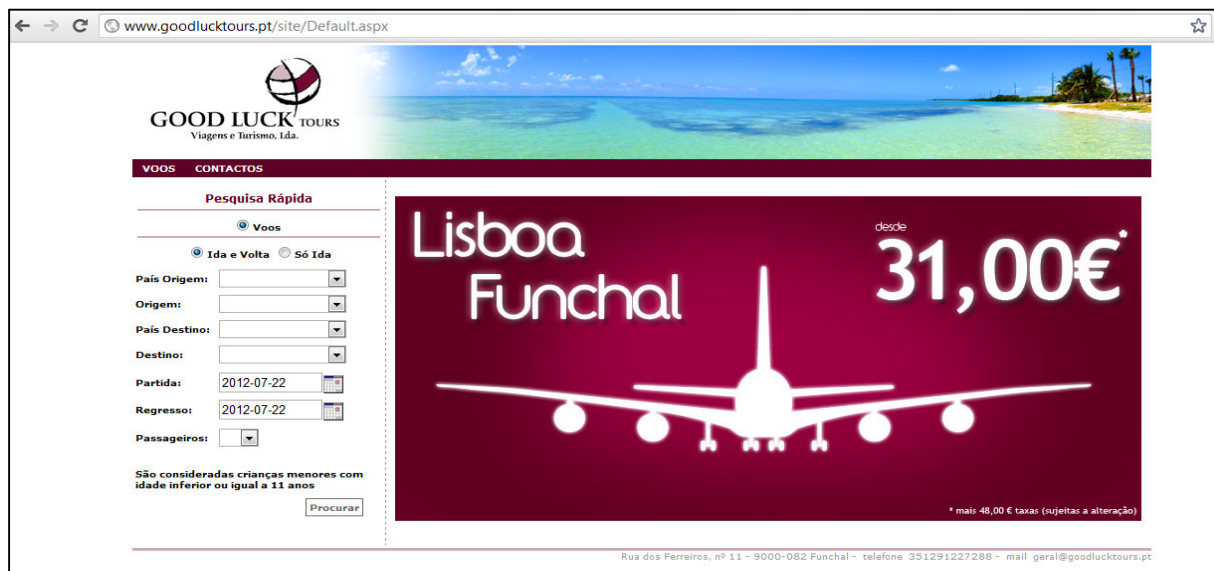


Figura 29 - Portal Web Good Luck Tours

O portal web pode ser acedido no seguinte endereço: <http://www.goodlucktours.pt>

O módulo de voos está praticamente concluído, estando neste momento a ser preparada a implementação de outros módulos, tais como hotéis, *rent-a-cars*, cruzeiros e pacotes turísticos.

Neste momento procuro por soluções relevantes para suportar e fazer crescer o negócio de viagens, aumentar a rentabilidade, reduzir custos e operar de forma eficiente.

O meu objetivo é a médio longo prazo comercializar este produto a outras agências de viagens ou operadores, encontrando o sucesso por ter em cada cliente um verdadeiro parceiro de negócios, tornando-os mais fortes.

---

[2] A Redunicre é a unidade de negócios da UNICRE responsável por disponibilizar ao mercado soluções actuais e outros serviços relacionados com a aceitação de cartões bancários como meio de pagamento.

## **Conclusão**

---

Voltar á universidade foi um desafio, porque foi importante vencer alguns medos, como por exemplo o de lidar com a falta de tempo e algumas rotinas que fazem parte do quotidiano de um aluno universitário. É óbvio que passar novamente pela experiência universitária anos depois da primeira vez é uma situação diferente, pois retornei à universidade com mais experiência de vida e mais maturidade.

Foi para mim um enorme prazer descrever a minha experiência profissional e as diversas áreas onde trabalhei pois esta multiplicidade de contextos permitiu a minha evolução profissional e humana. Com este relatório foi dado a conhecer o meu percurso de carreira no mundo profissional, onde descrevi toda a minha experiência profissional e as opções que fui tomando no desenrolar da mesma. De referir que com a maior experiência profissional que agora detenho, mas que na altura não detinha, muitas das tomadas de decisões poderiam ter sido diferentes.

Tive imensa sorte por ter colaborado com empresas importantes, porque aumentou as minhas competências técnicas e deu-me a possibilidade de pensar que às vezes consigo fazer melhor do que aquilo que penso. Deu-me autoestima. Ainda que sempre de pés bem assentes no chão, e com muita humildade acho que hoje sou uma pessoa mais confiante.

Assim, o facto de ter trabalhado em diversas áreas e em diversos projetos, capacitou-me de competências e de conhecimentos que posso canalizar não só para a gestão de negócios empresarial mas também para o ensino. Esta multiplicidade de contextos permitiu a minha evolução profissional e humana.

A nível pessoal sinto que estas experiências profissionais fomentaram o meu espírito de iniciativa, a capacidade de luta pelas minhas conquistas e a coragem de sobreviver num mundo de trabalho cada vez mais exigente e competitivo.

A nível profissional, proporcionaram-me um desenvolvimento que superou todas as minhas expectativas. Trabalhar em diversos projetos foi fundamental por ter a oportunidade de absorver todos os aspetos positivos que daí advêm. Sinto-me privilegiado por ter trabalhado em

empresas/instituições que encontram-se em forte crescimento e para qual pude dar o meu contributo.

Como se pode constatar ao longo da minha atividade profissional, o empreendedorismo reflete a minha atitude perante a vida, é a minha forma de estar, de tomar a iniciativa, de tentar criar algo inovador, revelando-se uma fonte de riqueza, não apenas individual, mas também coletiva, quer para as empresas onde trabalhei ou que continuo a colaborar ou para a escola onde atualmente ensino, e pelas repercussões que tem a nível económico e dos conhecimentos que traz, contribuindo para o desenvolvimento e crescimento das instituições.

Concluo que não detenho apenas um perfil empreendedor, mas vários, em função da minha experiência profissional. Sou um profissional multifuncional, desenvolvi capacidades para trabalhar em equipa, tento ser capaz de entender as estratégias das empresas/entidades e sou capaz de me autoliderar de maneira a fazer parte dos grandes desafios que as mesmas enfrentam.

Estes são os caminhos que penso que tenho de continuar a percorrer com vista a ter sucesso profissional, assim como a garantia da própria empregabilidade.

## **Bibliografia**

---

[MarquesLopes] – Nuno C. Marques, Gabriel Pereira Lopes “Tagging With Small Training Corpora”

[SNNS] – Stuttgart Neuronal Network Simulator (SNNS) User Manual

- <http://www-ra.informatik.uni-tuebingen.de/SNNS/UserManual/UserManual.html>

[Marques 1999] – Nuno C. Marques “Uma metodologia para modelação estatística da subcategorização verbal”, Dissertação de Doutoramento, Universidade Nova de Lisboa, 1999

[Postgresql] – SGBD (Sistema Gestão de Base de Dados)

- <http://www.postgresql.org/>
- <http://www.commandprompt.com/ppbook/>

[PT Comunicações] – Portugal Telecom Comunicações

- <http://www.ptcontact.pt/marcoshistoricos.htm>

[Rosa, P. 2011] – Paulo Ricardo da Silva Rosa “A Teoria de Vygotsky”, Departamento de Física, UFMS, 2011

- [http://www.dfi.ufms.br/prrosa/Pedagogia/Capitulo\\_5.pdf](http://www.dfi.ufms.br/prrosa/Pedagogia/Capitulo_5.pdf)

[Papert, S. 1986] – Seymour Papert “Constructionism: A New Opportunity for Elementary Science Education”, A proposal to the National Science Foundation, Massachusetts Institute of Technology, Media Laboratory, Epistemology and Learning Group, Cambridge, Massachusetts, 1986

[Papert, S. 1980] – Seymour Papert “Mindstorms: Children, Computers and Powerful Ideas”, Basic Books, New York, Traduzido para o Português em 1985, como Logo:Computadores e Educação, Editora Brasiliense, São Paulo, 1980