



Licenciatura em Sistemas e Tecnologias de Informação

**Reconhecimento de conceitos e atitudes em opiniões sobre hotéis na  
Web 2.0**

Projeto Final de Licenciatura

Elaborado por Marco Luciano Silva Sequeira

Aluno nº 20101335

Orientador: Professor Doutor Carlos Alberto Palma Pinto

Co-Orientador: Professor Doutor Marcirio Silveira Chaves

Barcarena

Dezembro 2013

Universidade Atlântica

Licenciatura em Sistemas e Tecnologias de Informação

**Reconhecimento de conceitos e atitudes em opiniões sobre hotéis na  
Web 2.0**

Projeto Final de Licenciatura

Elaborado por Marco Luciano Silva Sequeira

Aluno nº 20101335

Orientador: Professor Doutor Carlos Alberto Palma Pinto

Co-Orientador: Professor Doutor Marcirio Silveira Chaves

Barcarena

Dezembro 2013

O autor é o único responsável pelas ideias expressas neste relatório

## **Agradecimentos**

Ao meu Orientador Professor Doutor Carlos Pinto, sempre disponível para me apoiar

Ao meu coorientador, Professor Doutor Marcirio Chaves que me lançou e apoiou neste desafio

Ao Rui Faria, sempre disponível para me apoiar e esclarecer as minhas dúvidas técnicas

Aos voluntários que me ajudaram a classificar as frases, sem os quais não seria possível apresentar resultados deste trabalho

Aos meus colegas e professores da faculdade, com os quais vivi 3 anos incríveis!

Aos meus colegas e hoje grandes amigos, Carlos Mareco, José Andrade, Luís Correia e Paulo Dias, os 5UatlaBoys. Sem vocês isto não tinha metade da piada!

A todos os meus amigos que de uma forma ou outra sempre me apoiaram neste projeto e o viveram comigo.

Á Elsa Ferreira, por o seu amor, apoio incondicional e a sua capacidade de me motivar. Sem ti nunca teria sido possível aqui chegar!

Á minha Mãe, porque sem ti não era nada

Em especial ao meu Pai, que acredito estar algures a desfrutar este momento com muito orgulho!

## **Resumo**

### **Reconhecimento de conceitos e atitudes em opiniões sobre hotéis na Web 2.0**

Devido à forma como a expansão e consolidação da web 2.0 levou a que fosse radicalmente alterada a maneira como comunicamos hoje, e o crescente número de comentários *on-line* que surgem todas os dias na web em vários canais dedicados exclusivamente para esse efeito, tornou muito apelativa a ideia de conseguir extrair de uma forma direta o essencial da informação presente nesses mesmos comentários. O objetivo deste trabalho é contribuir para esse fim, desenvolvendo um algoritmo que permite o reconhecimento de conceitos e atitudes em comentários em português no âmbito dos hotéis. Para isso, foi utilizada uma abordagem lexical para o reconhecimento de conceitos e uma abordagem léxico-sintática para reconhecer atitudes. Os resultados do trabalho foram animadores. Os valores obtidos por o algoritmo quando comparado com a avaliação manual dos voluntários não diferem em muitos dos resultados alcançados quando sujeito ao mesmo conjunto de frases a uma comparação direta entre os voluntários que classificaram manualmente as frases.

Palavras-chave

Prospecção de textos em português, teoria de avaliação, avaliações *on-line*, Web 2.0

## **Abstract**

### **Recognition of concepts and attitudes in opinions on hotels in Web 2.0**

Because of the way the expansion and consolidation of Web 2.0 has led to a radically changed the way we communicate today, and the growing number of on-line comments that appear every day on the web in several dedicated channels exclusively for this purpose, it was very appealing the idea of getting a direct extract from the substance of this information in those reviews. The aim of this work is to contribute to this end by developing an algorithm that allows the recognition of concepts and attitudes in comments in the context of hotels for Portuguese language. For this, the methodology used was a purely lexical approach to concepts and lexical-syntactic approach to attitudes. The findings were encouraging. For the same set of sentences, the score obtained by the algorithm versus a human show little difference to the scores obtained for human versus human.

### Keywords

Exploration of texts in Portuguese, appraisal theory, on-line reviews, Web 2.0

## Índice

Agradecimentos .....	iii
Resumo .....	iv
Abstract .....	v
Índice.....	vi
Índice de Figuras .....	viii
Índice de Tabelas .....	viii
1. Introdução .....	1
2. Revisão da Literatura .....	4
2.1. Trabalhos relacionados em detecção de conceitos e atitudes .....	4
2.2. A Teoria da Avaliação .....	7
2.3. Considerações sobre o capítulo.....	14
3. Metodologia .....	15
4. A Solução Proposta.....	16
4.1. Arquitetura de informação .....	17
4.2. Desenvolvimento de algoritmo para detecção de conceitos .....	20
4.3. Desenvolvimento de algoritmo para detecção de Atitudes .....	20
4.4. Implementação da plataforma de desenvolvimento.....	22
4.5. Treino e teste dos algoritmos .....	23
4.6. Análise de resultados .....	23
4.7. Resultados do conjunto de treino .....	24
4.8. Resultados dos conjuntos de avaliação 1 e 2 .....	25
4.9. Análise de resultados .....	30

5. Considerações finais .....	31
5.1. Oportunidades de melhoria e limitações do trabalho.....	32
5.2. Trabalhos futuros .....	33
Bibliografia .....	34
Anexo 1 – Algoritmo para divisão de comentários em frases .....	38
Anexo 2 – Algoritmo para geração aleatória de conjuntos de frases.....	38
Anexo3 – Algoritmo para geração de conjuntos sem avaliação duplicando já existentes.....	38
Anexo 4 – Algoritmo para detecção de conceitos .....	39
Anexo 5 – Algoritmo para detecção de Atitudes.....	40
Anexo 6 - Fig. 4 - Home Page .....	46
Anexo 7 - Fig. 5 - Criação aleatória de Análises .....	46
Anexo 8 -Fig. 6- Lista de análises criadas .....	47
Anexo 9 -Fig. 7 - Detalhe do conjunto de análise.....	47
Anexo 10 -Fig. 8 - Detalhe da classificação da frase.....	48

## Índice de Figuras

Fig. 1 - Análise de proeminência de adjetivos em texto. ....	5
Fig. 2 - Avaliação de Atitudes. ....	8
Fig. 3 - Relação de Afeto com Julgamento e Apreciação .....	10
Fig. 4 – Transformação de comentário em frases .....	15
Fig. 5 - Processo de contagem de palavras para criação de tabela de chaves de conceito .....	17
Fig. 6 - Modelo de arquitetura de informação .....	17
Fig. 7 - Home Page .....	46
Fig. 8 - Criação aleatória de Análises .....	46
Fig. 9- Lista de análises criadas .....	47
Fig. 10 - Detalhe do conjunto de análise.....	47
Fig. 11 - Detalhe da classificação da frase.....	48

## Índice de Tabelas

Tabela 1 - Tipos de Apreciação. ....	13
Tabela 2 - Conceito X Chave Detecção.....	18
Tabela 3 - Atitudes X Chave de detecção.....	19
Tabela 4 - Resultados do conjunto de treino para detecção de Conceitos.....	24
Tabela 5 - Resultados do conjunto de treinos para detecção de Atitudes.....	25
Tabela 6 - Detecção conceito para conjunto 1 User3In Vs. Automático.....	25
Tabela 7 - Detecção conceito para conjunto 2 User1Ex Vs. Automático.....	26
Tabela 8 - Detecção conceito para conjunto 1 User3In Vs. User1In.....	26

Tabela 9 - Detecção conceito para conjunto 1 User3In Vs. User2In.....	26
Tabela 10 - Detecção conceito para conjunto 2 User1Ex Vs. User3Ex .....	27
Tabela 11 - Detecção conceito para conjunto 2 User1Ex Vs. User2Ex .....	27
Tabela 12 - Valores de concordância do conjunto 1 para conceitos entre os 3 avaliadores .....	27
Tabela 13 - Valores de concordância do conjunto 2 para conceitos entre os 3 avaliadores .....	28
Tabela 14 - Detecção Atitude User3In Vs. Automático .....	28
Tabela 15 - Detecção Atitude User1Ex Vs. Automático .....	28
Tabela 16 - Detecção Atitude User3In Vs. User 1In .....	29
Tabela 17 - Detecção Atitude User3In Vs. User 2In .....	29
Tabela 18 - Detecção Atitude User1Ex Vs. User 3Ex.....	29
Tabela 19 - Detecção Atitude User1Ex Vs. User 2Ex.....	30
Tabela 20 - Valores de concordância do conjunto 1 para Atitudes entre os 3 avaliadores .....	30
Tabela 21 - Valores de concordância do conjunto 2 para Atitudes entre os 3 avaliadores .....	30

## 1. Introdução

A Era de expansão e consolidação da Web 2.0 alterou dramaticamente a forma como comunicamos e expressamos as nossas opiniões *on-line* (Liu, 2010). Observamos um crescente surgir de comentários sobre os mais variados temas e produtos disseminados na Internet, de blogues a *websites* dedicados, passando por os comentários colocados nas redes sociais onde as empresas optam por estar representadas.

Um dos temas que têm vindo a florescer neste novo conceito de web participativa, são as críticas por parte dos utilizadores. Essas críticas geralmente contêm sentimentos e atitudes sobre a experiência deles enquanto clientes que compram produtos ou utilizam serviços. Por exemplo, espaços dedicados à hotelaria, nomeadamente em locais como *TripAdvisor* ([www.tripadvisor.com](http://www.tripadvisor.com)), *Booking* ([www.booking.com](http://www.booking.com)), *Oyster* ([www.oyester.com](http://www.oyester.com)) possuem opiniões *on-line*. Reflexos disso são os números de acessos apresentados no *TripAdvisor*, com mais de 60 milhões de visitantes por mês e mais de 75 milhões de avaliações e opiniões em todo o mundo (“Sobre o TripAdvisor,” 2012).

Torna-se assim um tema interessante desenvolver funcionalidades que permitam tratar de uma forma sistemática toda a informação gerada e que se encontra dispersa nestes espaços. É neste âmbito que se define este trabalho, tentar ser um contributo para a identificação de conceitos e atitudes presentes nas frases em comentários *on-line* em português no âmbito da hotelaria, baseado na teoria da avaliação de (Martin & White, 2005).

Este objetivo não é isento de dificuldades, como é facilmente perceptível tendo como ponto de partida a definição de atitude que encontramos em (Martin & White, 2005), que classifica quanto a polaridade positiva ou negativa afeto, julgamento e a apreciação, conseguimos identificar como exemplo nas seguintes frases retiradas de (“Tivoli Lisboa (Lisboa, Portugal),” 2012) em termos de atitude:

- “Não apreciei a falta de limpeza do quarto.” – Afeto negativo no conceito quarto. Considera-se também uma apreciação negativa ao mesmo conceito.

- “Elevado profissionalismo e simpatia do pessoal” – Julgamento positivo do conceito pessoal
- “A garagem é muito pequena” – Apreciação negativa no âmbito do estacionamento

A dificuldade que apresenta o reconhecimento automático de atitudes em textos informais curtos, prende-se com o facto de que a opinião gerada por utilizadores em comentários *on-line* pode ser expressa de diversas formas, utilizando palavras distintas para manifestar tipos de atitude e sentimentos, sejam eles positivos, negativos ou até neutros. Para ser possível captar este tipo de informação de uma forma eficaz, é necessário primeiro entender o significado de cada frase presente no comentário (M. Chaves & Picoto, 2012).

Uma outra dificuldade prende-se com a distinção entre apreciação e afeto. Existem claras semelhanças entre ambos os sentimentos e mesmo a nível lexical é muitas vezes difícil fazer a distinção, sendo importante separar quando possível entre o sentimento que uma pessoa tem e a atribuição desse sentimento como uma apreciação (Martin & White, 2005). Na verdade, nos textos de informação curtos é muitas vezes difícil ou impossível fazer essa distinção, tendo de ser avaliadas as atitudes de uma forma cumulativa.

O objetivo deste trabalho é desenvolver um algoritmo com base numa abordagem lexical, que permita automaticamente identificar em textos informais em Português, as áreas que caracterizam um hotel e que são objeto de análise por parte dos utilizadores. Esse algoritmo reconhece numa segunda fase os tipos de atitude presentes em cada uma das áreas do hotel anteriormente identificadas, sejam estes afetos, julgamentos ou apreciações e qual a sua polaridade, positiva, negativa ou neutra.

No universo empresarial, quando no passado era necessário conduzir inquéritos entre clientes para ter uma visão da imagem que estes tinham do hotel, hoje essas empresas contam com o valioso contributo dos comentários gerados pelo utilizador nos *sites* dedicados ou nas redes sociais. Essa informação é valiosa como ferramenta de Marketing, Psicologia Social e outros campos interessados em extrair e analisar visões, comportamentos e atitudes dos comentários dos utilizadores (Tang, Tan, & Cheng,

2009). São também relevantes para os futuros e atuais utilizadores destes espaços, pois a informação retirada da Web e análise de comentários sobre unidades hoteleiras reduzem as incertezas sobre a qualidade e os riscos de tomar uma determinada opção, tornando o processo de seleção mais eficaz e até divertido (Huang, Basu, & Hsu, 2010).

De uma tarefa difícil para as empresas e clientes monitorizarem e analisarem todos os comentários que surgem de locais distribuídos pela Internet e os milhares de comentários de opinião aí presentes, surge a necessidade de automatização destes processos (Liu, 2010). Consequentemente uma análise de sentimentos e atitudes, que extrai e reconhece automaticamente as subjetividades e sentimentos (positivos ou negativos) presentes num comentário tornou-se uma área de investigação relevante (Kumar & Sebastian, 2012).

Este trabalho apresenta na sua primeira parte uma análise de literatura em trabalhos relacionados com as metodologias empregues em classificação de sentimentos, atitude e suas polaridades, bem como os conceitos inerentes a afeto, julgamento e apreciação subjacentes à definição de atitude. O capítulo 2 descreve a arquitetura de informação inerente ao trabalho, detalhando o fluxo e tratamento dos dados (entrada, processamento e resultados) bem como a estrutura definida para o algoritmo de deteção de atitudes. No capítulo seguinte discutem-se os resultados, comparando-se a avaliação humana com o resultado da avaliação conseguida com o algoritmo. A última parte do trabalho apresenta as conclusões, limitações e dificuldades encontradas no desenvolvimento do projeto, bem como o que o autor pensa ser o caminho a desenvolver no âmbito de trabalhos futuros.

## 2. Revisão da Literatura

Este capítulo tem como objetivo apresentar uma análise de literatura sobre as técnicas mais difundidas de classificação de sentimentos e atitudes, além dos trabalhos realizados nesse âmbito em Inglês e Português, justificando com esta análise a relevância do trabalho que aqui se apresenta.

### 2.1. Trabalhos relacionados em detecção de conceitos e atitudes

Encontramos ao longo da literatura em inglês diversas formas de definir análise e extração de orientação semântica, como análise de sentimentos (Pang & Lee, 2008), subjetividade (Langacker, 1985; Lyons, 1981), avaliação de sentimentos (*Appraisal*) (Martin & White, 2005) entre outros.

Neste trabalho, análise de sentimentos refere-se à polaridade (negativa ou positiva) das frases presentes em comentários *on-line* em Português, analisados em função dos conceitos aí presentes, e avaliação de sentimentos ou atitudes (*Appraisal*) ao sentimento presente nessas frases, afeto, julgamento ou apreciação, segundo o subsistema apresentado por (Martin & White, 2005),

Existem duas técnicas principais para a classificação de sentimentos, técnicas simbólicas (também chamada de abordagem lexical), e técnicas de aprendizagem de máquina (Boiy, Hens, Deschacht, & Moens, 2007) ou classificação de texto estatisticamente. A abordagem lexical utiliza regras manuais e/ou regras gramaticais, em conjunto com léxicos da língua utilizada, por oposição às técnicas de aprendizagem de máquina que podem ser não supervisionadas, com pouca supervisão ou totalmente supervisionadas com vista à construção de um modelo baseado num completo conjunto de dados de treino.

A maioria das técnicas de classificação estatística de texto baseia-se em algoritmos de aprendizagem, nomeadamente classificadores *Support Vector Machine*, que são treinados com um conjunto específico de dados sobre determinado tema. Quando sujeitos ao domínio para o qual foram treinados, esta técnica revela resultados muito positivos (Chaovalit & Zhou, 2005; Kennedy & Inkpen, 2006; Boiy et al., 2007), mas alterando-se o âmbito, esses resultados baixam para um nível de desempenho perto do nulo (Taboada, Brooke, Tofiloski, Voll, & Stede, 2011).

Existem termos que podem ter uma conotação positiva ou negativa dependendo do âmbito ou tema em que estão inseridos. A abordagem lexical inclui o uso de listas de nomes, verbos adjetivos e advérbios, e uma lista de conjunções e conectividades (M. Chaves & Picoto, 2012), sendo assim muito mais fácil estender esta técnica a qualquer

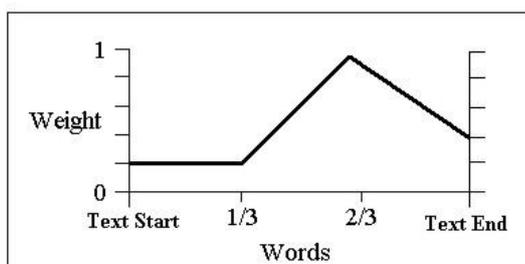
tipo de domínio que se pretenda, pois estas listas são independentes do domínio a que são aplicados, razão que justifica a sua utilização no presente trabalho.

Aplicado ao inglês, (Turney, 2002) classifica os comentários como recomendados (*Thumbs up*) ou não recomendados (*Thumbs down*), sendo esta classificação definida pela média da “orientação semântica” positiva ou negativa dos adjetivos ou verbos presentes nas frases.

(Taboada & Grieve, 2004), baseadas no *Appraisal* tal como sistematizado por (Martin, 2000), desenvolvem um método para classificar textos automaticamente sustentado na subjetividade do seu conteúdo, dando mais relevância às partes do texto em que acreditam estar concentrado conteúdo mais subjetivo. Sendo que os textos são normalmente estruturados no seu nível mais básico em princípio, meio e fim, as autoras teorizam que as opiniões expressas se encontram tendencialmente no meio e no final dos textos, sendo especialmente verdade em comentários, onde os autores tendem a terminar com a sumarização da sua opinião. Este estudo é baseado em textos curtos de informação escritos em Inglês, sendo necessária a sua validação para o Português.

Para implementarem esta teoria, quando analisam o texto têm em consideração o peso da orientação semântica de cada adjetivo baseado na sua localização no texto, revelando que existe uma menor proeminência no início do texto, um claro aumento no meio com tendência descendente no final conforme a Fig. 1 - Análise de proeminência de adjetivos em texto.

**Fig. 1 - Análise de proeminência de adjetivos em texto.**



(Taboada & Grieve, 2004)

Definidas as partes do texto onde consideram estar concentrado mais conteúdo sujeito a análise, torna-se necessário saber se o conteúdo é positivo ou negativo baseado no método de (Turney, 2002), para depois determinar com suporte no potencial de avaliação dos adjetivos, em que grau esse conteúdo expressa Afeto, Julgamento e Apreciação.

Esta abordagem para classificação de orientação semântica para adjetivos conseguiu resultados encorajadores, expandindo o método básico exposto acima tendo em atenção a posição de cada adjetivo presente no texto. Foi detetado que o desempenho tem variações dependendo do âmbito dos comentários analisados. A extração dos valores de *Appraisal* revela também características diferentes de acordo com o âmbito dos comentários.

(Read & Carroll, 2012) apresentam uma abordagem interessante para a forma como comentários publicados eletronicamente podem também ser classificados com base no sistema “*Appraisal*” de (Martin & White, 2005). Mais uma vez a abordagem ao tema tem como base o trabalho desenvolvido por (Turney, 2002), na sua noção que o sentimento de um texto pode ser determinado estimando as semelhanças com exemplos protótipos de sentimentos positivos e negativos. Neste caso são utilizados dois métodos para estimar a semelhança entre as palavras, associação lexical e espaços semânticos, que são eficazes para lidar com a subjetividade da linguagem, segundo (Read & Carroll, 2012).

As medidas de associação lexical examinam as semelhanças de primeira ordem entre palavras (Grefenstette, 1994), e assim determinam a semelhança de um par de palavras, considerando a probabilidade de estas aparecerem juntas. Por outro lado, os espaços semânticos representam conceitos em conjunto de pontos num grande número de dimensões. A localização de cada ponto ao longo de cada dimensão é a medida da força de associação nesse eixo. Assim é possível avaliar a semelhança entre conceitos aplicando distâncias métricas aos vetores extraídos do resultado dessa matriz.

A aplicação destes métodos com fraca supervisão é apelativa no âmbito das técnicas de aprendizagem máquina, pois não necessita que os dados de treino sejam etiquetados, mas apenas um “*corpus*” (grande conjunto de textos), e um conjunto manualmente especificado de exemplos protótipos para cada classe.

Dos dois métodos aplicados, o que apresentou melhores resultados segundo os autores foi a versão dos espaços semânticos, apesar de ser mais exigente a nível computacional, sendo indicado também que o método é tão robusto quanto maior for o número de classes.

Tendo como objetivo conseguir uma maior granularidade na deteção de polaridade, classificando as opiniões sobre cada característica do objeto que é monitorizado, (N. R. Silva, Lima, & Barros, 2012) apresentam o “SAPair: Um Processo de Análise de Sentimento no Nível de Característica”. Propõe-se classificar pares (característica, palavra opinativa) tendo em conta que alguns adjetivos mudam a sua polaridade em função dos substantivos que classificam (p.ex. “cerveja quente”, “pizza quente”). Foram utilizadas técnicas baseadas em Estatística e Linguística, com o apoio da ferramenta

*SentiWordNet*, tendo os resultados experimentais, segundo os autores, demonstrado que o processo tem uma alta eficácia.

Mais centrado na linguagem portuguesa, (M. Chaves, de Freitas, Souza, & Vieira, 2012) apresentam o algoritmo denominado “PIRPO”, que foi desenvolvido com o objetivo de identificar a polaridade em comentários gerados por utilizadores em Português. Cada comentário é decomposto em frases para assim ser possível aferir a polaridade existente em cada conceito de uma ontologia. O módulo do algoritmo para deteção de polaridade recebe uma lista de adjetivos e conceitos de uma ontologia sendo o resultado uma lista de frases com a sua polaridade reconhecida em função de cada conceito detetado. Os melhores resultados incluem um F-score de 0.32 em comentários com polaridade mista.

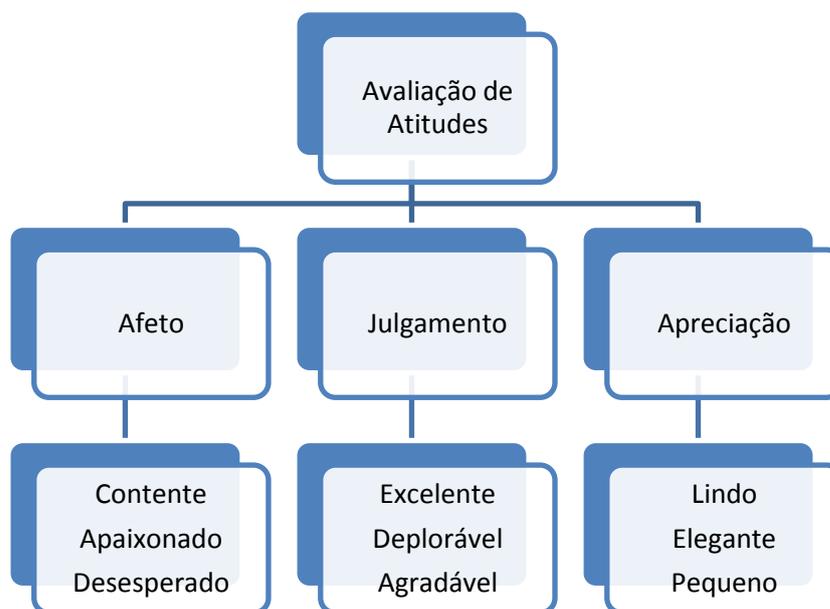
Este algoritmo é o que mais se aproxima do âmbito deste trabalho, pois toda a primeira fase do mesmo terá como objetivo o *input* de comentários e a sua divisão em frases, para com base numa ontologia detetar os conceitos presentes em cada uma delas. O avanço sobre o “PIRPO” será a inclusão do subsistema de *Appraisal* (Afeto, Julgamento ou Apreciação) à saída atual “frase, conceito, polaridade” do algoritmo.

## **2.2. A Teoria da Avaliação**

Sendo o objetivo deste trabalho utilizar uma abordagem lexical na avaliação de sentimentos e atitudes presentes em comentários *on-line* em português no âmbito da hotelaria, segundo o sistema apresentado por (Martin & White, 2005) no seu livro “*The language of evaluation*”, define-se em seguida o mesmo baseado nesses autores.

Avaliação de sentimentos, ou *Appraisal* é uma teoria linguística de subjetividade. O sistema *Appraisal*, foi desenvolvido na tradição do sistema funcional linguístico de (Halliday, 1994), e é uma tentativa de modelar a capacidade da linguagem em expressar e negociar opiniões e atitudes dentro de um texto. Este sistema, segundo (Martin & White, 2005), divide-se em três subsistemas distintos: Afeto, Julgamento e Apreciação, que modelam a capacidade de expressar opiniões de uma forma emocional, moral e estética, respetivamente. Esses termos, estão definidos na secção 2.2.1, 2.2.2 e 2.2.3, respetivamente, e o sistema representado esquematicamente na Fig. 2, onde podem ser encontrados exemplos de palavras capazes de modelar cada um dos subsistemas. Tanto quanto é conhecimento do autor, para a língua portuguesa não existe registo de nenhum trabalho efetuado a este nível de granularidade dos subsistemas.

**Fig. 2 - Avaliação de Atitudes.**



**Baseado no Sistema *Appraisal* de (Martin & White, 2005).**

### 2.2.1. Afeto

Afeto diz respeito ao registo positivo e negativo (polaridade) de sentimentos, o estado emocional do sujeito, se está feliz ou triste, confiante ou ansioso, interessado ou chateado (Martin & White, 2005).

Analisando atitude como um sistema de discurso semântico, é expectável que o afeto nas suas diferentes conjugações encontre variações ao longo das estruturas gramaticais. Voltando a (Halliday, 1994) estas conjugações incluem a variação de participantes (qualidades), processos mentais ou comportamentais e adjuntos modais.

Assim, podemos ter afeto como:

- Uma qualidade
  - Descrevendo participantes com um epíteto ou qualificação
    - (...) **muito atenciosos** os funcionários.
  - Atribuído a participantes, como um atributo
    - (...) os funcionários são **muito atenciosos**.
  - Como modo de fazer algo, uma circunstância
    - (...) a **grande atenção** dos funcionários.
- Um processo
  - Mental
    - A localização do Hotel a 10 metros da praia **vai permitir** desfrutar com grande comodidade da praia.
  - Comportamental
    - A localização do Hotel a 10 metros da praia **permitiu desfrutar** com grande comodidade da praia
- Como um comentário
  - Desiderato, sentido de perda. Adjunto modal.
    - **Infelizmente** tiveram de partir.

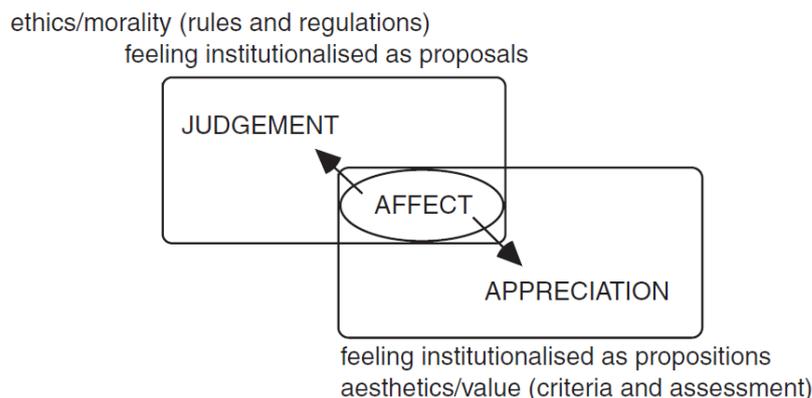
Para identificação de afeto (Martin & White, 2005) sugerem seis fatores que devem ser considerados:

- Sentimentos são considerados culturalmente positivos e negativos?
- Os sentimentos são o resultado de emoções?
- São os sentimentos resultantes de alguma reação externa?
- A graduação dos sentimentos é lexicalizada?
- Sentimentos envolvem intenções mais do que reações?
- Qual a variação final da tipologia de afeto?

Todas estas questões têm de ser respondidas para ser possível identificar o afeto enquanto atitude em textos informais em Português.

Afeto é provavelmente a mais complicada atitude de ser reconhecida, pois têm frequentemente associados atitudes de julgamento ou apreciação, encontra-se esta no centro da definição de *Appraisal*, segundo (Martin & White, 2005) e apresentado esquematicamente na Fig. 3 - Relação de Afeto com Julgamento e Apreciação.

**Fig. 3 - Relação de Afeto com Julgamento e Apreciação**



(Martin & White, 2005)

São várias as palavras ou expressões que nos ajudam a caracterizar afeto, como medo, amor, ódio, ânsia, prazer, satisfação, de uma forma positiva ou negativa, sendo que um dos desafios se encontra em identificar quando esta atitude está ou não associada a outra forma, seja ela julgamento ou apreciação.

O outro desafio é claro, conseguir identificar automaticamente, que partes do texto devem ser identificadas e como, para assim ser possível caracterizar a atitude como afeto. No exemplo seguinte, temos identificado o conceito quarto:

- “A primeira impressão foi péssima, mal entramos no quarto (minúsculo) o " bafo " de ar quente que se libertou deu logo uma péssima impressão.”

Este conceito tem uma conotação claramente negativa, pela adjectivação de “péssima”, da primeira impressão, “minúsculo” e “bafo de ar quente” quando se refere explicitamente ao conceito quarto.

Conseguimos caracterizar o afeto, pois denota claramente o estado de espírito do sujeito como sendo de desilusão, mas essa é uma interpretação humana do que está escrito! A dificuldade está em conseguir detetar automaticamente que a expressão “A primeira impressão foi péssima...” como um afeto negativo que denota desilusão quanto ao quarto, sendo que esta não está explícita no texto apresentado. Este comentário encerra em si também uma apreciação, o que será explicado na Secção 2.2.3 deste capítulo.

### 2.2.2. Julgamento

Também segundo (Martin & White, 2005), julgamento trata de atitudes em relação a comportamentos, que admiramos ou criticamos, elogiamos ou condenamos de outrem. Segundo o mesmo autor, com o julgamento entramos no conceito da forma como nos

comportamos e que atitudes temos perante outros e a forma com vemos o seu comportamento. De uma forma geral, o julgamento pode ser dividido em duas formas, a que lida com estima social e a orientada a sanção social. Julgamento de estima social enquadra-se na chamada normalidade, o quão normal ou pouco usual uma pessoa é, a sua capacidade, o que são ou não capazes de fazer, e na sua tenacidade, o quão decidido um individuo é.

A sanção social num julgamento enquadra-se mais no âmbito da veracidade que é atribuída a uma pessoa, qual o seu nível de honestidade, e da honra que reflete os seus valores éticos. A estima social tende a ser avaliada na cultura oral, em conversas, mexericos, anedotas e outros tipos de discurso oral, com o humor muitas vezes a ter um papel de destaque (Eggins & Slade, 2004). A partilha de valores nesta área é fundamental para a formação de redes sociais, como família, amigos, colegas de trabalho, etc.

Ainda segundo a definição de (Martin & White, 2005), a sanção social por outro lado é mais vezes expressa em escrita, em editais, decretos, regras, regulamentos e leis de como nos devemos comportar segundo os critérios do estado e da religião, com sanções e castigos contra quem não cumprir essas regras. A partilha de valores nesta área sustenta a sociedade e a sua relação com os deveres civis e religiosos.

Este trabalho centrará a sua atenção na estima social, pois é essa particularidade do julgamento que encontramos nos textos curtos de informação aqui analisados.

Numa síntese do que atrás foi dito podemos assumir com segurança que sempre que for necessário detetar automaticamente um julgamento, é necessário que texto informal curto a ser analisado contenha referências explícitas a pessoas.

Existem várias expressões que nos permitem caracterizar um julgamento, sempre associado à adjetivação de uma pessoa ou a comportamentos desta, como sortudo, especialista, heroico, cuidadoso, gentil, leal, honesto, dececionante, malvado, incompetente, e muitas outras. Estas expressões permitem também definir a polaridade do julgamento que pretendemos classificar:

Incompetente, dececionante e mal-educado são exemplos claros de uma polaridade negativa. Por outro lado competente, útil, educado e atencioso contém polaridades positivas.

É importante ter em atenção que a conjugação de palavras pode fazer variar a polaridade atribuída. Na expressão “O empregado foi muito competente” temos uma polaridade positiva, mas a alteração do advérbio do adjetivo competente muda essa polaridade. “O empregado foi pouco competente”, apesar do adjetivo ter em si uma

conotação positiva quanto à sua polaridade, a utilização do advérbio muda essa polaridade para negativa.

Esta é a dificuldade que se encontra na automatização do reconhecimento de sentimento e atitude em textos informais curtos, tem de ser considerado sempre a conjugação de adjetivo e advérbio para definir essa polaridade.

Como referido, a automatização de reconhecimento de julgamento terá como primeiro princípio a localização do conceito, sendo que para ser considerado um julgamento este deve referir-se ao pessoal que exerce funções no hotel, como apresentado no exemplo seguinte:

- “Não sei se por casualidade, mas presenciamos pelo menos duas vezes desentendimentos deles com os hóspedes ou com outros funcionários.”

Neste exemplo é claro para um humano que analise a frase que estão referidos os funcionários, logo podemos considerar que é imediato a deteção de um conceito, o pessoal, neste caso funcionários. E que a conotação ou polaridade é negativa perante o verbo desentender.

Mas se pensarmos num plano de linguagem de máquina, existe de facto alguma complexidade associada, uma vez que no tratamento da sintaxe é necessário considerar a existência de um verbo pronominal “hóspedes” e um substantivo masculino, “funcionários”, que podem ser identificados como objeto de julgamento. O desafio é conseguir conjugar a palavra “desentendimento”, verbo desentender, com o objeto correto, “funcionários” ou “hóspedes”.

### 2.2.3. Apreciação

Ainda considerando (Martin & White, 2005), apreciação envolve a avaliação de fenómenos naturais e dos diversos sistemas de sinais de comunicação entre indivíduos, de acordo com a forma como avaliam positiva ou negativamente determinado objeto ou situação.

Com apreciação entramos no campo de avaliar “coisas”, coisas que nós ou outros fazem, performances que atingimos ou avaliamos, mas também se inclui neste âmbito os fenómenos naturais, de que forma esses fenómenos são dignos ou não de valor e a forma como os apreciamos.

(Martin & White, 2005) diz-nos que em termos gerais apreciação pode-se dividir nas nossas **reações** aos acontecimentos, o seu impacto ou qualidade, à sua **composição**, o seu equilíbrio e complexidade (uma paisagem por exemplo) e à nossa **avaliação**, de que

forma algo é inovador, autêntico ou intemporal. Apresenta-se alguns exemplos de reação, composição e avaliação na Tabela 1, com respetiva polaridade.

**Tabela 1 - Tipos de Apreciação.**

<b>Apreciação</b>	<b>Identificador</b>	<b>Positivo</b>	<b>Negativo</b>
Reação	Impacto -> como afeta?	Cativante, emocionante, comovente, animado	Detestável, previsível, aborrecido, chato
	Qualidade -> gosto disto?	Bom, esplendido, apelativo, encantador	Mau, deplorável, fraco, grotesco
Composição	Equilíbrio -> como se enquadra?	Harmonioso, simétrico, proporcionado, consistente	Dissonante, assimétrico, desproporcionado, inconsistente
	Complexidade -> é difícil de entender?	Simples, claro, elegante, detalhado	Complicado, intrincado, deselegante, simplista
Avaliação	Valeu a pena?	Inovador, original, excepcional, apropriado	Convencional, falso, medíocre, deslocado

**Baseado em** (Martin & White, 2005)

A deteção automática da atitude apreciação é provavelmente a mais ambígua de detetar, pois tanto a um afeto como a um julgamento pode estar associada uma apreciação.

Se retomarmos o exemplo dado no ponto 2.2.1 para Afeto:

- “A primeira impressão foi péssima, mal entramos no quarto (minúsculo) o " bafo " de ar quente que se libertou deu logo uma péssima impressão.”

Constatamos que mais que a deteção de afeto já descrita, encontramos também uma apreciação com polaridade negativa sobre o conceito “quarto”, apreciação essa adjetivada por as expressões “minúsculo” e “bafo de ar quente”.

O desafio aqui é a deteção não de uma, mas de duas atitudes, e relacionar a polaridade e o conceito com cada uma dessas atitudes! O mesmo é válido para o binómio julgamento / apreciação.

Consideremos o comentário:

- “As atitudes displicentes e descorteses dos funcionários da receção”

Encontramos explicitamente uma referência aos empregados, o que como já descrito por ter como objeto comportamentos humanos nos remete para um julgamento, neste caso com uma polaridade negativa. Analisado com detalhe, encontramos também claramente uma apreciação negativa ao substantivo “atitude” adjetivado negativamente por “displicentes” e “descorteses”.

### 2.3. Considerações sobre o capítulo

Pretendeu-se ao longo deste capítulo apresentar uma visão global do que é hoje o estado da arte no âmbito da deteção de sentimentos e atitudes em textos de informação curta. Foram descritas as duas técnicas de classificação de sentimentos, técnicas simbólicas, também chamadas de abordagens lexicais, e as técnicas de aprendizagem de máquina.

A escolha para este trabalho será uma abordagem léxico-gramatical (Turney, 2002; Ding, Liu, & Yu, 2008; Chetviorkin & Loukachevitch, 2012; Taboada et al., 2011; M. Chaves et al., 2012), na medida em que é esta a que apresenta melhores resultados quando aplicada a diferentes âmbitos, por oposição à abordagem com técnicas de aprendizagem máquina, que apresentando resultados muito positivos no âmbito para qual foi treinada, alterando-se esse âmbito a performance desce para níveis perto da adivinhação (Taboada et al., 2011).

Apesar de se basear numa abordagem de aprendizagem de máquina o trabalho desenvolvido por (N. R. Silva et al., 2012) , encerra em si um conceito muito interessante de análise de pares, onde tem em conta a forma como os adjetivos mudam a sua polaridade em função dos substantivos que classificam.

Pretende-se assim evoluir, para definição de conceitos do domínio da hotelaria, com definidos em *Hontology* (M. S. Chaves, Freitas, & Vieira, 2012) mas com uma menor detalhe que o proposto por estes autores, e em regras com a utilização do SentiLex-PT02 (M. J. Silva, Carvalho, & Sarmiento, 2012) para a análise de atitudes em textos de informação curtos em português no âmbito da hotelaria, baseado no sistema “*Apraisal*” definido por (Martin & White, 2005) e já aplicado ao inglês por (Read & Carroll, 2012; Taboada & Grieve, 2004) em outros âmbitos de comentários.

### 3. Metodologia

A metodologia utilizada neste trabalho foi o método experimental, com o objetivo de desenvolver um algoritmo para reconhecer atitudes em frases retiradas de comentários *on-line* em português no âmbito da hotelaria. Revelou-se no decorrer do experimento ser uma valia também a detecção de conceitos representativos do mesmo âmbito, conforme descrito em (M. S. Chaves et al., 2012).

Para tal foi utilizado um *corpus* inicial gentilmente cedido e retirado do trabalho de (Gomes, 2011) onde partindo de uma amostra de 499 comentários, que foram divididos em frases (Fig. 4) - os objetos de análise deste trabalho<sup>1</sup>. Este processo resultou em 1581 frases, tendo sido retiradas as que continham estrangeirismos, resultando num *corpus* final de 1564 frases para análise.

**Fig. 4 – Transformação de comentário em frases**

O passo seguinte passou por detetar os conceitos conforme definidos em (M. S. Chaves et al., 2012), que se encontram presentes nessas frases, utilizando a técnica de frequência de palavras proposta por (Pang, Lee, & Vaithyanathan, 2002). No entanto, não se recorreu a uma análise por especialistas que indicassem uma lista de maior frequência de palavras que servissem de chave à detecção com aconteceu com o trabalho de Pang, sendo o *corpus* apenas alvo de uma contagem de palavras utilizando uma aplicação para esse efeito (Vangos, 2009).

Para a detecção de atitudes foi utilizada uma abordagem léxico-sintática. Na implementação dessa abordagem foram seguidos os seguintes passos:

- Criação aleatória de conjuntos de frases para análise a partir do *corpus* principal de 1564 frases<sup>2</sup>, dos quais se destacam:
  - Conjunto de testes com 470 frases
  - Dois conjuntos de frases para análise de voluntários
- Análise (não especialista) das frases quanto à existência de maior ocorrência de categorias gramaticais, atributos morfológicos e número.
  - Adjetivos
    - Masculino / feminino
    - Singular / Plural
  - Nomes

---

<sup>1</sup> Anexo 1 – Algoritmo para divisão de comentários em frases disponível no Anexo 1 – Algoritmo para divisão de comentários em frases

<sup>2</sup> O algoritmo para obtenção dos conjuntos aleatórios de frases disponível no Anexo 2 – Algoritmo para geração aleatória de conjuntos de frases

- Masculino / feminino
- Singular / Plural
- Verbos
  - Masculino / feminino
  - Singular / Plural
- Criação de uma tabela onde foi carregado o léxico SentiLex (M. J. Silva et al., 2012), permitindo assim a associar a palavra detetada à categoria gramatical;
- Criação de uma tabela de Tipo de Atitude, que permite associar a categoria gramatical à Atitude pretendida;

Foi assim possível proceder à deteção automática de conceitos e tipos de atitudes no conjunto de frases de testes acima mencionado, que foi também sujeito a uma avaliação manual por parte do autor, avaliação essa baseada nas definições para conceitos de (M. S. Chaves et al., 2012), e tipos de sentimento de (Martin & White, 2005).

Não são todas as frases que tem associados conceitos e atitudes, pois no caso dos conceitos, não foi utilizado todo o espectro possível enunciado em (M. S. Chaves et al., 2012), e no caso das atitudes nem todas se enquadram nos conceitos definidos na Teoria de Avaliação, segundo (Martin & White, 2005).

Tendo sido sempre o objetivo do trabalho poder submeter a análise automática conjuntos diferenciados de frases, foram gerados dois novos conjuntos aleatórios, partindo mais uma vez do *corpus* inicial de 1564 frases, excluindo o conjunto de 470 frases utilizadas para teste.

Estes dois conjuntos não foram sujeitos a qualquer análise de frequência prévia para deteção automática de conceitos e atitudes, tendo sido disponibilizado a seis avaliadores externos. Cada avaliador avaliou um conjunto de frases. No total, existem três conjuntos de frases distintas, sendo o primeiro denominado conjunto de teste, utilizado na exploração e análise para chegar ao desenvolvimento do algoritmo e dois outros destinados a serem avaliados por os voluntários.

Os resultados comparados das avaliações manuais e automáticas de cada conjunto são apresentados no subcapítulo 4.6 Análise de resultados.

#### **4. A Solução Proposta**

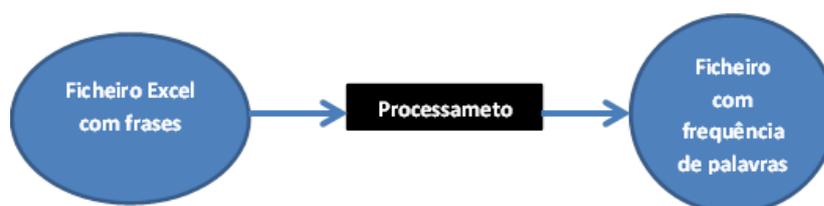
Este capítulo detalha a solução proposta e que foi implementada para alcançar os objetivos do trabalho. A seção 4.1 descreve a arquitetura de informação que levou à construção dos algoritmos, e em seguida apresentam os resultados alcançados. A seção 4.4 apresenta a descrição da criação de uma plataforma de desenvolvimento que

suportou todo o processo de deteção e avaliação dos conjuntos de frases retirados do *corpus* inicial.

#### 4.1. Arquitetura de informação

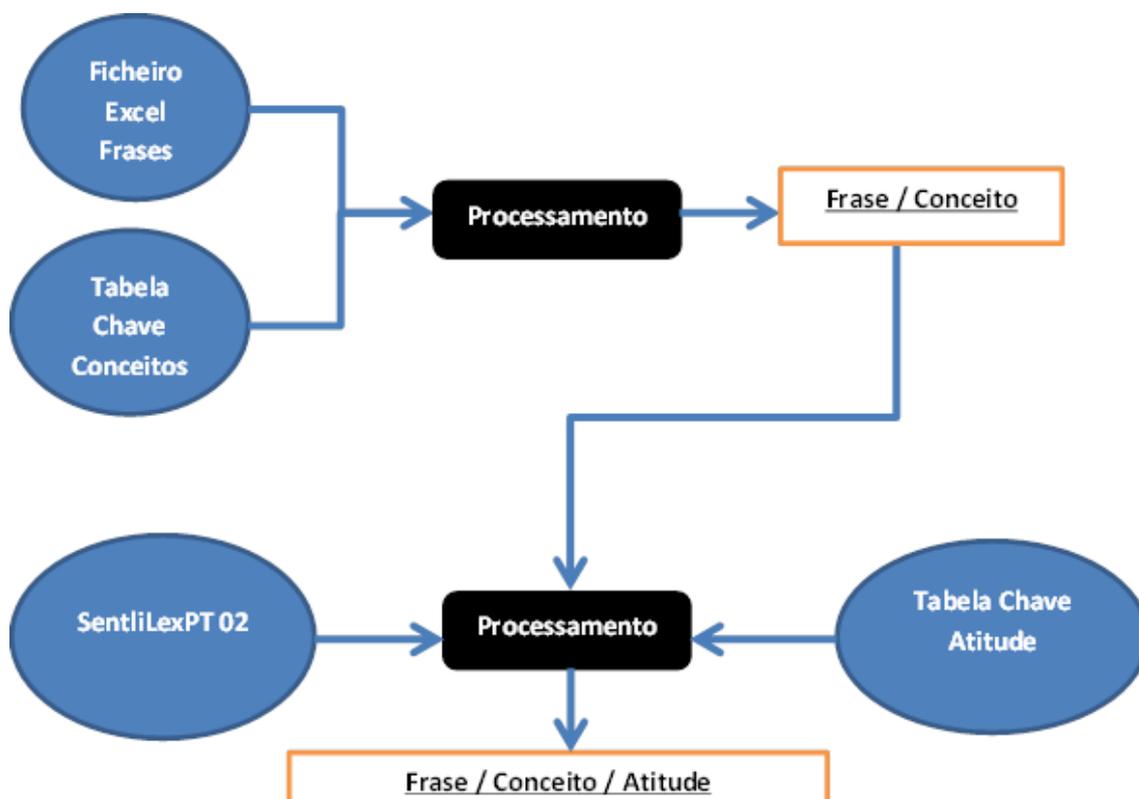
A primeira fase do processo passou por sujeitar o *corpus* completo das frases à análise de frequência de palavras (Fig. 5), com o auxílio da aplicação *Word Frequency Counter* (Vangos, 2009)

Fig. 5 - Processo de contagem de palavras para criação de tabela de chaves de conceito



A Fig. 6 Representa um modelo da arquitetura de informação utilizada no processo de obtenção de conceitos e atitudes associados a frases

Fig. 6 - Modelo de arquitetura de informação



- Na primeira fase é sujeito a processamento o conjunto de frases a ser avaliado conjuntamente com a tabela de chaves de conceito
  - a. A criação da tabela de chaves de conceito detalha-se da seguinte forma:
    - i. Foram recolhidas do Top 30 de entradas com maior frequência as palavras mais relevantes para a identificação dos conceitos, resultando na tabela Tabela 2 - Conceito X Chave Detecção
    - ii. Com a utilização de uma expressão regular desenvolvida para o efeito permitiu a utilização de *wildcards*, reduzindo assim a complexidade da tabela e o número de entradas necessária na mesma. Por exemplo a deteção de “*simpati*”, permite extrapolar a deteção para:
      1. Simpático
      2. Simpática
      3. Simpáticos
      4. Simpáticas

**Tabela 2 - Conceito X Chave Detecção**

Conceito	Chave
Hotel	Hote
Hotel	Relacao
Hotel	Internet
Staff	Funcionari
Staff	Simpati
Staff	atendimento
Staff	Recep
Staff	Pessoa
Localização	Local
Quarto	Quart
Quarto	Banh
Quarto	Limpe
Quarto	Cama
pequeno almoço	Almoc

- Desse processamento resulta um conjunto com frases às quais temos associado o devido conceito.
- Esse conjunto é então sujeito a um novo processamento e, em conjunto com o ficheiro SentiLex-PT02 (M. J. Silva et al., 2012) e a tabela que constitui as chaves para o conhecimento de atitudes.
  - a. A criação de tabela para deteção de tipos de Atitude detalha-se da seguinte forma:
    - i. Foi analisado o conjunto de teste sobre a incidência de formas gramaticais que melhor indicassem

1. Julgamento
  2. Apreciação
  3. Afeto
- ii. Neste ponto foi introduzido o SentiLex-PT02 numa tabela da base de dados, com o objetivo de servir de como uma parte da criação da chave que permite a deteção de Tipos de Atitudes.
  - iii. Da análise referida no ponto “a”, conjugada com as entradas no SentiLex-PT02, foram consideradas as colunas POS e FLEX, que representam as formas gramaticais de cada entrada de palavra.
  - iv. Foram ainda acrescentadas duas entradas no SentiLex, com as palavras “POS” e “NEG”, classificadas na coluna Pos como “M” e na coluna Flex como Mn para “NEG” e Mp para “POS”, pois são dois metadados presentes em muitas frases que reflete apreciações negativas e positivas.
  - v. Destas análises resultou a tabela Tabela 3 - Atitudes X Chave de deteção

**Tabela 3 - Atitudes X Chave de deteção**

<b>Atitude</b>	<b>Pos</b>	<b>Flex</b>
Apreciação	Adj	fs ms
Apreciação	M	Mn
Apreciação	M	Mp
Afeto	V	J:1s

1. No decorrer da análise às categorias gramaticais, foi observado que a melhor forma de deteção do tipo de atitude “Julgamento”, conforme definição de (Martin & White, 2005) seria a sua ligação direta ao conceito “Staff”, o que resulta que sempre que o conceito “Staff” é detetado, o algoritmo indexa diretamente esse conceito ao tipo de atitude “Julgamento”.
2. Com o desenvolvimento do algoritmo, detetou-se a impossibilidade de conseguir indexar diretamente os tipos de atitude “Apreciação” e “Afeto” aos conceitos detetados nas frases. A forma encontrada para minimizar esse problema passou por em todos os casos em que apenas se deteta um conceito e um desses tipos de atitude, forçar sua indexação.

3. No caso de existir o conceito Staff, estando este associado ao tipo de atitude “Julgamento”, este não é considerado no processo descrito no ponto anterior
- b. A saída final deste processo apresenta cada frase classificada quanto ao seu conceito e atitude, quando aplicável

#### 4.2. Desenvolvimento de algoritmo<sup>3</sup> para deteção de conceitos

Abaixo descreve-se as principais etapas que o algoritmo percorre para a deteção de conceito em frases

- Obtém lista de *keywords* que definem o conceito
- Obtém lista de Frases associadas à Análise
- Prepara expressão regular para ser *case insensitive*
- Percorre cada frase da Análise
  - a. Normaliza frase, retirando caracteres especiais
- Acede à lista de palavras-chave que definem os conceitos
  - a. Define que palavra-chave do conceito a pesquisar na Frase
- A Expressão regular que é utilizada para encontrar a palavra-chave do conceito na frase
  - a. Funciona encontrando uma frase onde se encontra a palavra-chave definida na lista de palavras-chave.
    - i. `string strRegex = @"(.*" + seekText + @"[a-z]*(?=[\.\|\!|\?|\,|\;|\:|\n|\r]).*)|(.*" + seekText + @"[a-z]*(?=\s).*)|(.*" + seekText + @"[a-z]*$)";`
- Em seguida Itera sobre sequência de matches que ocorreram da *keyword* sobre a frase
- Quando deteta Match com sucesso, marca como *keyword* de conceito encontrado.
- Verifica se o conceito detetado pela *keyword* já existe
  - a. Apenas considera se não existir, visto que não se pretende ter conceitos repetidos.

#### 4.3. Desenvolvimento de algoritmo<sup>4</sup> para deteção de Atitudes

Abaixo descreve-se as principais etapas que o algoritmo percorre para a deteção de atitudes em frases.

- Obtém lista de pares Flex e Pos que definem o sentimento
- Obtém lista de Frases associadas à Análise
- Itera sobre cada frase da Análise
  - a) Verifica se existe algum conceito “Staff” na Frase.

---

<sup>3</sup> Anexo 4 – Algoritmo para deteção de conceitos

<sup>4</sup> Anexo 5 – Algoritmo para deteção de Atitudes

- b) Caso o objeto não esteja vazio, é porque existe o conceito “Staff” sendo feita a criação automática do Sentimento “Julgamento”.
- c) Gravar Sentimento “Julgamento” associado ao Conceito “Staff”
- d) Adicionar à lista o Sentimento para que o mesmo não seja repetido na frase
- e) Verifica se existem apenas 2 conceitos, onde caso se confirme é marcado o 2º como target para receber os novos sentimentos. O 1º é assumido que será o Staff.
- Processo de detecção dos sentimentos com base na lista de SentiLex
  - a) Dividir a frase em palavras, separando pelos caracteres “ “, “,”, “.”, “:”
    1. Iteração pelas palavras da frase
      - a. Ignorar palavras com menos de 3 caracteres
      - b. Normalizar Palavra retirando caracteres especiais
    2. Pesquisar Palavra na tabela SentiFlexLex
      - a. Se objeto for null, indica que não foi encontrada nenhuma palavra na tabela SentiFlexLex, pelo que passa para a próxima palavra.
    3. Encontrada Palavra na tabela SentiFlexLex, verificar agora se a mesma se encontra na tabela de definição de Sentimentos
      - a. Verifica se Pos e Flex da palavra existem na definição de sentimento.
      - b. Obtém todas as definições de sentimento que tenham o POS da palavra encontrada na tabela SentiFlexLex.
      - c. Caso a lista encontrada tenha elementos vai tentar encontrar o Flex.
      - d. Separar Lista Flex (palavra encontrada) pelo caracter “|”
      - e. Iterar sobre lista de palavras
      - f. Limita a lista agora também por cada Flex existente.
      - g. Verifica se após todas as restrições impostas sobra a tabela de definição de sentimentos existe ainda alguma definição.
      - h. Guarda a definição de sentimento encontrada
    4. Verifica se foi encontrada alguma definição de sentimento
      - a. Verifica se já existe algum sentimento igual na frase, para não permitir criar repetições
      - b. Obtém Informação de sentimento, com base na definição de sentimento encontrada
      - c. Verifica se existe um conceito target (quando existem 2 conceitos). Se == 0 não tem target e será necessário criar conceito “Não Aplicável”
      - d. Verifica se apenas existe 1 conceito (excluindo não aplicável) para assim associar os sentimentos ao conceito.
      - e. Guarda conceito que deve ficar associado ao sentimento.
      - f. Procura conceito “Não Aplicável” associado à frase.
      - g. Se não existe, tem de ser criado novo conceito
      - h. Se existe, utilizar esse ID de conceito
      - i. Gravar sentimento

Foi ainda desenvolvido código<sup>5</sup> com o objetivo de ser possível a criação de conjuntos de frases baseados em conjuntos já anteriormente classificados manualmente, de modo a ser possível sujeitar esses novos conjuntos a análises automáticas de conceitos e julgamentos para posterior comparação dos resultados obtidos automaticamente com os resultados obtidos por avaliação manual

---

<sup>5</sup> Algoritmo para geração de conjuntos sem avaliação duplicando já existentes disponível no Anexo3 –

#### 4.4. Implementação da plataforma de desenvolvimento

No sentido de agilizar as diferentes fases de desenvolvimento do algoritmo, e por ser necessário disponibilizar a voluntários conjuntos de frases para análise manual e posterior comparação com análises automáticas, optou-se pelo desenvolvimento de uma aplicação MVC ASP.NET (*Model, View, Controller*) (Microsoft, 2013).

Foi assim possível disponibilizar um interface web, que permite não só o envio das análises aos voluntários, como também gerar automaticamente conceitos e sentimentos baseado em conjuntos de frases gerados aleatoriamente, também por esta aplicação.

Os passos mais significativos para o desenvolvimento retratam-se em seguida:

- Criação de um modelo de dados
- Importação para BD das frases objeto de análise
- Criação de tabela de conceitos (Hotel, Staff, Localização, Quarto, Pequeno-almoço)
- Criação de tabela de sentimento (Apreciação, Julgamento e Atitude)
- Criação de tabelas para guardar dados sujeitos a análises do tipo: (Automáticas e Manuais)
- Criação das restantes tabelas auxiliares
- Desenvolvimento de código para gestão das bases de dados
  - a. O MCV permite que todas as relações entre as tabelas sejam geradas e geridas através de código desenvolvido em C#
- Desenvolvimento dos interfaces web sendo os principais:
  - a. - Home Page
  - b. - Criação aleatória de Análises
  - c. - Lista de análises criadas
  - d. - Detalhe do conjunto de análise
  - e. - Detalhe da classificação da frase

Resultou este processo numa aplicação que pode ser consultada em <http://marcolss-001-site1.smarterasp.net/>, estando os *screen shots* disponíveis em:

- Anexo 6 - Fig. 7 - Home Page
- Anexo 7 - Fig. 8 - Criação aleatória de Análises
- Anexo 8 -Fig. 9- Lista de análises criadas
- Anexo 9 -Fig. 10 - Detalhe do conjunto de análise
- Anexo 10 -Fig. 11 - Detalhe da classificação da frase

#### 4.5. Treino e teste dos algoritmos

Cada uma das análises manuais produzidas pelos voluntários quanto a conceito e atitude foi comparada com os resultados gerados pelo algoritmo sobre os mesmos conjuntos de frases. Foi utilizado o mesmo processo para o conjunto de testes, uma avaliação manual e uma avaliação automática do mesmo conjunto. Para cada conjunto foi selecionado um avaliador considerado especialista. No conjunto 1, uma técnica de turismo e para o conjunto 2 um avaliador que apresentou a sua tese de mestrado com um estudo sobre a utilização da Teoria de Avaliação para análise de comentários *on-line* na área da restauração. Foram ainda solicitadas a avaliação de mais quatro voluntários, dois para cada conjunto de frases sem qualquer experiência na área, para um conjunto final de três avaliadores por conjunto. Foram então consideradas como referências as avaliações destes especialistas, sendo essas avaliações comparadas tanto com as avaliações dos restantes avaliadores como com as avaliações automáticas.

No caso do conjunto de treino, foi considerada como referência a avaliação manual feita por o autor. As comparações entre análises manuais e automáticas (três manuais para os conjuntos 1 e 2, contra detecções automáticas para o conjunto 1 e 2) foram comparadas utilizando a fórmula “*procv*” do Excel. Foram criados índices nas frases na base de dados para esse efeito.

#### 4.6. Análise de resultados

A folha de dados com todos os cálculos e comparações está disponível *on-line* em [http://marcolss-001-site1.smarterasp.net/files/Teste\\_Finais.xlsx](http://marcolss-001-site1.smarterasp.net/files/Teste_Finais.xlsx)

A análise de resultados foi baseada na fórmula que apresenta o resultado *f-score* que devolve a exatidão do teste comparativo entre dois conjuntos.

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

A precisão (*precision*) é a fração de documentos recuperados que são relevantes para a pesquisa.

$$\text{precision} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{retrieved documents}\}|}$$

A abrangência (*recall*) é a fração dos documentos relevantes para a consulta que são recuperados com sucesso.

$$\text{recall} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{retrieved documents}\}|}$$

#### 4.7. Resultados do conjunto de treino

Esta secção apresenta os resultados obtidos para o conjunto de treino utilizado para análise das frases e desenvolvimento dos algoritmos para deteção de conceitos e atitudes.

##### 4.7.1. Deteção de Conceitos

As tabelas abaixo apresentem os valores obtidos na deteção de conceitos por o algoritmo quando comparado com o conjunto de treino classificado manualmente por o autor.

**Tabela 4 - Resultados do conjunto de treino para deteção de Conceitos.**

	Conceito Manual	Conceito Automático	Conceito $\cap$ manual	P 1	R 1	F-Score
Staff	96	86	77	0,895	0,802	0,846
Quarto	146	154	132	0,857	0,904	0,880
Localização	114	77	75	0,974	0,658	0,785
Hotel	93	111	65	0,586	0,699	0,637
pequeno-almoço	57	44	42	0,955	0,737	0,832
<b>Total</b>				<b>0,853</b>	<b>0,760</b>	<b>0,796</b>

O resultado total da avaliação apresenta um *F-Score* total de 0,796, sendo este valor prejudicado por os valores conseguidos na deteção do conceito “Hotel”, que podem se revelar mais fracos devido este ser um conceito demasiado abrangente do domínio analisado.

##### 4.7.2. Deteção de Atitudes

As tabelas abaixo apresentadas refletem os valores obtidos na deteção de atitudes no conjunto de treino, quando comparada a performance do algoritmo com a avaliação manual do autor.

**Tabela 5 - Resultados do conjunto de treinos para detecção de Atitudes**

	Atitude Manual	Atitude Automática	Atitude Man $\cap$ Atitude Auto	P 1	R 1	F-Score
<b>Apreciação</b>	<b>296</b>	<b>315</b>	<b>231</b>	<b>0,733</b>	<b>0,780</b>	<b>0,756</b>
<b>Julgamento</b>	<b>88</b>	<b>86</b>	<b>72</b>	<b>0,837</b>	<b>0,818</b>	<b>0,828</b>
<b>Afeto</b>	<b>44</b>	<b>31</b>	<b>28</b>	<b>0,903</b>	<b>0,636</b>	<b>0,747</b>
<b>Total</b>				<b>0,825</b>	<b>0,745</b>	<b>0,777</b>

A detecção de Atitudes releva-se neste conjunto treino com um resultado *f-Score* de 0.777, destacando-se a detecção de julgamento como sendo a que melhor valor apresenta com *um f-score* de 0.828.

#### 4.8. Resultados dos conjuntos de avaliação 1 e 2

Para o conjunto 1 o avaliador especialista é identificado como User3In e é a sua avaliação que se apresenta como referência para comparação com os restantes resultados. No conjunto 2 o avaliador especialista é identificado como User1Ex. As próximas subsecções apresentam a comparação dos resultados obtidos entre os avaliadores especialistas (User2In e User1Ex) e o algoritmo, bem como entre os especialistas e os restantes avaliadores do conjunto. Por último apresenta-se uma tabela de concordância total entre os 3 avaliadores para cada conjunto, onde a relevância dos resultados é dada por a coluna que traduz o valor absoluto (a intersecção) do número de itens em que todos avaliam da mesma forma para conceitos e atitudes (User1...  $\cap$  User2...  $\cap$  User3...).

##### 4.8.1. Detecção de conceitos

Abaixo apresentam-se os valores obtidos na detecção automática conceitos quando comparada com os avaliadores especialistas.

**Tabela 6 - Detecção conceito para conjunto 1 User3In Vs. Automático**

	Conceito User3In	Conceito Automático	Conceito User3In $\cap$ Conceito Auto	P 1	R 1	F-Score
<b>Staff</b>	<b>59</b>	<b>58</b>	<b>46</b>	<b>0,793</b>	<b>0,780</b>	<b>0,786</b>
<b>Quarto</b>	<b>65</b>	<b>89</b>	<b>63</b>	<b>0,708</b>	<b>0,969</b>	<b>0,818</b>
<b>Localização</b>	<b>71</b>	<b>48</b>	<b>46</b>	<b>0,958</b>	<b>0,648</b>	<b>0,773</b>
<b>Hotel</b>	<b>43</b>	<b>69</b>	<b>35</b>	<b>0,507</b>	<b>0,814</b>	<b>0,625</b>
<b>pequeno-almoço</b>	<b>30</b>	<b>22</b>	<b>21</b>	<b>0,955</b>	<b>0,700</b>	<b>0,808</b>
<b>Total</b>				<b>0,784</b>	<b>0,782</b>	<b>0,762</b>

**Tabela 7 - Detecção conceito para conjunto 2 User1Ex Vs. Automático**

	Conceito User1Ex	Conceito Automático	Conceito User1Ex $\cap$ manual	P 1	R 1	F-Score
Staff	69	66	54	0,818	0,783	0,800
Quarto	50	74	44	0,595	0,880	0,710
Localização	60	46	42	0,913	0,700	0,792
Hotel	48	77	20	0,260	0,417	0,320
pequeno-almoço	34	30	26	0,867	0,765	0,813
Total				0,690	0,709	0,687

Na análise das tabelas anteriores, onde o algoritmo é comparado na deteção de atitudes com os avaliadores especialistas, denota-se uma quebra nos valores de *f-score* quando comparados com o conjunto de teste. É também detetável que o conceito hotel é o responsável por grande parte dessa quebra, com 0,62 no conjunto 1 e apenas 0,32 no conjunto 2.

Para uma melhor enquadramento dos resultados obtidos na identificação de conceitos por o algoritmo quando comparada a sua performance com a avaliação dos especialistas, seguem-se as tabelas que refletem a comparação entre o avaliador especialista de cada conjunto e os restantes avaliadores de cada conjunto.

**Tabela 8 - Detecção conceito para conjunto 1 User3In Vs. User1In**

	Conceito User3In	Conceito User1In	Conceito User3In $\cap$ User1In	P 1	R 1	F-Score
Staff	59	46	42	0,712	0,913	0,800
Quarto	65	52	46	0,708	0,885	0,786
Localização	71	50	47	0,662	0,940	0,777
Hotel	43	76	28	0,651	0,368	0,471
pequeno-almoço	30	27	25	0,833	0,926	0,877
Total				0,713	0,806	0,742

**Tabela 9 - Detecção conceito para conjunto 1 User3In Vs. User2In**

	Conceito User3In	Conceito User2In	Conceito User3In $\cap$ User2In	P 1	R 1	F-Score
Staff	59	56	51	0,911	0,864	0,887
Quarto	65	68	57	0,838	0,877	0,857
Localização	71	70	63	0,900	0,887	0,894
Hotel	43	110	35	0,318	0,814	0,458
pequeno-almoço	30	31	27	0,871	0,900	0,885
Total				0,768	0,869	0,796

**Tabela 10 - Detecção conceito para conjunto 2 User1Ex Vs. User3Ex**

	Conceito User1Ex	Conceito User3Ex	Conceito User1Ex $\cap$ User3Ex	P 1	R 1	F-Score
Staff	69	72	64	0,889	0,928	0,908
Quarto	50	68	45	0,662	0,900	0,763
Localização	60	71	57	0,803	0,950	0,870
Hotel	48	116	40	0,345	0,833	0,488
pequeno-almoço	34	40	32	0,800	0,941	0,865
Total				0,700	0,910	0,779

**Tabela 11 - Detecção conceito para conjunto 2 User1Ex Vs. User2Ex**

	Conceito User1Ex	Conceito User2Ex	Conceito User1Ex $\cap$ User2Ex	P 1	R 1	F-Score
Staff	69	71	61	0,859	0,884	0,871
Quarto	50	76	43	0,566	0,860	0,683
Localização	60	58	49	0,845	0,817	0,831
Hotel	48	132	43	0,326	0,896	0,478
pequeno-almoço	34	34	30	0,882	0,882	0,882
Total				0,696	0,868	0,749

Na comparação direta entre a avaliação dos especialistas com os restantes avaliadores, mais uma vez notamos uma clara discordância na identificação do conceito hotel, com valores sempre abaixo de 0.50 em todos os casos. Note-se ainda que os valores finais de *f-score* não se enquadram com os valores conseguidos com a comparação direta entre os avaliadores especialistas e os algoritmos e os mesmos especialistas e os restantes avaliadores, sendo mais clara essa aproximação no conjunto 1.

**Tabela 12 - Valores de concordância do conjunto 1 para conceitos entre os 3 avaliadores**

	Conceito User1In	Conceito User2In	Conceito User3In	Conceito User1In $\cap$ User2In $\cap$ User3In
Staff	46	56	59	39
Quarto	52	68	65	41
Localização	50	70	71	44
Hotel	76	110	43	25
pequeno-almoço	27	31	30	23

**Tabela 13 - Valores de concordância do conjunto 2 para conceitos entre os 3 avaliadores**

	Conceito User1Ex	Conceito User2Ex	Conceito User3Ex	Conceito User1Ex $\cap$ User2Ex $\cap$ User3Ex
Staff	69	71	72	59
Quarto	50	76	68	42
Localização	60	58	71	48
Hotel	48	132	132	38
pequeno-almoço	34	34	40	30

A Tabela 12 e Tabela 13 permitem entender os níveis de concordância que existem entre os 3 avaliadores (quais as frases em que todos concordam na identificação do mesmo conceito) através da coluna “Conceito User1Ex  $\cap$  User2Ex  $\cap$  User3Ex”.

#### 4.8.2. Detecção de Atitudes

Da mesma forma que para conceitos, abaixo apresentam-se os valores obtidos por o algoritmo na deteção automática de atitudes quando comparado com os avaliadores especialistas de cada conjunto.

**Tabela 14 - Detecção Atitude User3In Vs. Automático**

	Atitude User3In	Atitude Automática	Atitude User3In $\cap$ Atitude Auto	P 1	R 1	F-Score
Apreciação	237	189	166	0,878	0,700	0,779
Julgamento	57	58	45	0,776	0,789	0,783
Afeto	6	17	1	0,059	0,167	0,087
Total				0,571	0,552	0,550

**Tabela 15 - Detecção Atitude User1Ex Vs. Automático**

	Atitude User1Ex	Atitude Automática	Atitude User1Ex $\cap$ Atitude Auto	P 1	R 1	F-Score
Apreciação	162	190	116	0,611	0,716	0,659
Julgamento	63	66	46	0,697	0,730	0,713
Afeto	3	14	2	0,143	0,667	0,235
Total				0,483	0,704	0,536

Os valores de *f-score* conseguidos em ambos os conjuntos avaliados por especialistas apresentam uma proximidade relevante com 0,550 e 0,536 respetivamente. É também imediatamente identificável que a atitude que mais afeta negativamente estes resultados é o afeto.

Aa tabelas abaixo apresentam a comparação para detecção de atitudes entre os avaliadores especialistas e os restantes avaliadores para cada um dos conjuntos, de forma a ser possível enquadrar os resultados obtidos por o algoritmo quando sujeito à comparação com os avaliadores especialistas.

**Tabela 16 - Detecção Atitude User3In Vs. User 1In**

	Atitude User3In	Atitude User1In	Atitude User3In $\cap$ User1In	P 1	R 1	F-Score
<b>Apreciação</b>	<b>237</b>	<b>227</b>	<b>208</b>	<b>0,878</b>	<b>0,916</b>	<b>0,897</b>
<b>Julgamento</b>	<b>57</b>	<b>41</b>	<b>39</b>	<b>0,684</b>	<b>0,951</b>	<b>0,796</b>
<b>Afeto</b>	<b>6</b>	<b>23</b>	<b>2</b>	<b>0,333</b>	<b>0,087</b>	<b>0,138</b>
<b>Total</b>				<b>0,632</b>	<b>0,651</b>	<b>0,610</b>

**Tabela 17 - Detecção Atitude User3In Vs. User 2In**

	Atitude User3In	Atitude User2In	Atitude User3In $\cap$ User2In	P 1	R 1	F-Score
<b>Apreciação</b>	<b>237</b>	<b>250</b>	<b>213</b>	<b>0,852</b>	<b>0,899</b>	<b>0,875</b>
<b>Julgamento</b>	<b>57</b>	<b>18</b>	<b>14</b>	<b>0,778</b>	<b>0,246</b>	<b>0,373</b>
<b>Afeto</b>	<b>6</b>	<b>34</b>	<b>1</b>	<b>0,029</b>	<b>0,167</b>	<b>0,050</b>
<b>Total</b>				<b>0,553</b>	<b>0,437</b>	<b>0,433</b>

**Tabela 18 - Detecção Atitude User1Ex Vs. User 3Ex**

	Atitude User1Ex	Atitude User3Ex	Atitude User1Ex $\cap$ User3Ex	P 1	R 1	F-Score
<b>Apreciação</b>	<b>162</b>	<b>246</b>	<b>157</b>	<b>0,638</b>	<b>0,969</b>	<b>0,770</b>
<b>Julgamento</b>	<b>63</b>	<b>50</b>	<b>42</b>	<b>0,840</b>	<b>0,667</b>	<b>0,743</b>
<b>Afeto</b>	<b>3</b>	<b>21</b>	<b>1</b>	<b>0,048</b>	<b>0,333</b>	<b>0,083</b>
<b>Total</b>				<b>0,509</b>	<b>0,656</b>	<b>0,532</b>

Tabela 19 - Detecção Atitude User1Ex Vs. User 2Ex

	Atitude User1Ex	Atitude User3Ex	Atitude User1Ex $\cap$ User3Ex	P 1	R 1	F-Score
Apreciação	162	249	156	0,627	0,963	0,759
Julgamento	63	48	41	0,854	0,651	0,739
Afeto	3	2	0	0,000	0,000	0,000
Total				0,494	0,538	0,499

Os valores totais de *f-score* conseguidos entre os especialistas e entre os restantes avaliadores apresentam números que se aproximam também dos obtidos entre especialistas e o algoritmo. Mais uma vez os números demonstram a maior discordância e dificuldade em identificar a atitude afeto, onde encontramos mesmo valores de 0.

Tabela 20 - Valores de concordância do conjunto 1 para Atitudes entre os 3 avaliadores

	Atitude User1In	Atitude User2In	Atitude User3In	Atitude User1In $\cap$ User2In $\cap$ User3In
Apreciação	227	250	237	193
Julgamento	41	18	57	10
Afeto	23	34	6	0

Tabela 21 - Valores de concordância do conjunto 2 para Atitudes entre os 3 avaliadores

	Atitude User1Ex	Atitude User2Ex	Atitude User3Ex	Atitude User1Ex $\cap$ User2Ex $\cap$ User3Ex
Apreciação		249	246	152
Julgamento	63	48	50	33
Afeto	3	2	21	0

Da mesma forma que na Tabela 12 e Tabela 13, podemos identificar aqui os níveis de concordância entre os avaliadores, desta vez para Atitudes.

#### 4.9. Análise de resultados

Em ambos os conjunto avaliados, o conceito “Hotel” é claramente o que apresenta resultados mais fracos quando comparados com os restantes conceitos. Isto pode ser atribuído ao facto de ser um conceito demasiado abrangente o que torna mais difícil a

sua deteção. No caso da atitude, existe também uma clara dificuldade na deteção de afeto. Esta situação tinha já sido antecipada na abordagem teórica descrita na secção 2.2. Como também descrito na secção 2.2, quase todas as atitudes podem ter em si inerentemente um afeto. Por esse facto e após a análise de resultados, questiona-se neste momento a relevância da identificação deste tipo de atitude.

Numa visão global dos resultados é gratificante entender que os valores obtidos entre as avaliações automáticas e os especialistas, para conceitos e atitudes, quando comparados com os valores registados entre os avaliadores especialistas e os restantes avaliadores para ambos os conjuntos não apresentam divergências assinaláveis ao ponto de inviabilizar a metodologia utilizada. Na realidade, as dificuldades registadas por o algoritmo na deteção de determinados itens, sejam estes conceitos ou atitudes, vão ao encontro das mesmas dificuldades encontradas por os próprios avaliadores humanos quando avaliados entre si.

Se na deteção de conceitos os números se apresentam relativamente distribuídos, é relevante notar que para atitudes, a apreciação é a mais detetada na amostra avaliada, o que pode indicar que a grande maioria dos comentários são apreciações aos conceitos identificados, logo seguido de julgamento que denota a importância que é dada ao serviço que é prestado nos hotéis por parte dos funcionários.

## **5. Considerações finais**

Segundo (M. Chaves & Picoto, 2012), o processamento de dados textuais permite oferecer um conhecimento mais refinado para auxiliar a uma tomada de decisão. Ainda segundo os mesmos autores, comentários *on-line* podem ser expressos das mais diversas formas, utilizando diferentes palavras para expressar sentimentos positivos, negativos ou neutrais. Para tal é necessário capturar tantas informações quanto for possível sobre o produto e serviço em análise no comentário, sendo para isso necessário primeiro entender cada frase presente no comentário. No mesmo sentido, (Taboada & Grieve, 2004) nos dizem que acreditam que a Teoria de Avaliação ajuda a categorizar opiniões contidas em textos, referindo-se estas a objetos, emoções ou comportamentos.

Este trabalho pretendeu precisamente ser um contributo para aumentar o nível de granularidade da informação extraída de um comentário ou frase em pequenos textos informais em Português, através da deteção do tipo atitude que está associado ao sentimento detetado nas frases que constituem o comentário, baseado na Teoria de Avaliação descrita na Secção 2.2.

No decorrer do mesmo entendeu-se ainda que seria importante enquadrar também a deteção de conceitos, uma vez que pareceu ao autor mais motivador apresentar

resultados que permitissem enquadrar cada tipo de atitude, ou várias atitudes, a um determinado objeto (neste caso um conceito que pertence-se ao âmbito da hotelaria).

Assim assume-se que os algoritmos desenvolvidos para a deteção de conceitos e atitudes em comentários *on-line* em português no âmbito da hotelaria possam ser um primeiro passo nesta área para a língua portuguesa, na medida em que trabalhos semelhantes foram realizados para a língua inglesa e em outros âmbitos por (Read and Carroll, 2012; Taboada and Grieve, 2004).

### **5.1. Oportunidades de melhoria e limitações do trabalho**

Sendo este trabalho uma primeira abordagem a um tema já de alguma forma estudada para outras línguas e âmbitos que não o português e a hotelaria existem sem dúvida diversas oportunidades de melhoria. Nomeadamente a utilização completa de *Hontology* (Chaves et al., 2012) para a deteção de conceitos no âmbito da hotelaria, sendo que neste trabalho foram apenas utilizados os conceitos considerados mais relevantes por o autor.

Ao conseguir um *corpus* de análise mais alargado, seria possível um estudo mais aprofundado das dicotomias presentes em cada frase que possam levar à deteção de atitudes de uma forma mais rigorosa. É também importante ter um maior número de avaliadores, especialistas e não especialistas que classifiquem as frases para que seja possível assim ter uma avaliação mais abrangente dos algoritmos, bem como uma maior base de estudo que o permita a sua melhoria.

Revelou-se uma clara limitação a este trabalho os fracos conhecimentos gramaticais do autor, para que fosse possível aferir de uma forma mais correta as variações gramaticas que permitam uma deteção mais fina das atitudes. O apoio de um especialista em gramática e semântica portuguesa teria sido sem dúvida uma mais-valia no desenvolvimento deste trabalho.

Existia inicialmente a ideia de conseguir associar aos resultados apresentados a sua polaridade, positiva ou negativa, mas esse objetivo não foi alcançado devido à complexidade associada à associação de polaridade a cada tipo de atitude. Não sendo esse o foco principal do trabalho, deixou-se esse ponto para um trabalho futuro.

Sendo um projeto em curso, a utilização do SentiLex-PT02 (Silva et al., 2012) foi sem dúvida uma base de trabalho importante, mas como um projeto em evolução demonstrou algumas fragilidades no sentido do seu léxico não se encontrar completo.

A associação de conceitos às atitudes foi mais complexa do que o suposto inicialmente. O que parecia ser um caminho simples revelou-se complicado por não ter sido possível relacionar diretamente um conceito a uma atitude. Foi possível detetar um

ou vários conceitos e uma ou várias atitudes para a mesma frase, mas apenas foi possível relacionar diretamente o conceito *staff* a julgamento, e as restantes atitudes só foi possível relacionar quando uma frase apenas tinha um conceito e uma atitude.

## 5.2. Trabalhos futuros

Como trabalho futuro, propõem-se a utilização de *Hontology* (Chaves et al., 2012), para a deteção de conceitos, permitindo assim um maior detalhe a este nível, por estar baseado num projeto que dá grande detalhe a este domínio.

É também relevante para um trabalho futuro a colaboração de um especialista em gramática e semântica portuguesa, que certamente trará com o seu conhecimento a possibilidade de um estudo mais aprofundado da abordagem léxico-sintática para a deteção de atitudes.

Conseguir uma relação direta através da abordagem léxico-sintática dos conceitos com as atitudes seria também um forte contributo para o desenvolvimento de um projeto futuro nesta área.

O desenvolvimento de um ou mais algoritmos que permitam o relacionamento direto entre uma frase sujeita a análise, o conceito que se encontra associado a essa frase, que por sua vez se relaciona-se diretamente com a atitude aí presente e por fim associar a todo este conjunto Frase/Conceito/Atitude a respetiva polaridade (o sentimento positivo ou negativo inerente ao comentário), será sem duvida objeto aliciante para qualquer trabalho futuro nesta área.

## Bibliografia

- Boiy, E., Hens, P., Deschacht, K., & Moens, M. F. (2007). Automatic sentiment analysis in on-line text. In *Proceedings of the 11th International Conference on Electronic Publishing* (pp. 349–360). Retrieved from [https://www.law.kuleuven.be/icri/publications/892138\\_elpub2007.content.pdf](https://www.law.kuleuven.be/icri/publications/892138_elpub2007.content.pdf)
- Chaovalit, P., & Zhou, L. (2005). Movie review mining: A comparison between supervised and unsupervised classification approaches (p. 112c–112c). Presented at the System Sciences, 2005. HICSS'05. Proceedings of the 38th Annual Hawaii International Conference on, IEEE.
- Chaves, M., de Freitas, L., Souza, M., & Vieira, R. (2012). PIRPO: An Algorithm to deal with Polarity in Portuguese Online Reviews from the Accommodation Sector. *Natural Language Processing and Information Systems*, 296–301.
- Chaves, M., & Picoto, W. (2012). A Multidomain and Multilingual Conceptual Data Model for Online Reviews Representation. Retrieved from <http://repositorio-cientifico.uatlantica.pt/handle/10884/637>
- Chaves, M. S., Freitas, L. A., & Vieira, R. (2012). Hontology: a multilingual ontology for the accommodation sector in the tourism industry. Presented at the Proceedings of the 4th International Conference on Knowledge Engineering and Ontology Development, Barcelona, Spain. Retrieved from <http://repositorio-cientifico.uatlantica.pt/bitstream/10884/654/1/keod-mchavesetal-cr.pdf>
- Chetviorkin, I., & Loukachevitch, N. (2012). Cross-domain opinion word extraction model. *VI Russian Summer School in Information Retrieval, August*, 6–10.

- Ding, X., Liu, B., & Yu, P. S. (2008). A holistic lexicon-based approach to opinion mining (pp. 231–240). Presented at the Proceedings of the international conference on Web search and web data mining, ACM.
- Eggins, S., & Slade, D. (2004). *Analysing casual conversation*. Equinox Publishing Ltd.
- Gomes, R. (2011). *Tecnologias Web 2.0: O caso de Pequenos e Médios Hotéis no distrito de Lisboa*. Universidade Técnica de Lisboa, Instituto Superior de Economia e Gestão.
- Grefenstette, G. (1994). *Corpus-derived First, Second, and Third-order Word Affinities*. Rank Xerox Research Centre.
- Halliday, M. A. K. (1994). *Functional grammar*. London: Edward Arnold.
- Huang, Y., Basu, C., & Hsu, M. K. (2010). Exploring motivations of travel knowledge sharing on social network sites: an empirical investigation of us college students. *Journal of Hospitality Marketing & Management*, 19(7), 717–734.
- Kennedy, A., & Inkpen, D. (2006). Sentiment classification of movie reviews using contextual valence shifters. *Computational Intelligence*, 22(2), 110–125.
- Kumar, A., & Sebastian, T. M. (2012). Sentiment Analysis: A Perspective on its Past, Present and Future. *International Journal of Intelligent Systems and Applications*, 4(10), 1–14. doi:10.5815/ijisa.2012.10.01
- Langacker, R. W. (1985). Observations and speculations on subjectivity. *Iconicity in syntax*, 109–150.
- Liu, B. (2010). Sentiment analysis and subjectivity. *Handbook of Natural Language Processing*, 627–666.
- Lyons, J. (1981). *Language, meaning and context* (Vol. 5923). Fontana London.

- Martin, J. R. (2000). Beyond exchange: APPRAISAL systems in English. *Evaluation in text: Authorial stance and the construction of discourse*, 175.
- Martin, J. R., & White, P. R. R. (2005). *The language of evaluation*. Palgrave Macmillan Great Britain.
- Microsoft. (2013). ASP.NET MVC. *ASP.NET MVC*. Retrieved January 4, 2013, from <http://www.asp.net/mvc/overview>
- Pang, B., & Lee, L. (2008). *Opinion mining and sentiment analysis*. Now Pub.
- Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10* (pp. 79–86). Retrieved from <http://dl.acm.org/citation.cfm?id=1118704>
- Read, J., & Carroll, J. (2012). Weakly-supervised Appraisal Analysis. *Linguistic Issues in Language Technology*, 8.
- Silva, M. J., Carvalho, P., & Sarmiento, L. (2012). SentiLex-PT02. In *Building a Sentiment Lexicon for Social Judgement Mining*. Coimbra.
- Silva, N. R., Lima, D., & Barros, F. (2012). SAPair: Um Processo de Análise de Sentimento no Nível de Característica. Presented at the IV International Workshop on Web and Text Intelligence (WTI - 2012). Retrieved from <http://www.labic.icmc.usp.br/wti2012/artigos/105283.pdf>
- Sobre o TripAdvisor. (2012). Retrieved October 14, 2012, from [http://www.tripadvisor.com.br/pages/about\\_us.html](http://www.tripadvisor.com.br/pages/about_us.html)
- Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011). Lexicon-based methods for sentiment analysis. *Computational Linguistics*, 37(2), 267–307.

- Taboada, M., & Grieve, J. (2004). Analyzing appraisal automatically. In *Proceedings of AAAI Spring Symposium on Exploring Attitude and Affect in Text (AAAI Technical Report SS-04-07)*, Stanford University, CA, pp. 158-161. AAAI Press. Retrieved from <http://www.aaai.org/Papers/Symposia/Spring/2004/SS-04-07/SS04-07-029.pdf>
- Tang, H., Tan, S., & Cheng, X. (2009). A survey on sentiment detection of reviews. *Expert Systems with Applications*, 36(7), 10760–10773.  
doi:10.1016/j.eswa.2009.02.063
- Tivoli Lisboa (Lisboa, Portugal): 460 avaliações - TripAdvisor. (2012). Retrieved January 12, 2013, from [http://www.tripadvisor.com.br/Hotel\\_Review-g189158-d207174-Reviews-Tivoli\\_Lisboa-Lisbon\\_Estremadura.html](http://www.tripadvisor.com.br/Hotel_Review-g189158-d207174-Reviews-Tivoli_Lisboa-Lisbon_Estremadura.html)
- Turney, P. D. (2002). Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews (pp. 417–424). Presented at the Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Association for Computational Linguistics.
- Vangos, P. (2009). *Word Frequency Counter*. Department of Informatic, Athens University of Economics and Business.

## Anexo 1 – Algoritmo para divisão de comentários em frases

```
for (String li : lines) {
    BreakIterator boundary =
        BreakIterator.getSentenceInstance(Locale.forLanguageTag("pt-PT"));

    boundary.setText(li);
    int start = boundary.first();
    for (int end = boundary.next();
        end != BreakIterator.DONE;
        start = end, end = boundary.next()) {
        this.sentence.add(li.substring(start, end));
    }
}
```

## Anexo 2 – Algoritmo para geração aleatória de conjuntos de frases

```
// Obter nº de frases
int NumFrases =
    Convert.ToInt32(WebConfigurationManager.AppSettings["NumberOfFrasesPerAnalyses"].ToString());

// Tabela de Frases
var listFrases = db.Frases.Where(d => d.isTest == 0 && d.Estado == 1).OrderBy(r =>
    Guid.NewGuid()).Take(NumFrases).ToList();

foreach (var frase in listFrases)
    {
        analiseFrases.AnaliseID = lastInsertId;
        analiseFrases.FraseID = frase.ID;
        analiseFrases.Frase = frase.Frase;

        db.AnaliseFrases.Add(analiseFrases);
        db.SaveChanges();

        Frases frases = db.Frases.Find(frase.ID);
        frases.Estado = 0;
        db.Entry(frases).State = EntityState.Modified;
        db.SaveChanges();
    }
```

## Anexo 3 – Algoritmo para geração de conjuntos sem avaliação duplicando já existentes

```
// If Origin exists, recreate same frases
if (analise.OriginID > 0)
{
    // Tabela de Frases
    var listFrases = db.AnaliseFrases.Where(d => d.AnaliseID ==
        analise.OriginID).ToList();

    foreach (var frase in listFrases)
        {
            analiseFrases.AnaliseID = lastInsertId;
            analiseFrases.FraseID = frase.FraseID;
            analiseFrases.Frase = frase.Frase;

            db.AnaliseFrases.Add(analiseFrases);
        }
```

```
        db.SaveChanges();  
    }
```

## Anexo 4 – Algoritmo para detecção de conceitos

### Obter dados da Análise

```
Analise analise = db.Analise.Find(id);
```

### Obtém lista de keywords que definem o conceito - de acordo com versão definida na análise

```
List<ConceitosDefinicao> oDefinicoes = db.ConceitosDefinicao.Where(d =>  
d.VersionID == analise.ConceitoVersionID).ToList();
```

### Obtém lista de Frases associadas à Análise

```
List<AnaliseFrases> oFrases = db.AnaliseFrases.Where(d => d.AnaliseID ==  
id).ToList();
```

### Prepara expressão regular para ser case insensitive

```
RegexOptions myRegexOptions = RegexOptions.IgnoreCase;
```

### Itera sobre cada frase da Análise

```
foreach (var oFrase in oFrases)  
{
```

#### Normaliza frase, retirando caracteres especiais

```
string strTargetString = ReplaceDiacritics(oFrase.Frase);
```

### Itera sobre lista de keywords que definem os conceitos

```
    foreach (var oDefinicao in oDefinicoes)  
    {
```

#### Define Keyword de conceito a pesquisar na Frase

```
        string seekText = oDefinicao.Texto;
```

### Expressão regular que permite encontrar a keyword de conceito na frase

```
        string strRegex = @"(.*)" + seekText + @"[a-  
z]*(?=[\.\!|\?|\,|;|\:|\n|\r]).*|(.*)" + seekText + @"[a-z]*(?=\s).*|(.*)" +  
seekText + @"[a-z]*$";
```

#### Inicialização da expressão regular

```
myRegex = new Regex(strRegex, myRegexOptions);
```

### Itera sobre sequência de matches que ocorreram da keyword sobre a Frase

```
    foreach (Match myMatch in myRegex.Matches(strTargetString))  
    {
```

#### Quando deteta Match com sucesso, interrompe iteração e marca como keyword de conceito encontrada.

```
        if (myMatch.Success)  
        {  
            hasMatch = true;  
            break;  
        }
```

```
    }  
  
    Encontrou keyword de conceito na Frase  
        if (hasMatch)  
        {  
            Verifica se o conceito detectado pela keyword já existe  
            Apenas considera se não existir, visto que não se pretende ter conceitos  
            repetidos.  
                if (!conceitosFound.Exists(delegate(int i) { return i ==  
oDefinicao.ConceitoID; })))  
                {  
                    Guarda no objecto o Conceito encontrado, para ser utilizado na gravação em Base  
                    de Dados  
                    Conceitos conceitoFind = db.Conceitos.Find(oDefinicao.ConceitoID);  
  
                    Grava Conceito  
                    AnaliseConceitos oAnaliseConceitos = new AnaliseConceitos();  
                    oAnaliseConceitos.AnaliseID = id;  
                    AnaliseConceitos.AnaliseFraseID = oFrase.AnaliseFraseID;  
                    oAnaliseConceitos.ConceitoID = oDefinicao.ConceitoID;  
                    oAnaliseConceitos.Conceito = conceitoFind.Nome;  
                    oAnaliseConceitos.UpdateDate = today;  
                    oAnaliseConceitos.UpdateUser =  
                    System.Web.Security.Membership.GetUser().UserName;  
                    db.AnaliseConceitos.Add(oAnaliseConceitos);  
                    db.SaveChanges();  
  
                    conceitosFound.Add(oDefinicao.ConceitoID);  
  
                    Grava Log Conceito  
                    AnaliseConceitosLog oConceitoLog = new AnaliseConceitosLog();  
                    oConceitoLog.AnaliseConceitoID = oAnaliseConceitos.AnaliseConceitoID;  
                    oConceitoLog.Palavra = seekText;  
                    db.AnaliseConceitosLog.Add(oConceitoLog);  
                    db.SaveChanges();  
                }  
            }  
        }  
  
        Update tabela Análise, de modo a indicar que o mesmo já teve os conceitos  
        processados.  
            analise.isProcessed = 1;  
            db.Entry(analise).State = EntityState.Modified;  
            db.SaveChanges();  
  
            return View();  
        }  
    }  
}
```

## Anexo 5 – Algoritmo para deteção de Atitudes

### Obter dados da Análise

```
Analise analise = db.Analise.Find(id);
```

### Obtém lista de pares Flex e Pos que definem o sentimento - de acordo com versão definida na análise

```
List<SentimentosDefinicao> oDefinicoes = db.SentimentosDefinicao.Where(d =>  
d.VersionID == analise.SentimentoVersionID).ToList();
```

**Obtém lista de Frases associadas à Análise**

```
List<AnaliseFrases> oFrases = db.AnaliseFrases.Where(d => d.AnaliseID == id).ToList();
```

**Itera sobre cada frase da Análise**

```
foreach (var oFrase in oFrases)
{
```

...

**Verifica se existe algum conceito "Staff" na Frase.**

```
AnaliseConceitos oFindStaff =
    db.AnaliseConceitos
        .Where(d => d.AnaliseFraseID == oFrase.AnaliseFraseID &&
            d.ConceitoID == (int)tipoConceito.Staff)
        .FirstOrDefault();
```

**Caso o objeto não esteja vazio, é porque existe o conceito "Staff" sendo feita a criação automática do Sentimento "Julgamento".**

```
if (oFindStaff != null)
{
    Gravar Sentimento "Julgamento" associado ao Conceito "Staff"
    AnaliseSentimentos oAnaliSentimentos = new AnaliseSentimentos();
    oAnaliSentimentos.AnaliseConceitoID = oFindStaff.AnaliseConceitoID;
    oAnaliSentimentos.SentimentoID = (int)tipoSentimento.Julgamento;
    oAnaliSentimentos.Sentimento = "Julgamento";
    oAnaliSentimentos.UpdateDate = today;
    oAnaliSentimentos.UpdateUser =
        System.Web.Security.Membership.GetUser().UserName;
    db.AnaliseSentimentos.Add(oAnaliSentimentos);
    db.SaveChanges();
}
```

**Gravar Sentimento Log - palavra \_automatic\_ para identificar que foi uma ação forçada**

```
AnaliseSentimentosLog oAnaliSentimentosLog = new
AnaliseSentimentosLog();
oAnaliSentimentosLog.AnaliseSentimentoID =
oAnaliSentimentos.AnaliseSentimentoID;
oAnaliSentimentosLog.Palavra = "_automatic_";
db.AnaliseSentimentosLog.Add(oAnaliSentimentosLog);
db.SaveChanges();
```

**Adicionar à lista o Sentimento para que o mesmo não seja repetido na frase**  
sentimentosFound.Add(oAnaliSentimentos.SentimentoID);

**Verifica se existem apenas 2 conceitos, onde caso se confirme é marcado o 2º como target para receber os novos sentimentos. O 1º é assumido que será o Staff.**

```
var conceitosRule = db.AnaliseConceitos
    .Where(d => d.AnaliseFraseID == oFrase.AnaliseFraseID &&
        d.ConceitoID != (int)tipoConceito.Staff)
    .ToList();
if (conceitosRule.Count() == 1)
    mainAnaliseConceitoId = conceitosRule[0].AnaliseConceitoID;
}
```

**Afectação de variável com a Frase**

```
string strTargetString = oFrase.Frase;
```

**Processo de deteção dos sentimentos com base na lista de sentilexflex**

**Dividir a frase em palavras, separando pelos caracteres “ “, “,”, “.”, “:”**

```
string[] frasePalavras = strTargetString.Split(new char[] { ' ', ',', '.', ':' }, StringSplitOptions.None);
```

**Iteração pelas palavras da frase**

```
foreach (var frasePalavra in frasePalavras)  
{
```

**Ignorar palavras com menos de 3 caracteres**

```
if (palavra.Length < 3)  
    continue;
```

**Normalizar Palavra retirando caracteres especiais**

```
string palavraPlain = ReplaceDiacritics(palavra);
```

**Pesquisar Palavra na tabela SentiFlexLex**

```
SentilexFlex oFlex = db.SentilexFlex.FirstOrDefault(w => w.Palavra == palavra || w.Palavra == palavraPlain);
```

**Se objeto for null, indica que não foi encontrada nenhuma palavra na tabela SentiFlexLex, pelo que passa para a próxima palavra.**

```
if (oFlex == null)  
{  
    continue;  
}  
else  
{
```

**Encontrada Palavra na tabela SentiFlexLex, verificar agora se a mesma se encontra na tabela de definição de Sentimentos.**

**Verifica se Pos e Flex da palavra existem na definição de sentimento.**

```
foreach (SentimentosDefinicao oDefinicao in oDefinicoes)  
{
```

**Obtém todas as definições de sentimento que tenham o POS da palavra encontrada na tabela SentiFlexLex.**

```
IEnumerable<SentimentosDefinicao> oDetect = oDefinicoes.Where(d => d.Pos.ToLower() == oFlex.Pos.ToLower());
```

**Caso da lista encontrada tenha elementos vai tentar encontrar o Flex.**

```
if (oDetect.Count() > 0)  
{
```

**Separar Lista Flex (palavra encontrada) pelo caracter “|”**

```
string[] flexList = oFlex.Flex.Split('|');
```

**Iterar sobre lista de palavras**

```
foreach (var flex in flexList)  
{
```

**Limita a lista agora também por cada Flex existente.**

```
oDetect = oDetect.Where(d => d.Flex.ToLower().Contains(flex.ToLower()));  
}
```

**Verifica se após todas as restrições impostas sobre a tabela de definição de sentimentos existe ainda alguma definição.**

```
        if (oDetect.Count() > 0)
        {
            Guarda a definição de sentimento encontrada
            foreach (SentimentosDefinicao oFound in
oDetect.Take(1).ToList())
            {
                oSentimento = oFound;
                break;
            }
        }
    }

    Verifica se foi encontrada alguma definição de sentimento
    if (oSentimento != null)
    {
        Verifica se já existe algum sentimento igual na frase, para não
        permitir criar repetições
        if (!sentimentosFound.Exists(delegate(int i) { return i ==
oSentimento.SentimentoID; }))
        {
            Obter Informação de sentimento, com base na definição de
            sentimento encontrada
            Sentimentos sentimentoFind =
db.Sentimentos.Find(oSentimento.SentimentoID);

            Verifica se existe um conceito target (quando existem 2
            conceitos). Se == 0 não tem target e será necessário criar conceito “Não
            Aplicável”
            if (mainAnaliseConceitoId == 0)
            {
                List<AnaliseConceitos> oFind =
                db.AnaliseConceitos
                    .Where(d => d.AnaliseFraseID ==
oFrase.AnaliseFraseID && d.ConceitoID != (int)tipoConceito.NaoAplicavel)
                    .ToList();

                Verifica se apenas existe 1 conceito (excluindo não
                aplicável).. para assim associar os sentimentos ao conceito.
                if (oFind.Count() == 1)
                {
                    Guarda conceito que deve ficar associado ao sentimento.
                    AnaliseConceitoID = oFind[0].AnaliseConceitoID;
                }
                else
                {
                    Procura conceito “Não Aplicavel” assoicado à frase.
                    AnaliseConceitos oAnaliseConceitosFind =
                    db.AnaliseConceitos
                        .Where(d => d.AnaliseFraseID ==
oFrase.AnaliseFraseID && d.ConceitoID == (int)tipoConceito.NaoAplicavel)
                        .FirstOrDefault();

                    Se não existe, tem de ser criado novo conceito
                    if (oAnaliseConceitosFind == null)
                    {
                        AnaliseConceitos oAnaliseConceitos = new
AnaliseConceitos();
                    }
                }
            }
        }
    }
}
```

```
oAnaliseConceitos.AnaliseID = id;
oAnaliseConceitos.AnaliseFraseID =
oFrase.AnaliseFraseID;
oAnaliseConceitos.ConceitoID = 6;
oAnaliseConceitos.Conceito = "Não Aplicável";
oAnaliseConceitos.UpdateDate = today;
oAnaliseConceitos.UpdateUser =
System.Web.Security.Membership.GetUser().UserName;
db.AnaliseConceitos.Add(oAnaliseConceitos);
db.SaveChanges();

AnaliseConceitoID =
oAnaliseConceitos.AnaliseConceitoID;
}
else
{
    Se existe, utilizar esse ID de conceito
    AnaliseConceitoID =
oAnaliseConceitosFind.AnaliseConceitoID;
}
}
} else {
    Se existe, utilizar esse ID de conceito
    AnaliseConceitoID = mainAnaliseConceitoId;
}

Gravar sentimento
AnaliseSentimentos oAnaliSentimentos = new
AnaliseSentimentos();
oAnaliSentimentos.AnaliseConceitoID = AnaliseConceitoID;
oAnaliSentimentos.SentimentoID = sentimentoFind.SentimentoID;
oAnaliSentimentos.Sentimento = sentimentoFind.Sentimento;
oAnaliSentimentos.UpdateDate = today;
oAnaliSentimentos.UpdateUser =
System.Web.Security.Membership.GetUser().UserName;
db.AnaliseSentimentos.Add(oAnaliSentimentos);
db.SaveChanges();

Gravar sentimento Log
AnaliseSentimentosLog oAnaliSentimentosLog = new
AnaliseSentimentosLog();
oAnaliSentimentosLog.AnaliseSentimentoID =
oAnaliSentimentos.AnaliseSentimentoID;
oAnaliSentimentosLog.Palavra = oFlex.Palavra;
db.AnaliseSentimentosLog.Add(oAnaliSentimentosLog);
db.SaveChanges();

Adicionar sentimento à lista, para a mesma não se repetir
sentimentosFound.Add(oAnaliSentimentos.SentimentoID);
}
}
}
}
```

**Atualizar tabela Análise de modo a indicar que os sentimentos foram processados.**

```
    analyse.isProcessed = 2;  
    db.Entry(analise).State = EntityState.Modified;  
    db.SaveChanges();  
  
    return View();  
}
```

## Anexo 6 - Fig. 7 - Home Page

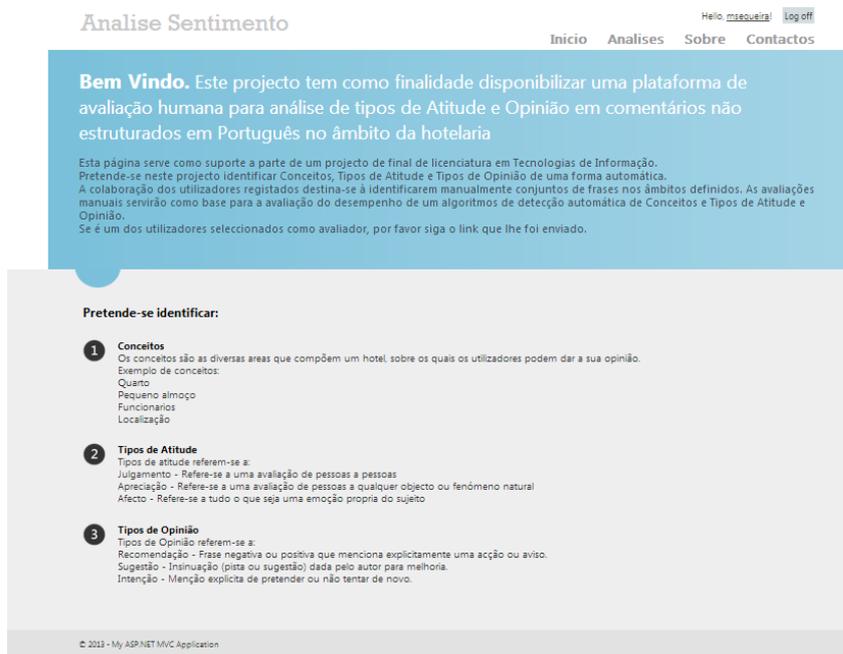


Fig. 7 - Home Page

## Anexo 7 - Fig. 8 - Criação aleatória de Análises

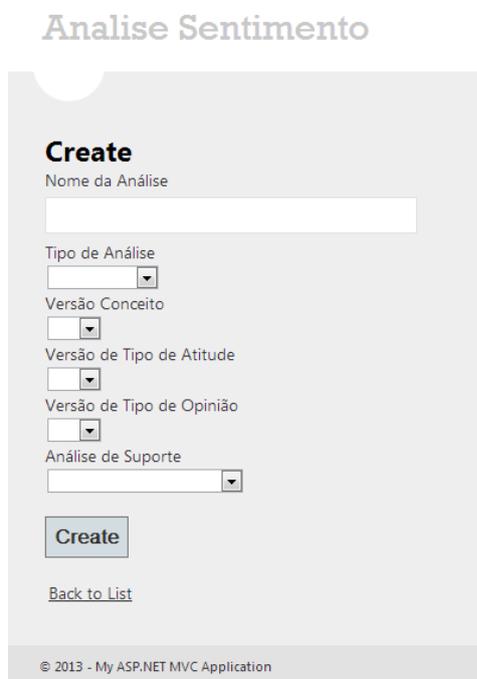


Fig. 8 - Criação aleatória de Análises

## Anexo 8 -Fig. 9- Lista de análises criadas

Analise Sentimento Hello, msequeira! [Log out](#)

[Início](#) [Analises](#) [Sobre](#) [Contacto](#)

### Análises

[Nova Análise](#)

Data	Criado por	Nome	Tipo	Versão Conceito	Tipo Atitude	Tipo Opinião	Baseado em	
9/24/2013 12:00:00 AM	msequeira	Ensaio de Testes	Manual	1.0	1.0	1.0		<a href="#">Detalhe</a>   <a href="#">Relatorio</a>
9/25/2013 3:15:13 PM	msequeira	Conjunto treino Marcirio	Manual	1.0	1.0	1.0	Ensaio de Testes	<a href="#">Detalhe</a>   <a href="#">Relatorio</a>
10/5/2013 7:34:29 AM	msequeira	Conjunto Teste 1.1	Manual	1.0	1.0	1.0		<a href="#">Detalhe</a>   <a href="#">Relatorio</a>
10/5/2013 7:56:58 AM	msequeira	Conjunto Teste 1.2	Manual	1.0	1.0	1.0	Conjunto Teste 1.1	<a href="#">Detalhe</a>   <a href="#">Relatorio</a>
10/5/2013 7:57:55 AM	msequeira	Conjunto Teste 1.3	Manual	1.0	1.0	1.0	Conjunto Teste 1.1	<a href="#">Detalhe</a>   <a href="#">Relatorio</a>
10/5/2013 7:58:52 AM	msequeira	Conjunto Teste 2.1	Manual	1.0	1.0	1.0		<a href="#">Detalhe</a>   <a href="#">Relatorio</a>
10/5/2013 7:59:33 AM	msequeira	Conjunto Teste 2.2	Manual	1.0	1.0	1.0	Conjunto Teste 2.1	<a href="#">Detalhe</a>   <a href="#">Relatorio</a>
10/5/2013 8:00:21 AM	msequeira	Conjunto Teste 2.3	Manual	1.0	1.0	1.0	Conjunto Teste 2.1	<a href="#">Detalhe</a>   <a href="#">Relatorio</a>

Fig. 9- Lista de análises criadas

## Anexo 9 -Fig. 10 - Detalhe do conjunto de análise

Analise Sentimento Hello, msequeira

[Início](#) [Analises](#) [Sobre](#) [Contacto](#)

### Analises > Detalhe

**Nome** : Ensaio de Testes  
**Tipo** : Manual  
**Criado por** : msequeira

Frase	Conceito	Sentimento	Atitude	
A casa de banho podia ser mais quente	Quarto		Sugestão	<a href="#">Editar</a>
pos: A localização mesmo no centro da cidade.	Localização			<a href="#">Editar</a>
pos: Localização	Localização			<a href="#">Editar</a>
pos: Atendimento e localização co centro de Cascais.	Staff			<a href="#">Editar</a>
	Localização			
Os colchões das camas também necessitam de ser trocados, sendo bastante desconfortável dormir em tão más condições.Não há elevador para o primeiro andar.	Quarto		Sugestão	<a href="#">Editar</a>
	Não Aplicável			
Os quartos do primeiro piso onde fiquei não têm acesso pelo elevador e não têm televisão!	Quarto		Sugestão	<a href="#">Editar</a>
Neg:Comodidade péssima.	Hotel			<a href="#">Editar</a>
Quarto muito antigo, colchão velho e ar condicionado estava estragado.	Quarto		Sugestão	<a href="#">Editar</a>
pos: Do parque de estacionamento	Não Aplicável			<a href="#">Editar</a>

Fig. 10 - Detalhe do conjunto de análise

## Anexo 10 -Fig. 11 - Detalhe da classificação da frase

**Analises > Detalhe > Edição**

**Frase**

A casa de banho podia ser mais quente

Notas:

Save

---

1 de 470 [Proximo >](#)

**Editor**

Conceito ?	Tipo de Atitude ?	Tipo de Opinião ?	
Quarto	-- not used --	Sugestão	Save
	-- not used --	-- not used --	
	-- not used --	-- not used --	
-- not used --	-- not used --	-- not used --	Save
-- not used --	-- not used --	-- not used --	Save

Fig. 11 - Detalhe da classificação da frase