DAS Departamento de Automação e Sistemas
CTC Centro Tecnológico
UFSC Universidade Federal de Santa Catarina

Comparação entre técnicas de IA e desenvolvimento de um filtro de mensagens

Monografia submetida à Universidade Federal de Santa Catarina

como requisito para a aprovação da disciplina:

DAS 5511: Projeto de Fim de Curso

Cristian Gunsch Moura

Florianópolis, Julho 2012

Comparação entre técnicas de IA e desenvolvimento de um filtro de mensagens

Cristian Gunsch Moura

Esta monografia foi julgada no contexto da disciplina

DAS 5511: Projeto de Fim de Curso

e aprovada na sua forma final pelo

Curso de Engenharia de Controle e Automação

Banca Examinadora:

Eng. Márcio Bittencourt Pires Jr.
Orientador Empresa

Professor Jomi Fred Hübner, Dr.
Orientador do curso

Professor Hector Bessa Silveira, Dr.
Responsável pela disciplina

Prof. Ricardo José Rabelo, Dr.
Avaliador

Leandro Feltrin Zanellatto
Debatedor

Felipe Santos Eberhardt
Debatedor

1 Agradecimentos

Primeiramente gostaria de agradecer meus pais, que nunca mediram esforços para possibilitar que eu atingisse o meu sucesso, também gostaria de agradecer aos meus irmãos que sempre foram ótimos companheiros e conselheiros durante toda a minha vida.

Gostaria de agradecer ao meu orientador na empresa, Eng. Márcio Bittencourt Pires Jr., que sempre me auxiliou com muita boa vontade no desenvolvimento do projeto e na confecção deste trabalho escrito,também foi de suma importância para o meu desenvolvimento profissional, sempre compartilhando conhecimento e auxiliando no desenvolvimento de tarefas desde o momento que entrei na empresa. Assim também como o professor Jomi Fred Hübner que foi meu orientador da UFSC.

Aos colegas de curso e amigos que sempre estiveram presentes nos momentos de estudos e de lazer, tornando a faculdade algo prazeroso e memorável.

Abstract

This work was carried out in the company Cellmídia, founded and based in Florianopolis, it presents the services offered by the company, forming part of the scope of work.

Is also presented in a superficial way the the theory Artificail Intelligence, as well as some methodologies pertinent to the discussion for the lay reader can understand this area work altogether.

In developing this work is presented the implementation of two classifiers messages, based on the theory of a tree and another decsão Baysianas Network. Following a comparative result between the two methodologies.

Finishing the job we have the results of all the work, also the conclusions and future perspectives for the development to be performed.

Resumo

O presente trabalho foi realizado na empresa Cellmídia, fundada e com sede em Florianópolis, nele é apresentado o serviços oferecidos pela empresa, que fazem parte do escopo do trabalho.

Também é apresentado de forma superficial o a teoria de Inteligência Artificail, bem como os algumas metodologias referentes ao assunto, para que o leitor leigo nessa área possa entender o trabalho por completo.

No desenvolvimento do trabalho é apresentado a implementação de dois classificadores de mensagens, um baseado na teoria de Árvore de decsão e outro em Redes Bayesianas. Seguindo de uma resultado comparativo entre as duas metodologias.

Finalizando o trabalho temos os resultados de todo o trabalho desenvolvido, assim também como as conclusões e perpectivas futuras referentes ao desenvolvimento realizado.

Sumário

Lista de Figuras

Lista de Tabelas

1 Introdução

1.1: A Cellmidia

A empresa Cellmídia Serviços de Informática Ltda está localizada na Rod. SC 401, KM 01, Parque Tec Alfa, Edifício Celta, 4^{o} Andar, Sala 3.09 João Paulo - Florianópolis

A Cellmídia é uma empresa de serviços com foco em mobile marketing e marketing digital multiplataformas que auxilia seus clientes a obterem uma comunicação precisa, agregando novos diferenciais a ações de marketing direto. Atendendo clientes de todo o país de diversificados segmentos, tamanhos e necessidade.

A empresa foi criada a partir da percepção dos seus sócios para a enorme penetração que o telefone celular atingiu na sociedade. O celular vem estabelecendo novos comportamentos e padrões culturais em todo o mundo, participando cada vez mais da vida das pessoas.

A Cellmídia é a única empresa brasileira do segmento de mobile marketing a ter uma base de dados própria e autorizada, com cerca de quatro milhões e meio de pessoas cadastradas de todo o país.

Essa base de dados é formada a partir do CoolSMS, disponibilizado pela empresa na internet, que possibilita o envio de mensagens SMS gratuitamente para todas as operadoras do Brasil. Para usá-lo, o internauta precisa se cadastrar, informando alguns dados pessoais e optando por receber ou não mensagens publicitárias em seu e-mail e celular.

1.1.1: Serviços

1.1.1.1: Cellmídia Center

A Cellmídia desenvolveu um aplicativo chamado *Cellmídia Center*, que se adapta às mais diversa necessidades das empresas que precisam se comunicar com precisão, a qualquer hora e em qualquer lugar.

O *Cellmídia Center* é um aplicativo proprietário para planejamento, criação, execução e avaliação de campanhas de *mobile marketing*. O seu núcleo de envio de mensagens é capaz de entregar grandes quantidades de torpedos pro segundo, podendo ser utilizado por todos os ramos de negócios.

A Cellmídia oferece também a possibilidade de customização do *Cellmídia Center* para a integração com o sistema das empresas, facilitando o dia-a-dia dos profissionais envolvidos com o gerenciamento dos programas de marketing.
Podemos dividir o *Cellmídia Center* em três módulos principais:

1. **Contatos** Aqui o usuário gerencia seus contatos, inserindo números e organizando sua agenda de telefones. De fácil utilização, o cliente pode inserir sua lista de contatos ou contar com o *Cellmídia Database*, com mais de $4,5$ milhões de contatos organizados por diferentes perfís e parâmetros.

2. **Mensagens** Esse móulo atua na etapa de criação e envio de mensagens. De maneira simples, o usuário edita suas mensagens, escolhe seus contatos e determina a data e a hora para o disparo. Além de permitir o agendamento de uma data especifica para efetuar o disparo das mensagens, esse módulo oferece ainda outras funcionalidades:

   Personalização: as mensagens são enviadas com os nomes personalizados da lista de contatos;

   Aniversários: envio de mensagens automáticas para os contatos aniversariantes;

   Promoções: a ferramenta gera um código para cada contato enviado, facilitando o desenvolvimento de ações promocionais que utilizam identificação de códigos;

   Integração com sistemas: através do *Gateway Cellmída*, a ferramenta permite integração tecnológica com outros sistemas de informática, permitindo co-

municação com celulares sem a interação humana para a programação de envio de mensagens.

3. **Reatórios:** Diferentes de outras mídias de marketing direto, a comunicação via SMS do *Cellmídia Center* é monitorada com precisão, oferecendo os parâmentros para a verificação dos resultados em tempo real. Assim, os resultados e o retorno sobre o investimento pode ser avaliado de forma mais prática, clara e completa. A ferramenta de relatórios on-line do *Cellmídia Center* viabiliza o gerenciamento das mensagens enviadas e permite uma análise completa dos resultados. O *Cellmídia Center* disponibiliza os seguintes relatórios:

   - Mensagens programadas;
   - Mensagens enviadas;
   - Taxa de sucesso;
   - Número e as causas de falhas na entrega das mensagens;
   - Respostas que os clientes eventualmente enviam para o *Cellmídia Center* (Interação com o receptor que enviou a mensagen).

1.1.1.2: Cellmídia Database

O *Cellmídia Database* é hoje a maior base de dados de contatos multi-operadoras com *duplo opt-in* do país, com mais de 4,5 milhões de contatos. Essa lista é estratégica para desenvolver campanhas publicitárias, ações de marketing direto, relacionamento e promoções diversas.

A expressão *duplo opt-in* designa que a lista de contatos foi construida seguindo um critério de permissões, em acordo com as políticas de anti-spam. Todos os contatos da Cellmídia autorizam o recebimento de mensagens e ainda confirmam seus dados (telefone celular e e-mail), sem chances de utilizarem dados falsos.

Selecionando o perfil do público-alvo (idade, sexo, localização geográfica) a Cellmídia realiza a contagem de contatos que possuem nessas especificações.

A lista de contatos do *Cellmídia Database* é construida a partir do software *CoolSMS* (www.cool.com.br). A partir dele, os contatos autorizam os envio de mensagens publicitárias para os seus telefones celulares para acumularem pontos no programa de relacionamento do *CoolSMS*.

1.1.1.3: CoolSMS

O *CoolSMS* é um software grautito, distribuido via internet, voltado aos usuários de internet que desejam enviar torpedos SMS para celulares, de maneira simples e fácil.

É hoje o maior software de envio de torpedos grátis da internet brasileira, já que contabiliza mais de 120 mil acessos diários.

O *CoolSMS* é o responsável pela captação do banco de dados de celulares e e-mails do *Cellmídia Database*.

A Cellmídia disponibiliza espaços para publicidade on-linebi *CoolSMS* e no portal Cool.

1.1.1.4: Gateway

A Cellmída oferece uma solução para integração tecnológica com outros sistemas , permitindo comunicação com celulares sem a interação humana para a criação e programação de mensagens. Através de protocolos de comunicação, são disponibilizadas interfaces para outras empresas se conetarem ao *Gateway Cellmídia* e realizarem o envio de mensagens.

1.1.1.5: E-mail Markting

É ainda oferecido o serviço de disparo de e-mails para listas de contatos do cliente ou para os contatos do *Cellmídia Database*, bastanto selecionar o perfil do público-alvo desejado (idade, sexo, localixação geográfica).

O envio de e-mails pode ser acompanhado através de relatórios completos on-line, que indicam o número de e-mail lidos, e-mails com erros, cliques efetuados, entre outras informações relevantes para anãlise de campanhas de e-mail marketing. [2]

1.2: Problemas a serem resolvidos

O Cellmidia Center é utilizado por milhares de pessoas diariamente, com um alto trafego de mensagens. A empresa deseja controlar de alguma forma , dependendo do contexto das mensagens o seu conteúdo.

1.3: Objetivos do trabalho

O objetivo desse trabalho é elaborar um filtro que seja capaz de selecionar mensagens, que dentro de algum contexto, são consideradas imprórpias. Para isso a soluçao deve considerar alguns fatores como o contexto das mensagens e o conteúdo em si da mensagem. Os objetivos específicos são:

- Buscar no banco de dados da empresa mensagens e classificá-las de acordo com o seu conteúdo;
- Projetar um filtro a fim de identificar mensagens que tenha seu conteúdo relacionado a algum contexto impróprio;
- Maximizar o desempenho do filtro para minimizar o trabalho do operador humano em aprovar/reprovar mensagens consideradas impróprias.

1.4: Motivação

Uma das grandes motivações para este trabalho é a falta de uma ferramenta adequada para fazer a seleção das mensagens, que atenda os objetivos descritos acima, e a oportunidade de criar tal filtro, o qual trará grandes resultados para a Cellmídia principalmente na questão de controlar o conteúdo das mensagens de acordo com seu interesse.

Soma-se a isso os conhecimentos que serão adquiridos ao desenvolver esse filtro, principalmente os relacionados à área inteligência artificial e desenvolvimento de software, visto que esses dois tópicos são pouco abordados no curso de Engenharia de Controle e Automação, mas são de suma importância para a formação completa de um engenheiro.

1.5: Metodologia

Existem diversas metodologias para desenvolvimento de software. Cada uma delas tem características especiais e se adéquam melhor a determinadas situações e experiência do programador.

1.5.1: Processo em cascata

Esta é a metodologia mais antiga e mais bem difundida entre os programadores. O processo segue uma série de fases ordenadas (concepção, requisitos, modelagem, codificação e testes), sendo que ao final de cada etapa o projeto segue para a próximo passo sem fazer qualquer alteração no plano inicial .

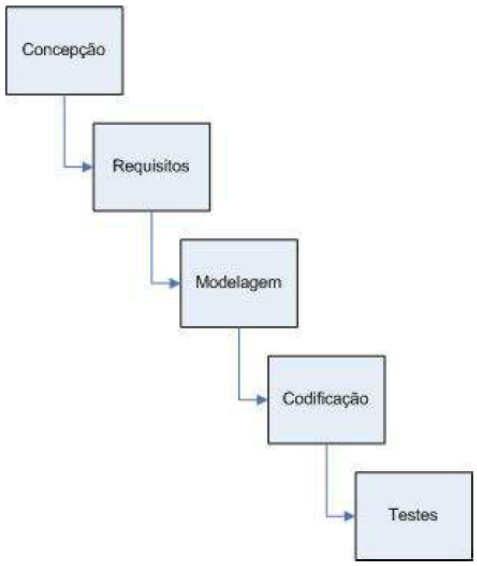


Figura 1.1: Passos da Metodologia em Cascata.

1.5.2: Processo "Codifica-corrige‘

Esta abordagem não chega a seguir uma metodologia formal, pois baseia-se em uma prática de desenvolvimento com ciclos rápidos de codificação seguidos por correção. Periodicamente, o programador apresenta uma nova versão do software ao cliente para obter feedback, e então continua com a codificação [3] (ver figura 1.2).

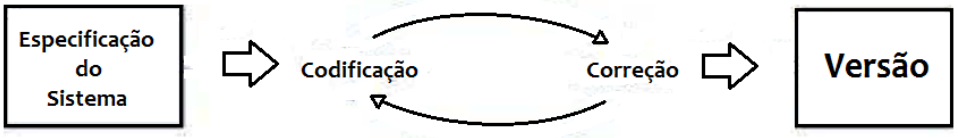


Figura 1.2: Funcionamento do processo "codifica-corrige".

Apesar de ser o método mais utilizado para o desenvolvimento de programas de pequeno porte, normalmente este método faz com que o produto final tenha uma baixa qualidade e escalabilidade, com falta de adaptabilidade, reuso e interoperabilidade. [3]

1.5.3: Processo Unificado Racional

O Processo Unificado Racional – Rational Unified Process (RUP) – é uma metodologia disciplinada de desenvolvimento utilizando um conjunto de ferramentas e modelos. A RUP utiliza em sua concepção a abordagem de orientação a objetos e seu projeto e documentação baseia-se na notação UML (Unified Modeling Language) para ilustrar os processos em ação. Esta metodologia tem foco na redução dos riscos do projeto, além de usar técnicas e práticas aprovadas comercialmente.

O Processo Unificado Racional é dividido em quatro etapas: concepção, elaboração, construção e transição. Na concepção são definidos os requisitos básicos do projeto, analisados os recursos necessários e feito o planejamento do projeto. A fase da elaboração abrange a análise do domínio do problema, estabelecimento da fundação arquitetural e eliminação dos elementos de alto risco.

Na construção é elaborada a modelagem e programação do software. Por fim, a etapa da transição consiste na implantação e manutenção do sistema. [4] As fases de elaboração e construção ocorrem dentro de ciclos iterativos.

Pelas razões expostas acima, além de ser uma metodologia já apresentada em uma das disciplinas da graduação, este processo foi o escolhido para desenvolver o sistema.

1.6: Cronograma

Para que fosse possível verificar o andamento do trabalho e corrigir eventuais atrasos, foi elaborado um cronograma e nele foram inseridas as principais atividades do desenvolvimento do classificador. Baseado na metodologia do Processo Unificado Racional, o plano de atividades foi divido em sete etapas.

A primeira etapa abrange selecionar as mensagens com conteudo inapropriado

A segunda etapa trata do estudo do problema. Nessa fase foram feitas as análises das tecnicas que poderiam ser utilizadas.

A terceira etapa do cronograma trata da concepção do sistema, em que implementados um filtro utilizando Árvore de decisão e um filtro utilizando Redes bayesianas.

A quinta fase do projeto abrange os testes e a validação dos classificadores, para decidir qual deles seria operacionalizado na empresa.

A última etapa do processo compreende a documentação final do projeto e a elaboração do relatório do PFC.

1.7: Contexto com o Curso

O curso de Engenharia de Controle e Automação forneceu conceitos importantes para o entendimento do problema e formulação da solução, principalmente no que diz respeito a:

- Inteligência artificial, que forneu a base para a concepção da solução.
- Engenharia de *software*, que serviu de base para organização, documentação e implementação do projeto;
- Banco de dados, que forneceu conhecimentos importantes para a criação de tabelas e processamento e armazenamento de informações.

2 Tecnicas de Classificação

2.1: Introdução

A inteligência tem sido matéria de estudo dos seres humanos por mais de 2000 anos. O sonho do desenvolvimento de uma máquina pensante tem sido fonte de pesquisa desde o desenvolvimento das primeiras máquinas computadas, em meados dos anos 40. A união destas duas áreas de pesquisa deu origem ao que se denomina, Inteligência Artificial. [5]

Abaixo, a figura 2.1 representa as várias definições de Inteligência Artificial:

	Sucesso em termos de desempenho humano	Conceito ideal de inteligência
Pensamento e raciocínio	Sistemas que pensam como o homem(I)	Sistemas que pensam racionalmente (II)
comportamento	Sistemas que agem como o homem(III)	Sistemas que agem racionalmente (IV)

Figura 2.1: Definições de IA. [1]

I - “A automação das atividades que nós associamos com o pensamento humano, atividades tais como tomada de decisão, resolução de problemas, aprendizagem, ...” (Bellman, 1978)

II - “O Estudo das computações que tornam possível a percepção, raciocínio, e ações” (Winston, 1992)

III - “O estudo da construção de computadores que podem executar tarefas que, neste momento, as pessoas são mais capazes” (Rich and Knight, 1991)

IV - “O ramo da Ciência da Computação que se preocupa com a automação do comportamento inteligente” (Luger and Stubblefield, 1993) [1]

Atualmente, estudos em inteligência artificial podem ser divididos em duas grandes áreas: o desenvolvimento de sistemas que agem como humanos (robôs) e o desenvolvimento de sistemas que agem racionalmente. Dentro do contexto dos sistemas que agem racionalmente, duas abordagens principais podem ser utilizadas: raciocínio lógico e raciocínio probabilístico. O raciocínio lógico pondera sobre o conhecimento prévio a respeito do problema e, sobre esta base de conhecimento retira suas conclusões. Esta abordagem, apesar de poderosa, pode não ser útil em situações onde não se conhece previamente todo o escopo do problema, para estes casos, o raciocínio probabilístico surge como uma boa opção. [5]

Um sistema que possa atuar em situações de incerteza deve ser capaz de atribuir níveis de confiabilidade para todas as sentenças em sua base de conhecimento, e ainda, estabelecer relações entre as sentenças. Redes bayesianas oferecem uma abordagem para o raciocínio probabilístico que engloba teoria de grafos, para o estabelecimento das relações entre sentenças e ainda, teoria de probabilidades, para a atribuição de níveis de confiabilidade. [5]

Boa parte das aplicações de relevãncia prática em inteligência artificial está basea da na concepção de modelos computacionais do conhecimento empregado pra um especialista humano. Na sintese de modelos de classificação , a associação entre categorias e o conjutnos de atributos que caracterizam os objetos a serem classificados pode se dar de formas variadas, empregando processamento simbólico ou numérico. [6]

A contrução de modelos computacionais de classificação geralmente emprega um dentre dois paradigmas alternativos:

- Top-down: obtenção do modelo de classificação a partir de informações fornecidas por especialista;
- Botton-up: obtenção do modelo de classificaçõa pela identificação de relacionamentos entre variáveis dependentes e independentes em bases de dados rotuladoas. O classificador é induzido pro mecanimos de generalização fundamentados em exemplos especificos (conjunto finito de objetos rotulados). Existem propostas também para dados não rotulados.[6]

2.1.1: Apredizagem

O principio fundamental do conceito de inteligência consite na capacidade de aprender, tornando um agente mais flexivel, o permitindo lidar com situações novas, dar autonomia ao agente.

Portanto, aprendizado é o processo de modificação dos parâmetros do agente, de modo a maximizar uma medida de desempenho.

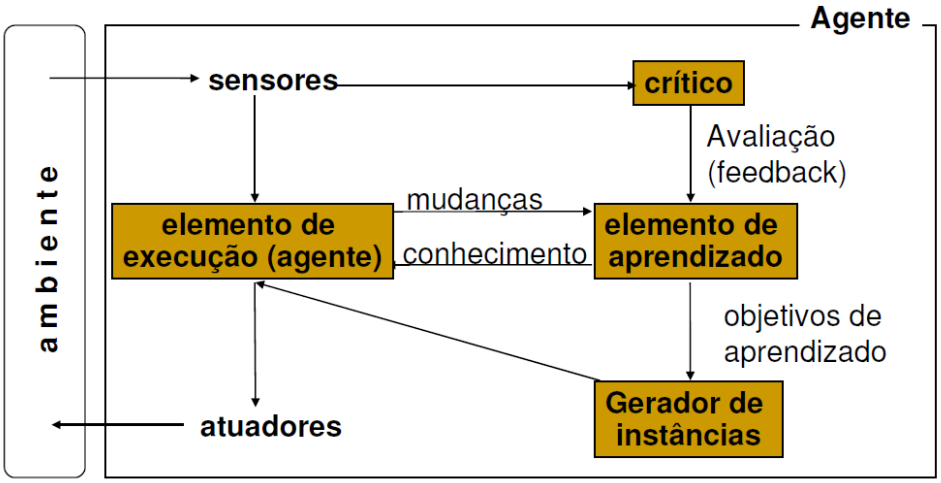


Figura 2.2: Modelo geral de apredizagem.

Podemos classificar o processo de aprendizagem em 3 tipos diferentes, que são:

Aprendizado supervisionado: O crítico comunica o elemento de aprendizado o erro relativo entre a ação que deve ser tomada idealmente pelo elemento de execução e a ação efetivamente escolhida pelo agente. Pares (corretos) de entrada/saída podem ser observados (ou demonstrados por um supervisor).

Aprendizado não-supervisionado: O crítico não envia nenhum tipo de informação ao elemento de aprendizado, não há “pistas” sobre as saídas corretas (geralmente utiliza-se regularidades, propriedades estatísticas dos dados sensoriais).

Aprendizado por reforço: O crítico comunica apenas uma indicação de desempenho (geralmente, indicação de quão bom ou ruim é o estado resultante), normalmente de modo intermitente e apenas quando situações dramáticas são atingidas.

2.2: Classificadores baseados em árvores de decisão

Árvores de decisão são similares a regras if-then. É uma estrutura muito usada na implementação de sistemas especialistas e em problemas de classificação. As árvores de decisão tomam como entrada uma situação descrita por um conjunto de atributos e retorna uma decisão, que é o valor predizido para o valor de entrada. Os atributos de entrada podem ser discretos ou contínuos. Para os exemplos tratados, serão considerados apenas valores discretos. O aprendizado de valores discretos é chamado classificação.

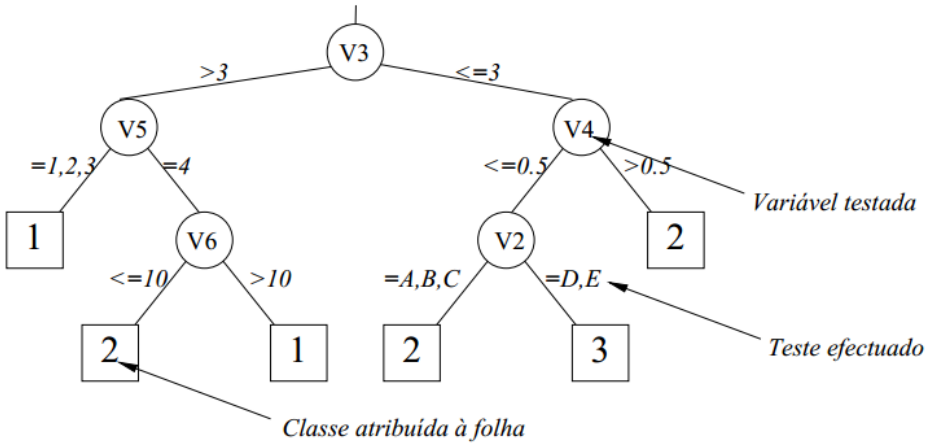
A árvore de decisão chega a sua decisão pela execução de uma seqüência de testes. Cada nó interno da árvore corresponde a um teste do valor de uma das propriedades, e os ramos deste nó são identificados com os possíveis valores do teste. Cada nó folha da árvore especifica o valor de retorno se a folha for atingida.

Para qualquer problema de árvore de decisão, deve-se inicialmente definir atributos disponíveis para descrever exemplos de possíveis casos do domínio.

Os classificadores baseados em árvores de decisão remontam aos anos 50, sendo uma referência fundamental o trabalho de Hunt, onde vários trabalhos de indução são apresentados [Hunt, 1966]. Posteriormente, talvez o trabalho mais importante seja o extraordinário livro de Breiman, Friedman, Olshen e Stone, em que o algoritmo CART é apresentado [Breiman, 1984]. Também o trabalho de Quinlan [Quinlan, 1986] teve grande aceitação nesta área científica tendo servido de inspiração a muitos dos sistemas posteriormente apresentados e em particular ao seu trabalho C4.5 [Quinlan, 1993].

A filosofia de funcionamento de qualquer algoritmo baseado em árvores de decisão é bastante simples. Na verdade, embora contendo por vezes diferenças importantes na forma de efectuar este ou aquele passo, qualquer algoritmo desta categoria se baseia na estratégia de dividir para conquistar. De uma forma geral, esta filosofia baseia-se na sucessiva divisão do problema em vários subproblemas de menores dimensões, até que uma solução para cada um dos problemas mais simples possa ser encontrada. Fundamentados neste princípio, os classificadores baseados em árvores de decisão procuram encontrar formas de dividir sucessivamente o universo em vários subconjuntos (criando para tal nós contendo os testes respectivos) até que cada um deles contemple apenas uma classe ou até que uma das classes demonstre uma clara maioria não justificando posteriores divisões (gerando nessa situação uma folha

contendo a classe maioritária). Como é evidente, a classificação consiste apenas em seguir o caminho ditado pelos sucessivos teste colocados ao longo da árvore até que seja encontrada uma folha que conterá a classe a atribuir ao novo exemplo. [7]



- Cada nó de decisão contem um teste num atributo.
- Cada ramo descendente corresponde a um possível valor deste atributo.
- Cada Folha está associada a uma classe.
- Cada percurso na arvore (da raiz à folha) corresponde a uma regra de classificação.

Embora a filosofia de base de todos os classificadores baseados em árvores de decisão seja idêntica, muitas são as possibilidades existentes para a sua construção. De entre todos os aspectos determinantes na seleção de um algoritmo de construção de árvores de decisão alguns deles devem ser destacados pela sua importância:

- Critério para a escolha da característica a utilizar em cada nó;
- Como calcular a partição do conjunto de exemplos;
- Quando decidir que um nó é uma folha;
- Qual o critério para seleccionar a classe a atribuir a cada folha.

Para além destes aspectos, muitos outros tais como a aplicação de janelas sobre o conjunto de treino e a aplicação de processamentos de redução das árvores designados como processos de poda, são determinantes no desempenho final deste tipo de sistemas. Algumas vantagens importantes podem ser apontadas às árvores de decisão, nomeadamente:

- Podem ser aplicadas a qualquer tipo de dados;
- A estrutura final do classificador é bastante simples podendo ser guardada e manipulada de uma forma bastante eficiente;
- Manipula de uma forma muito eficiente informação condicional subdividindo o espaço em sub-espaços que são manipulados individualmente;
- Revelam-se normalmente robustos e insensíveis a erros de classificação no conjunto de treino;
- As árvores resultantes são normalmente bastante compreensíveis podendo ser facilmente utilizadas para se obter uma melhor compreensão do fenómeno em causa. Esta é talvez a mais importante de todas as vantagens enunciadas.

2.2.1: Algoritmos de indução de árvores de decisão

Será apresentado o algoritmos de indução de árvores de decisão mais utilizado.

2.2.1.1: ID3

Árvore de Decisão Indutiva é um dos métodos de aprendizado simbólico mais amplamente utilizados e práticos para inferência indutiva. É um método para aproximar funções discretas robustas a dados com ruído e que permite o aprendizado de expressões disjuntas. É descrito um algoritmo extensamente estudado, o ID3, o qual dá preferência às árvores pequenas, evitando árvores grandes. Esta característica faz uma espécie de generalização sobre os exemplos de aprendizado.

2.2.1.2: Algoritmo ID3

2.3: Redes Baysianas

2.3.1: Introdução

Usando a teoria das probabilidades, podemos determinar, frequentemente a partir de um argumento *a priori*, as chances de ocorreem eventos. Podemos descrever, também, como combinações de eventos são capazes de se influenciar mutamente. A idéia que suporta a teoria das probabilidades é que podemos entender a frequência

```
ID3(Exemplos, Atributo-objetivo, Atributos)
    //  ID3 retorna uma árvore de decisão que classifica corretamente os Exemplos determinados
    //  Exemplos são os exemplos de treinamento.
    //  Atributo-objetivo é o atributo cujo valor deve ser predito pela árvore.
    //  Atributos são uma lista de outros atributos que podem ser testados pela árvore de decisão.
    Início
        Crie um nodo Raiz para a árvore
        Se todos os Exemplos são positivos
            Então retorna a Raiz da árvore com o rótulo = sim
        Se todos os Exemplos são negativos
            Então retorna a Raiz da árvore com o rótulo = não
        Se Atributos for vazio
            Então retorna a Raiz da árvore com o rótulo = valor mais comum do Atributo-objetivo em Exemplos
            Senão
              A ← um atributo de Atributos que melhor classifica Exemplos  (atributo de decisão)
              Raiz ← A   (rótulo = atributo de decisão A)
              Para cada possível valor v_i de A  faça
                 Acrescenta um novo arco abaixo da Raiz, correspondendo à resposta A = v_i
                 Seja Exemplos_vi o subconjunto de Exemplos que têm valor v_i para A
                 Se Exemplos_vi for vazio
                    Então acrescenta na extremidade do arco um nodo folha
                                  com rótulo = valor mais comum do Atributo-objetivo em Exemplos
                    Senão acrescenta na extremidade do arco a sub árvore
                              ID3(Exemplos_vi, Atributo-objetivo, Atributos -{A})
        Retorna Raiz  (aponta para a árvore)
    Fim
```

Figura 2.3: Algoritmo ID4.

com que eventos ocorrem e usar esta informação para raciocinar sobre as frequências de futuras combinações de eventos. Há uma série de situações em que a análise de probabilidades é apropriada. Quando os eventos não são necessariamente aleatórios, muitas vezes é impossível conhecer e medir, suficientemente bem, todas as causas e suas interações, para que se possa prever eventos. As correlações estatísticas são um substituto útil para esta análise causal.

2.3.2: Raciocínio Bayesiano

O raciocínio bayesiano é baseado na teoria formal das probabilidades e é usado, extensivamente, em várias áreas atuais de pesquisa, incluindo reconhecimento de padrões e classificação. Assumindo um amostragem aleatória de eventos, a teoria de Bayes suporta o cálculo de probabilidades mais complexas a partir de resultados conhecidos previamente. Na teoria matemática das probabilidades, as probabilidades individuais são calculadas analiticamente„ através de métodos combinatórios, ou empiricamente.

- **Probabilidade *a priori*:** A probablidade a priori, também chamanda de probablildade incondicional, de um evento é a probablildade atribuida a um evento na ausência de conhecimento que suporte a sua ocorrência ou ausência, isto é, a

probabilidade do evento anterior a qualquer evidência. A probablidade a priori de um evento é simbolizada por $P(evento)$.

- **Probabilidade *posterior*:** A probabilidade posterior (após o fato), também chamada de probabilidade condicional de um evento é a probabilidade de um evento dad alguma evidência. A probablidade posterior é simbolizada por $P(evento|evidncia)$.

2.3.3: Conceitos Básicos de Probabilidade

Para um melhor entendimento do conteúdo do trabalho, é apresentado um resumo dos principais conceitos e métodos de análise das variáveis aleatórias.[8]

2.3.3.1: Experimento aleatório

Um experimento aleatório é aquele no qual o resultado varia de modo imprevisível, quando é repetido nas mesmas condições [9].

2.3.3.2: Variável aleatória

É uma função que associa um número real *X(ξ)* a cada aparecimento ξ no espaço de amostragem do experimento aleatório [9]. É comum representar uma variável aleatória por uma letra maiúscula (como *X, Y, W*) e qualquer valor particular da variável aleatória por uma letra minúscula (tal como *x, y, w*).

Uma variável aleatória pode ser considerada uma função que mapeia todos elementos do espaço de amostragem (coisas) nos pontos da linha real (números) ou alguma parte dela [9]. Mais de um ponto do espaço amostral pode ser mapeado em um mesmo valor da variável aleatória.[8]

2.3.3.3: Probabilidade

Para definir probabilidade pode-se analisar o experimento de se jogar um dado (cubo com 6 faces numeradas) e observar o número que aparece em sua face superior. Existem seis números que podem ser o resultado. Podese assim definir dois conjuntos para este experimento: o conjunto de todos os possíveis resultados e o conjunto das possibilidades de ocorrência dos resultados. O conjunto de todos os possíveis resul-

tados é chamado de espaço amostral, simbolizado como S. Todo experimento possui o seu espaço amostral.

Um evento é definido como um subconjunto do espaço amostral. No exemplo de se jogar um dado, podese definir o evento “resultar em um número ímpar”. Este evento é um conjunto com três elementos.

Para cada evento definido em um espaço amostral S, deseja-se atribuir um número não negativo chamado probabilidade. Probabilidade é uma função dos eventos definidos. A notação adotada é *P(A)*, para “a probabilidade de ocorrência do evento A”. Dois axiomas importantes dizem que: $P(A) \geq 0$ e $P(S) = 1$, ou seja, a probabilidade de ocorrência de qualquer evento é sempre maior que zero (e menor que 1) e a probabilidade de ocorrência de um evento definido como sendo o espaço amostral S é sempre 1 [9]. Em termos gerais, a probabilidade de ocorrência de um evento A, $P(A)$, será igual ao número de ocorrências do evento $A(n_a)$ dividido pelo número de ocorrências total (N), do espaço amostral (Equação 1).[8]

$$P(A) = \frac{n_a}{N} \tag{1}$$

2.3.3.4: Probabilidade condicional

Define-se a probabilidade condicional de um evento A tendo ocorrido um evento B (com probabilidade diferente de zero) como sendo [9] mostrado pela Equação 2.2.[8]

$$P(A|B) = \frac{P(A \bigcap B)}{P(B)} \tag{2}$$

em que $P(A \bigcap)$ é a probabilidade de ocorrência simultânea dos eventos A e B, isto é, probabilidade conjunta de A e B (também descrita simplesmente como $P(A, B)$).

Para $P(A) \neq 0$, pode-se escrever também:

$$P(B|A) = \frac{P(B \bigcap A)}{P(A)} \tag{3}$$

Combinando as Equações 2 e 3 tem-se a principal forma do teorema de Bayes mostrados na Equação 4.

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)} \tag{4}$$

2.3.3.5: Independência estatística

Dois eventos (A e B) são estatisticamente independentes se a probabilidade da ocorrência de um evento não é afetada pela ocorrência do outro evento. Matematicamente, isto é descrito pelas Equações 5 e

$$6.$$

$$P(B|A) = P(B) \tag{5}$$

$$P(A|B) = P(a) \tag{6}$$

Independência também significa que a probabilidade da ocorrência conjunta (interseção) de dois eventos deve ser igual ao produto das probabilidades dos dois eventos (Equação 7).

$$P(A \bigcap B) = P(A)P(B) \tag{7}$$

2.3.4: Regra de Bayes

Suponha que se conheça a probabilidade prévia (a priori) $P(w_j)$ e a densidade condicional$p(x|w_j)$ para $j = 1, 2$. A densidade de probabilidade conjunta de se encontrar um padrão que é da categoria w_j e possui característica de valor x (ou seja:$p(w_j, x)$), pode ser escrito de duas maneiras, mostradas na Equação 8

$$p(w_j, x) = P(w_j|x)p(x) = p(x|w_j)P(w_j) \tag{8}$$

Rearranjando a Equação **??**, obtem-se a chamada fórmula de Bayes (Equação 9).

$$P(w_j|x) = \frac{p(x|w_j)P(w_j)}{p(x)} \tag{9}$$

em que para o caso de duas categorias:

$$p(x) = \sum_{j=1} 2p(x|w_j)P(w_j) \tag{10}$$

A fórmula de Bayes pode ser expressa informalmente em palavras como:

$$posterior = \frac{verossimilhanaXprvio}{evidncia} \tag{11}$$

A fórmula de Bayes mostra que observando o valor de x pode-se converter a probabilidade a *priori* $P(w_j)$ para a probabilidade a *posteriori* $P(w_j|x)$ - a probabilidade do estado natural de w_j, dado que o valor x da característica tem sido medido. Chama-se $p(x|w_j)$ a verossimilhança de w_j com respeito a x, um termo escolhido para indicar a categoria w_j para qual $p(x, w_j)$ é maior e mais parecida para ser a categoria verdadeira.[8]

O fator evidência, $p(x)$, pode ser visto meramente como fator de escala que garante que a probabilidade posterior soma para 1.

Tendo-se uma observação x para qual $P(w_1|x)$ é maior que $P(w_2|x)$, poderia-se naturalmente ser inclinado a decidir que o estado natural real é w_1 . Quando observa-se um x particular, a probabilidade do erro de classificação é dada pela Equação **??**.

$$P(erro|x) = \begin{cases} P(w_1|x) \text{ se decidir por } w_2 \\ P(w_2|x) \text{ se decidir por } w_1 \end{cases}$$

Para minimizar a probabilidade do erro, a regra de decisão de Bayes torna-se:

- "Decida w_1 se $P(w_1|x) > P(w_2|x)$, senão decida w_2"

Então:

$$P(erro|x) = min[P(w_1|x), P(w_2|x)] \tag{12}$$

Ou a seguinte de decisão equivalente:

- "Decida w_1 se $P(x|w_1) > P(x|w_2)$, caso contrário w_2."

2.4: Estimação da qualidade de um classificador

A estimação da qualidade de um classificador é sem dúvida um problema de grande importância. Dado um classificador como estimar a percentagem de erro que se espera ele venha a obter na classificação de exemplos futuros? Pode definir-se o erro de um classificador como sendo 13.

$$Procentagemdeerro = \frac{Nmerodeerro}{Nmerodecasostestasos} \tag{13}$$

Para conhecermos o valor do erro real do classificador R*(d) com exactidão teríamos de testar o classificador com todos os exemplos possíveis. Este processo não é, no entanto, realístico pois em casos normais é impraticável a obtenção de todos os valores do universo em causa, tendo-se geralmente disponíveis apenas um reduzido número de exemplos. Teremos então de estimar este valor tendo o cuidado de não efectuar um cálculo viciado, isto é, demasiado optimista ou demasiado pessimista. Um outro problema que se coloca na estimação da qualidade de um classificador prende-se com os custos do erro. Na verdade, em muitas aplicações existem diferentes custos associados aos vários erros possíveis. Como exemplo, podemos citar o caso de um sistema de diagnóstico médico no qual é geralmente considerado muito mais grave a classificação de um doente como saudável, do que a classificação de um indivíduo saudável como doente.

2.4.1: Estimação do custo

O custo de um erro pode ser encarado como a penalização imposta ao sistema no caso deste cometer um dado tipo de erro. Se o objectivo do nosso sistema for a minimização dos custos em lugar da minimização do erro teremos de definir as penalizações a atribuir. É normalmente utilizada para este efeito a chamada matriz de custos que contém o custo de cada tipo de erro possível.

2.4.1.1: Estimação por resubstituição ou erro aparente

Neste método [Breiman, 1984], depois do classificador ser construído, todo o conjunto de treino é por ele classificado.

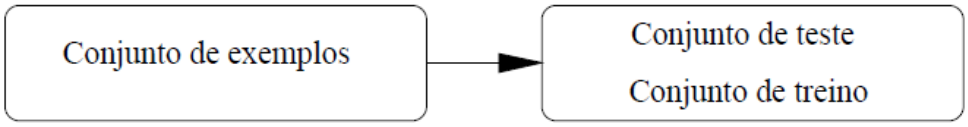


Figura 2.4: Relação entre o conjunto de treino e o conjunto de teste.

A proporção de resultados incorretos ditará o valor estimado para o erro de classificação. O erro de resubstituição pode portanto ser dado por 14.

$$R(d) = \frac{1}{N} \sum_{n=1}^{N} X(d(x_n) \neq j_n) \tag{14}$$

em que:

N - número de exemplos;
x_n - exemplo n
j_n - classe correspondente ao exemplo n
d(x) - resposta do classificador perante o exemplo x

$$X(A) = \begin{cases} 1 \text{ se A é verdadeiro} \\ 0 \text{ se A é falso} \end{cases}$$

Dado que é usado para a estimação do erro o mesmo conjunto de exemplos que foi usado para o cálculo do classificador surge um problema óbvio com este método. Como qualquer classificador é construído com vista a minimizar o erro obtido com o conjunto de treino, esta estimação é claramente optimista. Como exemplo extremo, suponha a situação em que d(X) constitui uma partição A_1,...,A_j tal que A_j contenha todos os exemplos para os quais $j_n = j$ e que os vetores não presentes no conjunto de treino estejam distribuídos aleatoriamente por todas as classes. Como é evidente, neste caso teremos R(d)=0 que estará certamente distante do valor real R*(d).

2.5: Distribuição de Gauss

A distribuição de Gauss será usada para determinar o valor de corte das mensagens classificadas pelo método de Rede Baysianas, que consequentemente definirá as mensagens adequadas e as inadequadas. Quando aplicamos o método, o resultado final é uma probabilidade, e devemos definir algum parametro que arbitre sobre a decisão.

A distribuição Gaussiana (normal) foi historicamente a chamada lei de erros. Foi usado por Gauss para modelar erros de observações astronômicas, que é por isso que é normalmente referido como a distribuição de Gauss. A função de densidade de probabilidade para a distribuição de Gauss padrão (média 0 e desvio padrão 1) e da distribuição de Gauss, com μ médios e desvio padrão σ é dado pelas seguintes fórmulas.

$$\phi(x) = \frac{1}{\sqrt{2\pi}} e^{\left(-\frac{x^2}{2}\right)}$$

$$\phi(x;\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)}$$

$$\phi(x;\mu,\sigma) = \int_{-\infty}^{z} \phi(x)dx$$

$$\phi(x;\mu,\sigma) = \int_{-\infty}^{z} \phi(x;\mu,\sigma)dx$$

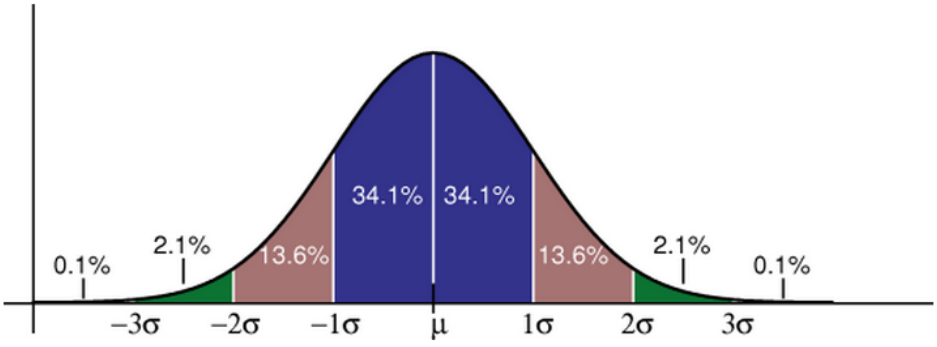


Figura 2.5: Gráfico da distribuição de Gauss

3 Implementação

Neste capítulo será abordada a implementatão de filtros utilizando cada uma das tecnícas de IA apresentadas para a solução do problema, estas são: Árvore de decisão e Redes bayesianas. Será mostrado como foi realizada a implentação e as ferramentas utilizadas para tal. Ao final do capitulo encontra-se uma comparação entre os resultados obtidos entre cada método, e a escolhe da melhor solução para o resolução pro problema proposto nesse trabalho.

3.1: Considerações gerais

Primeiramente foram definidos os contextos de mensagens que a empresa desejaria ter controle. A empresa deseja filtrar 4 tipo de mensagens com contexto especificos, isso porque a Cellmídia é uma empresa de marketing, e em algumas situação certos contextos são aceitáveis para a realização de divulgação. As mensagens que possuem o contexto considerados relevantes são:

- Mensagens homofóbicas;
- Mesangens racistas;
- Mensagens contendo informação sobre propaganda ou promoções;
- Mensagens de Bad Word's (baixo calão)

Posteriormente foi feito um levantamento das palavras que possivelmente poderiam estar associadas a cada um dos contextos de mensagens que seriam classificadas. Essas palavras chave foram selecioandas empiricamente. Tendo essas palavras, realizou-se consultas no banco de dados da empresa buscando mensagens que contivessem tais palavra. Essas mensagens serviram para o treinamento da rede e também para os testes.

No banco de dados, que contem aproximadamente 27 milhões de mensagens, foram selecionadas 390 mensagens considerada homofóbicas, 390 racistas, 680 comerciais e 730 contendo conteúdo com bad word. Essa mensagens foram inseridas em 4 tabelas MySQL. As tabelas são: mensagens_bad_works; mensagens_homofobicsa; mensagens_comerciais; mensagens_racistas.

Para realizar a implementação foi necessário também buscar mensagens genéricas que tivesse o conteúdo considerado adequado. Foram selecionada 498 mensagens, e estas foram importadas para as tabelas de mensagens. Optou-se por utilizar as mesmas mensagens consideradas adequadas, para, ao final gerar resultados mais padronizadoss.

Apesar da seleção das mensagens no banco ter sido feita utilizando palavras chaves, foi necessário realizar uma refino mais precisa. Pois mesmo uma frases possuindo uma determinada palavra, a frase poderia ter o contexto geral adequados aos moldes proposto. Esse processo de classificação foi realizado manualmente com o auxilio de uma interface feita em PHP e HTML e que pode ser observada na figura abaixo:

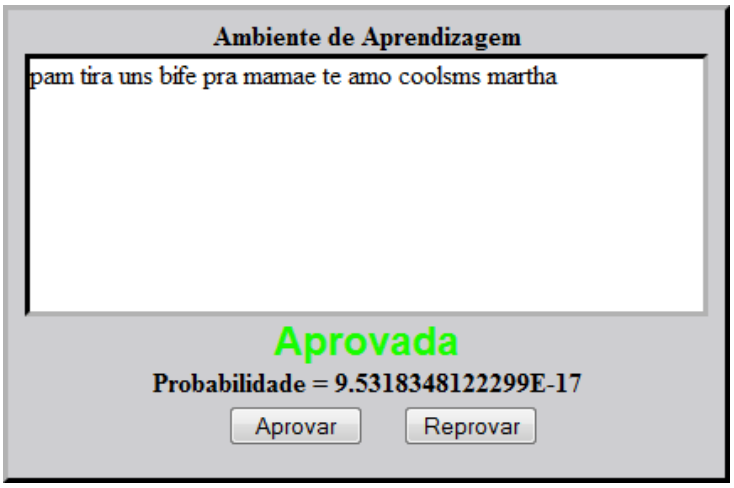


Figura 3.1: Ambiente de seleção e treinamento.

Ne processo de refino na classificação das mensagens era efetuadas outras duas funções. Primeiramente ocorre a normalização das mensagens, que consiste no processo de converter dos os caracteres para minúsculo, retirar pontuações, acentos, espaços extras e outros caracteres julgados não relevantes. A segunda operação consiste em guardar as palavras contidas na mensagem em tabelas MySQL. Como estamos separando as mensagens em 4 contextos diferentes, consequentemente precisamos alocar essas palavras em 4 tabelas distintas também. A estrutura dessas tabelas

são: str_mensagem, int_n_adequada, float_probabilidade_adequada, int_n_inadequada, float_probabilidade_inadequada.

Um mensagem é composta po um conjunto de palavras, e para a implementação do classificador foi necessário realizar esse desmembramento. como base para o treinamento dos métodos. Essa seleção foi realizada no banco de dado do *Cellídia Database*.

Foram criadas 8 tabelas, sendo 4 contendo as mensagens que são:

- mensagens_bad_works;
- mensagens_homofobicsa;
- mensagens_comerciais;
- mensagens_racistas.

E foram criadas também 4 tabelas que foram populadas com as palavras contidas nas mensagens, as tabelas de palavras são:

- palavras_bad_works;
- palavras_hoomofobicas;
- palavras_comerciais;
- palavras_racistas.

Explicar estrutura de cada tabela

3.2: Ferramentas

A Cellmídia adota, para o desenvolvimento o acrônimo LAMP, que corresponde as seguintes ferramentas:

- Linux;
- Apache;
- MySQL;

- PHP.

A combinação dessas tecnologias é bastante popular devido ao baixo custo de aquisição (Software Livre) e também pela performance e escalabilidade. Um outro bom motivo para adoção do LAMP é a facilidade de trocar de servidor já que a grande maioria dos serviços de hospedagem contam com estes softwares. Isso para outras linguagens e bancos de dados normalmente é um inconveniente, tendo em vista as diferentes configurações e restrições dos servidores.

3.2.1: Linux

A escolha do Linux, em relação aos outros sistemas operacionais, também está associado ao fato de ser um SO livre, mas principamente pelo fato de a maioria das bibliotecas disponíveis para a programação em PHP estarem para esse SO.

3.2.2: Apache

È o servidor web livre mais bem sucedido, foi criado em 1995. Um servidor web é um programa de computador responsável por aceitar pedidos HTTP de clientes, geralmente navegadores, e servi-los com respostas HTTP, incluindo opcionalmente dados, que geralmenete são páginas web, tais como documentos HTML com objetos embutidos (imagens, etc.).[10]

3.2.3: MySQL

O MySQL é um sistemas de gerenciamento de banco de dados (SGBD), que utiliza a linguagem SQL (Linguagem de Consulta Estruturada, do inglês Structured Query Language) como interface. É atualmente um dos bancos de dados mais populares, com mais de 10 milhões de instalações pelo mundo.[11]

Principais características:

- Portabilidade (suporta praticamente qualquer plataforma atual);
- Compatibilidade (existem drivers ODBC, JDBC e .NET e módulos de interface para diversas linguagens de programação;

- Excelente desempenho e estabilidade;
- Pouco exigente quanto a recursos de hardware;
- Suporta Stored Procedures e Functions;
- Interfaces gráficas (MySQL Toolkit) de fácil utilização cedidos pela MySQL Inc.

3.2.4: PHP

PHP é uma linguagem de programação interpretada livre e utilizada para gerar conteúdo web.[12]

Principais caracterísitcas:

- Velocidade e robustez;
- Estruturado e orientação a objetos;
- Portabilidade - independência de plataforma - escreva uma vez, rode em qualquer lugar;
- Tipagem dinâmica;
- Sintaxe similar a C/C++ e o Perl;
- Open-source.

3.3: Implementação utilzando Árvore de Decisão

Para a resolução do problema utilizando essa metodologia foi necessário reprocessar os dados, para deixar de uma forma compatível para a aplicação do método. Tendo em posse das mensagem foi feita a seguinte processamento. As frases foram lidas pelo script, se a mensagem tiver um contexto próprio as palavras dessa mensagem são adicionadas a uma tabela e é somado o valor da sua ocorrência referente a mensanges boa, no caso de uma mensagem reprova, essa palavras são adicionadas na mesma tabela, porem é somado o quantidade de sua ocorrencia na coluna de referente ao numero de vezes que ele a palavra estava em mensagen inapropriadas. Dessa forma, uma palavra pode ser divida tento sido encontrada somente com o contexto adequado, sendo encontrada nos dois contexto.

Tabela 3.1: Tabela diagrama de Árvore de Decisão

Palavras	Só-boa	Só-ruim	Reprovada
oi	sim	não	não
Jão	sim	sim	não
tudo	sim	sim	não
bem	sim	não	não
caraca		sim	não

Analisando a entropia de das frases percebeu-se que quando a mensagem continham alguma palavra que estava na apenas na tebelas das numero de ruim a frases é considereda impropria. Portanto,se na mensagem contiver alguma palavra que esteja apenas associadas a frases improprias, a mensagem é considerada impropria.

3.3.1: Resultados

Tabela 3.2: Tabela com os resultados obtidos utilizando Redes Bayesianas

Contexto	Quantidade Total de mensagens	Quantidade total de acertos	%
Bad works	725	335	46,20%
Comerciaiss	680	284	41,76%
Homofobicas	390	204	53,07%
Racistas	390	199	51,02%

3.4: Implementação utilizando Redes Baysianas

Foi desenvolvido um filtro utilizando o classificador de Naive Bayes. A estrutura de Naive Bayes, é mostrada na figura a serguir:

Em que:

- C é a variável de classificação;
- A_1, A_2, A_n são independentes entre si. Representam as variáveis representativas e importantes para o mecanismo de classificação (palavras de mensagens).

Pelo teorema de Bayes

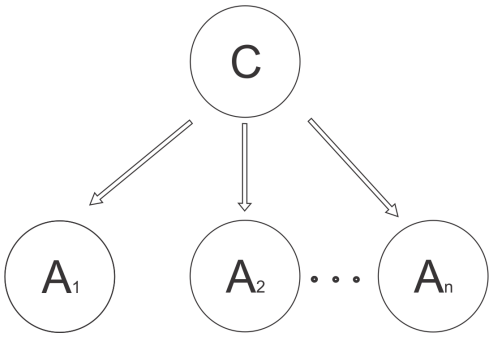


Figura 3.2: Estrutura da rede

$$p(F_1, ..F_n) = \frac{p(C).p(F_1, ..., F_n|C)}{p(F_1, ..., F_n)} \tag{15}$$

A probabilidade de ser uma mensagem imprópria

$$P_r(S|W) = \frac{P_r(W|S).P_r(S)}{P_r(W|S).P_r(S) + P_r(W|H).P_r(H)} \tag{16}$$

Sendo:

$\boldsymbol{P_r(S|W)}$: probabilidade de ser mensagem imprópria dado que existe uma certa palavra W nela;

$\boldsymbol{P_r(W|S)}$: probabilidade de ter tal palavra W, dado que é mensagem imprópria (obtida atráves de ensinamento do sistema);

$\boldsymbol{P_r(S)}$: probabilidade de ser mensagem imprópria;

$\boldsymbol{P_r(W|H)}$: probabilidade de ter tal palavra W, dado que não é mensagem imprópria (obtida atráves de ensinamento do sistema);

$\boldsymbol{P_r(H)}$: probabilidade de não ser mensagem imprópria.

Como:

$$P_r(S) = P_r(H) = 0,5 \tag{17}$$

Logo:

$$P_r(S|W) = \frac{P_r(W|S)}{P_r(W|S) + P_r(W|H)} \tag{18}$$

Probabilidades combinadas:

$$p = \frac{p_1.p_2...p_n}{p_1.p_2...p_n + (1-p_1)(1-p_2)...(1-p_n)} \tag{19}$$

- *p* : probabilidade de ser mensagem imprópria;
- p_1: probabilidade de ser mensagem imprópria, dado que possui a palavra. $W_1 \rightarrow p(S|W_1)$
- p_2: probabilidade de ser mensagem imprópria, dado que possui a palavra. $W_1 \rightarrow p(S|W_2)$

Para a realização dos teste utilizamos as tabelas de mensagens já mencionadas anteriormente.

Foi calculada a probabilidade de cada palavra aparecer em cada uma das tabelas de acordo com o número de mensagens de treinamento.

Por exemplo, a palavra "promocao"aparece 7 vezes nas mensagens consideradas impróprias das 340 utilizadas para o treinamento. Portanto $P('promocao'|S)$ = $\frac{n_{vezes}}{N_{totais}}$.

Sabendo as probabilidades de se ter uma mensagem imprópria, sabendo que possui determinada palavras. Basta calcular a probablidade combinada p de a mensagem completa ser inaceitável.

PAra efeito de maximização do resultados:

- Quando uma palavras estiver associada apenas ao contexto julgado apropriado, se atribuirá uma probabilidade final de $0,01 \rightarrow$P$_r(S|W) = 0,01$
- Já para o caso de a palavra estiver associada apenas ao contexto julgado inadequados, se atribuiraá um probabilidade final de $0,99 \rightarrow$P$_r(S|W) = 0,99$;

Para regular a porcentagem de corte utilizamos a regra de gauss de distribuição de populações. (Explicar 3σ - média, média, 3σ+ média)

Assim as mensagens que obtiverem probabilidade final final maior que a média - 3σ serão consideradas inadequadas.

Teste		Média	Desvio padrão	Corte
bad works	boas	0,0578233	0,197185295262378	0,234287602714858
	ruins	0,92658103	0,230764475921766	
comerciais	boas	0,105832119	0,275143550767298	0,777984724435771
	ruins	0,991945686	0,071320320668842	
homofóbicas	boas	0,17574575	0,353808584792674	0,984077328270166
	ruins	0,998431942	0,004784871397585	
racistas	boas	0,162681586	0,342219179858330	0,783485014298300
	ruins	0,992670798	0,069728594711261	

Figura 3.3: Resultados dos testes com Redes Bayesiana.

Tabela 3.3: Tabela com os resultados obtidos utilizando Redes Bayesianas

Contexto	Quantidade Total de mensagens	Quantidade total de acertos	%
Bad works	725	589	81,23%
Comerciaiss	680	598	87,94%
Homofobicas	390	323	82,82%
Racistas	390	335	85,89%

3.4.1: Resultados

3.5: Comparativo

Fica evidente, observando os resultados obtidos, que o método utilizando Rede Bayesianas é melhor que Árvode de decisão para esse problema. Analisando a essencia do método já era possível ter uma noção que essa fato ocorreria, uma vez que Bayes, analisa como um todo o item que está sendo avaliado, sendo mais genérico e levando em consideração a mensagem como um todo, não analisando somente as palavras que a compoem.

4 Resultados

Foi realizado uma compara entre duas tecnicas de IA para a classifica de mensagens, diante dos resultados obtidos. Optou-se por realizar a integra utilizando Redes Bayesianas.

Foram geradas 4 Tabelas MySQL que contem o informas sobre o treinamento de mensagens com 4 contextos.

Foi desenvolvida um ambiente em HTML para a classifica de novas mensagens.

Como forma de agradar ao cliente mensagens consideradas com contedo inapropriado poderam serm enviadas, contudo, caso um operador humano julgue a mensagem como inadequada, seroqueada. E a pessoa que enviou a mensagem receber alerta sobre o ocorrido.

5 Conclusão e Perspectivas Futuras

Este documento apresentou o desenvolvimento de um filtro classificador de mensagens, com intensa aplicação dos conhecimentos adquiridos durante a graduação do curso de Engenharia de Controle e Automação.

Os resultados obtidos foram compatíveis com os requisitos de projeto no que tange o a classificação de mensanges. Com este trabalho é possível verificar a importância da Inteligência Artificial, tornando possível tarefas que a principio seriam inrealizáveis humanamente. Podemos percebem quanto essa técnicas podem ser inseridas em nosso dia-a-dia para auxiliar na tomada de decisões.

O trabalho foi satisfatório para o tempo dispendido, consegui-se realizar todas as tarefas propostas, apesar de ter mudado o escopo do meu trabalho consegui realizar outras tarefas extras que que não estavam mais planejadas no começo do estágio, como por exemplo a, o desenvolvimento de um driver que convertia endereços em posições geográfica, uma mapa (Google Maps) em que esses pontos eram plotados e um driver que buscava CEP do sistema dos Correios, utilizando as informações que tinha no momento.

O trabalho foi interessante tanto do ponto de vista da empresa que conseguiu agregar valor aos seus serviços, como também foi muito interessante do ponto de vista do autor que teve muito conhecimento agregado na realização das tarefas além de aumentar a sua experiência de desenvolvimento em um ambiente de trabalho.

O projeto desenvolvido ainda não se encontra funcionamento, mas será integrado ao sistemas *Cellmídia Center* em Agosto de 2012.

Pontos a melhorar:

- Realizar mais treinamento, para melhorar o desempenho do filtro;

- Melhorar a função de normalização, para diminuir a possibilidade de burlar o filtro. Ex: Analisar quando uma palavra é inseridas com e s p a ç o s em entres as letras.

Referências

[1] CAMPONOGARA, P. E. DAS-5341: Inteligência Artificia - Parte I.

[2] SOUZA, M. Z. Monitoramente de recursos em sistemas de produção via SMS. 2008.

[3] GOMES, A. Metodologias de desenvolvimento de software. jan. 2012. Available from internet: <http://www.andreygomes.com/index.php?option=comunderline contentview=articleid=1:metodologias-de-desenvolvimento-desoftwarecatid=1:metodologiasItemid=2/>.

[4] RATIONAL Unified Process. jan. 2003. Available from internet: <http://www.angusyoung.org/arquivos/artigos/trabalh_$_{r}up.pdf$>.

[5] DUTRA, R. L. M. . I. Redes Bayesianas: o que são, para que servem, algoritmos e exemplos de aplicações. 2011.

[6] ATTUX, F. J. V. Z. . R. R. F. Árvores de decisão - DCA/FEEC/Unicamp. Available from internet: <ftp://ftp.dca.fee.unicamp.br/pub/docs/vonzuben/ia004_1s10/ notas_de_aula/topico7_IA004_1s10.pdf>.

[7] FONSECA, J. M. M. R. da. Indução de Árvores de Decisão . jul. 1994. Available from internet: <http://www.uninova.pt/ jmf/Documentos pdf/Tese Mestrado/Tese Mestrado.pdf>.

[8] LACERDA, W. S. Experimento de um Classificador de Padrôes Baseado na Regra Naive de Bayes.

[9] GARCIA, A. L. Probability and Random processes for Electrical Engineering. Addison-Wesley Publishing Company. 1989.

[10] WIKIPéDIA. Available from internet: <http://pt.wikipedia.org/wiki/Servidor_web>.

[11] WIKIPéDIA. Available from internet: <http://pt.wikipedia.org/wiki/MySQL>.

[12] WIKIPéDIA. Available from internet: <http://pt.wikipedia.org/wiki/PHP>.