

Faculdade de Ciências e Tecnologia  
Departamento de Engenharia Informática

# Reconhecimento Automático de Emoções em Texto com recurso a emojis

Luis Carlos Fernandes Duarte

Dissertação “Reconhecimento Automático de Emoções em Texto com recurso a emojis” no âmbito do Mestrado em Engenharia Informática, especialização em Sistemas Inteligentes orientada pelo Professor Doutor Hugo Gonçalo Oliveira e apresentada à Faculdade de Ciências e Tecnologia / Departamento de Engenharia Informática.

Setembro 2019



UNIVERSIDADE D  
COIMBRA

Esta página foi intencionalmente deixada em branco.

---

## Agradecimentos

Antes de mais queria agradecer aos meus orientadores, o Professor Hugo Oliveira e o Professor Luís Macedo pelo apoio e orientação ao longo deste último ano. Obrigado por estarem sempre prontos para me ajudar e pelos conhecimentos transmitidos tanto profissionalmente quanto pessoalmente.

Para a minha namorada Carlota Reimão que tanto me ajudou nestes últimos tempos, pois foi a tua ajuda que me fez ultrapassar os vários obstáculos que se impuseram durante a escrita desta tese. Desculpa todos os compromissos desmarcados e a pouca disponibilidade ao longo destes meses no qual espero agora compensar.

Também agradecer a todos os que de forma direta ou indirectamente se cruzaram comigo nestes dois anos. Quero que saibam que, independentemente de termos percorrido caminhos diferentes levo-vos a vocês e a esta cidade no coração. Um agradecimento especial à Inês André e ao meu amigo Zé Miguel, pois sem vocês não estaria aqui.

Um pequeno agradecimento para os meus avós que, mesmo já não estando entre nós, os seus valores de trabalho e de perseverança foram um pilar fundamental para que este trabalho fosse possível de realizar.

Por último mas não menos importante quero agradecer aos meus pais por sempre me incentivarem a lutar pelos meus objetivos e me darem as condições possíveis para tal. Obrigado por estarem sempre presentes e pelo vosso apoio incondicional.

Esta página foi intencionalmente deixada em branco.

---

## Resumo

A detecção automática de emoções é utilizada em várias aplicações, na tentativa de resolver problemas relevantes e atuais, nomeadamente nas áreas da saúde, de *marketing* e de interação humana com o computador. Tem sido feito com recurso a inúmeros métodos, sendo que os primeiros passos dados nesta área foram através de câmaras de vídeo. Hoje em dia, com o número crescente de utilizadores e a utilização diária das redes sociais, a detecção de emoções com base em texto, assume cada vez maior importância e melhores resultados.

Textos que os utilizadores publicam nestas redes contêm novas características, que os textos comuns não continham até agora, tais como os *hashtags* ou os *emojis*. Estas características permitem mais facilmente identificar o estado emocional do autor na altura da sua escrita, ou seja, são características a ter em conta no desenvolvimento de modelos para a detecção e classificação automática de emoções, apenas com recurso a texto. A língua inglesa tem sido objeto de estudo neste domínio. Já para o português o trabalho neste âmbito é escasso e particularmente inexistente no que toca à consideração de *emojis*.

Tendo isto em conta, este trabalho abordará o desenvolvimento de modelos de aprendizagem computacional que permitam a classificação de emoções em pequenos textos portugueses. Tais modelos foram baseados em arquiteturas com bons resultados para tarefas de classificação de texto. Exemplos destes modelos são o Naive Bayes, SVM, e ainda redes neuronais LSTM. Esta escolha deve-se ao facto de cada um deles recorrer a formas distintas de efetuar a classificação de dados.

Para cumprir este objetivo, foi recolhido um conjunto de textos curtos a partir da rede social Twitter, anotados automaticamente com base na presença *emojis*. Estes dados foram aplicados ao treino de modelos, para a classificação das emoções expressas no texto. Com os modelos desenvolvidos efetuamos uma série de experiências para verificar qual o modelo mais adequado para esta tarefa. Com as várias experiências aos modelos desenvolvidos retirou-se diversas conclusões sobre a utilização de *emojis* e dos conteúdos partilhados nas redes sociais.

---

## Palavras-Chave

- Detecção de Emoções
- Twitter
- Língua Portuguesa
- Emojis
- Processamento de Linguagem Natural
- Aprendizagem Computacional

Esta página foi intencionalmente deixada em branco.

# Conteúdo

<b>Acrónimos</b>	<b>ix</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Motivação e Objetivos . . . . .	3
1.2 Abordagens . . . . .	5
1.3 Contribuições . . . . .	6
1.4 Estrutura do Documento . . . . .	7
<b>2 Fundamentação Teórica</b>	<b>9</b>
2.1 Modelos de emoções . . . . .	9
2.2 Aprendizagem Computacional . . . . .	13
2.3 Processamento de Linguagem Natural . . . . .	20
<b>3 Trabalho Relacionado</b>	<b>25</b>
3.1 Recolha Automática de Textos e Emoções Associadas . . . . .	25
3.2 Léxicos Afetivos . . . . .	29
3.3 Modelos Utilizados para o Reconhecimento de Emoções . . . . .	32
<b>4 Abordagem</b>	<b>38</b>
4.1 Criação do conjunto de dados . . . . .	38
4.2 Limpeza de dados . . . . .	40
4.3 Análise inicial dos dados . . . . .	43
4.4 Extração de características . . . . .	45
4.5 Modelos para o Reconhecimento Automático de Emoções . . . . .	46
4.5.1 Naive Bayes . . . . .	47
4.5.2 Support Vector Machine . . . . .	48
4.5.3 Long Short Term Memory . . . . .	49
4.6 Disposição de emojis . . . . .	52
<b>5 Resultados Finais</b>	<b>56</b>
5.1 Abordagem Baseline . . . . .	56
5.2 Naive Bayes . . . . .	57
5.3 SVM . . . . .	65
5.4 LSTM . . . . .	69
5.5 Exploração das relações entre emojis . . . . .	74
5.6 Desempenho dos modelos em títulos de notícias . . . . .	77
<b>6 Conclusão</b>	<b>81</b>
6.1 Extração de dados . . . . .	81
6.2 Processamento dos dados . . . . .	82
6.3 Modelos desenvolvidos . . . . .	83
6.4 Discussão geral . . . . .	85

---

6.5 Trabalho Futuro . . . . .	86
<b>Referências</b>	<b>89</b>

Esta página foi intencionalmente deixada em branco.

# Acrónimos

**IHC** Interação Humana com o Computador.

**IoT** Internet of Things.

**LSTM** Long Short Term Memory.

**NLP** Natural Language Processing.

**PCA** Principal Component Analysis.

**POS** Part of Speech.

**SVM** Support Vector Machine.

**t-SNE** T distributed Stochastic Neighbor Embedding.

**TF-IDF** Term Frequency Inverse Document Frequency.

**URL** Uniform Resource Locator.

**VA** Valence e Arousal.

**VAD** Valence, Arousal e Dominance.

**VSM** Vector Space Model.

Esta página foi intencionalmente deixada em branco.

# Lista de Figuras

1.1	Exemplo de um <i>emoticon</i> e de um <i>emoji</i> . . . . .	2
2.5	Formula da probabilidade condicionada . . . . .	14
2.9	Exemplos de processo de NLP . . . . .	21
2.10	Exemplos do método de <i>bag-of-words</i> . . . . .	22
4.1	Distribuição de <i>tweets</i> por emoção . . . . .	42
4.2	Distribuição dos <i>emojis</i> no conjunto de dados . . . . .	43
4.3	Arquitetura da rede LSTM desenvolvida . . . . .	51
5.1	Exemplos de expressões que usem o mesmo <i>emoji</i> . . . . .	60
5.2	Matriz de confusão obtida para a classificação de emoção do modelo Naive Bayes . . . . .	61
5.3	Matriz de confusão para a classificação de emoção com dados balanceados do modelo Naive Bayes . . . . .	64
5.4	Matriz de confusão obtida para a classificação de emoção do modelo SVM . . . . .	67
5.5	Matriz de confusão para a classificação de emoção com dados balanceados do modelo SVM . . . . .	69
5.6	Matriz de confusão para a classificação de emoção do modelo LSTM . . . . .	71
5.7	Matriz de confusão para a classificação de emoção com dados balanceados do modelo LSTM . . . . .	73
5.8	Similaridade entre os <i>emojis</i> obtidos pelo modelo <i>word2vec</i> . . . . .	75
5.9	Exemplos de <i>tweets</i> semelhantes com <i>emojis</i> diferentes . . . . .	75
5.10	Distribuição do novo conjunto de dados a aplicar por emoção . . . . .	77

Esta página foi intencionalmente deixada em branco.

# Lista de Tabelas

2.1	Resultados possíveis durante a avaliação de modelos computacionais . . . .	17
3.1	Tabela da bibliografia existente sobre a recolha automática de dados . . . .	29
3.2	Tabela da bibliografia encontrada referente aos léxicos existentes . . . . .	32
3.3	Tabela da bibliografia encontrada referente aos modelos aplicados . . . . .	36
4.1	Lista de <i>emojis</i> e de <i>hashtags</i> aceites para armazenamento dos <i>tweets</i> . . .	39
4.2	Divisão dos <i>emojis</i> em emoções . . . . .	40
4.3	Divisão dos códigos <i>unicode</i> dos <i>emojis</i> em emoções . . . . .	40
4.4	Exemplos de textos contraditórios . . . . .	41
4.5	Termos mais relevantes por emoção . . . . .	44
4.6	Exemplos de ocorrência de pontuação expressiva . . . . .	46
4.7	Definição dos parâmetros para o modelo Naive Bayes . . . . .	48
4.8	Definição dos parâmetros para o modelo SVM . . . . .	49
4.9	Definição dos parâmetros para a rede LSTM . . . . .	52
4.10	Definição dos parâmetros para o modelo <i>word2vec</i> . . . . .	53
5.1	Resultados obtidos para a tarefa de classificação de emoção usando o léxico NRC . . . . .	57
5.2	Resultados obtidos para a tarefa de classificação de emoção do modelo Naive Bayes . . . . .	58
5.3	Resultados obtidos para a tarefa de previsão de <i>emojis</i> do modelo Naive Bayes . . . . .	59
5.4	Resultados obtidos para a classificação de emoção do melhor modelo Naive Bayes . . . . .	60
5.5	Melhores resultados obtidos da previsão de <i>emojis</i> do modelo Naive Bayes .	62
5.6	Resultados obtidos da previsão dos <i>emojis</i> mais usados do modelo Naive Bayes . . . . .	62
5.7	Resultados obtidos com dados balanceados para a classificação de emoção do modelo Naive Bayes . . . . .	63
5.8	Resultados obtidos para a tarefa de classificação de emoção do modelo SVM	65
5.9	Resultados obtidos para a tarefa de previsão de <i>emojis</i> do modelo SVM . .	66
5.10	Resultados obtidos para a classificação de emoção do melhor modelo SVM .	66
5.11	Melhores resultados obtidos da previsão de <i>emojis</i> do modelo SVM . . . .	67
5.12	Resultados obtidos da previsão dos <i>emojis</i> mais usados do modelo SVM . .	68
5.13	Resultados obtidos com dados balanceados para a classificação de emoção do modelo SVM . . . . .	68
5.14	Resultados obtidos para a classificação de emoção do modelo LSTM . . . .	70
5.15	Resultados obtidos para a tarefa de previsão de <i>emojis</i> do modelo LSTM .	70
5.16	Melhores resultados obtidos da previsão de <i>emojis</i> do modelo LSTM . . . .	71
5.17	Resultados obtidos da previsão dos <i>emojis</i> mais usados do modelo LSTM .	72

5.18	Resultados obtidos com dados balanceados para a classificação de emoção do modelo LSTM . . . . .	72
5.19	Clusters obtidos efetuando o método <i>k-means</i> . . . . .	76
5.20	Resultados obtidos do modelo Naive Bayes para o novo conjunto de dados .	78
5.21	Resultados obtidos do modelo SVM para o novo conjunto de dados . . . . .	78
5.22	Resultados obtidos do modelo LSTM para o novo conjunto de dados . . . . .	78
5.23	Resultados obtidos da abordagem <i>baseline</i> para o novo conjunto de dados .	78

Esta página foi intencionalmente deixada em branco.

# Capítulo 1

## Introdução

As emoções dos seres humanos desde sempre influenciaram as suas ações. Trabalhos com vista à sua previsão automática têm atraído vários investigadores, sendo uma matéria de estudo muito solicitada e de interesse a várias áreas, da saúde ao *marketing*. Vivemos atualmente numa era em que a categorização de todos os aspetos da vida mundana são vistos como progresso. Assim é natural que temas de teor psicológico sejam cada vez mais investigados, de maneira a tentar classificar, ao mais ínfimo detalhe, todas as interações e ações humanas. Estas ações, por sua vez, são influenciadas pelo estado psicológico de cada indivíduo, mais concretamente pelo estado de emoção em que este se encontra (Reisenzein e Weber, 2009).

A questão que se coloca é: como podemos nós classificar as emoções de cada indivíduo, sendo que cada um tem crenças e valores diferentes? Temos ainda diversidades biológicas e neurológicas, bem como percursos sócio-económicos e vivências distintas, que dificultam ainda mais o processo de classificação.

Como as emoções têm influência nas ações executadas por cada indivíduo, é natural que, para cada ação, haja uma tentativa de perceber o seu estado emocional, antes e durante esse momento. O nosso estado emocional tem influência direta na maneira como desempenhamos as nossas ações diárias. Uma pessoa feliz irá dedicar-se mais ao seu trabalho e às suas relações pessoais do que outra num estado emocional de tristeza, como é descrito por Frijda (2004). Um exemplo que realça este ponto, é a interação entre humanos. Duas pessoas felizes têm uma probabilidade mais elevada de ter uma conversa positiva. Do lado oposto, duas pessoas num estado de raiva ou descontentamento terão maior tendência a partir para ações consideradas menos positivas. Em casos mais extremos, esta situação pode chegar a casos de violência ou de agressividade (Smith et al., 2004). Devido a esta possibilidade, é bastante importante a previsão e impedimento destas situações.

As teorias no domínio das emoções, encontram-se divididas em três categorias distintas. As teorias de emoção com desencadeamentos **Psicológicos** consideram que o estado emocional é resultante de respostas biológicas do corpo humano (James, 1884). As teorias de origem **Neurológica** afirmam que certo tipo de ações desencadeadas no cérebro, ao nível das sinapses, produzem diversas emoções (Caridakis, Karpouzis e Kollias, 2008). Por fim, as teorias a nível **Cognitivo** consideram que os nossos pensamentos e ações definem e produzem emoções (Lazarus, 1991). No entanto, apesar de todos estes grupos de estudo de emoções, nunca se chegou a um consenso num modelo de categorização geral, no seio da comunidade científica. Em relação à categorização de um estado emocional, existe uma divisão em dois modelos, os categóricos e os dimensionais. Esta divisão deve-se ao facto de cada um deles fazer a distinção entre estados emocionais de maneira diferente.

Nos modelos categóricos, tenta-se definir um rótulo para cada um destes estados. Existe uma clara preferência pelas categorizações apresentadas por Ekman (1999), com o seu modelo de seis emoções básicas e pela roda de emoções de Plutchik (2001). Por sua vez, no modelo dimensional, os estados emocionais são categorizadas através da sua localização espacial, numa dimensão selecionada. Neste caso, o mais recorrente é a utilização de duas dimensões (modelo VA) utilizando para tal os valores de valência e excitação (tradução dos termos em inglês *valence* e *arousal*). Para além desta, a utilização de três dimensões (modelo VAD) também é comum, acrescentado o valor de dominância (*dominance*) aos dois valores anteriores, descritos em Sreeja e Mahalakshmi (2017).

Inicialmente, os métodos de classificação dos estados emocionais de um indivíduo eram feitos com recurso a câmaras de vídeo, de maneira a conseguir captar as expressões faciais dessa pessoa (Ekman, 1971). Após a implementação deste método, prosseguiu-se explorando outros métodos, como a adição da captação de áudio, para obter resultados mais diversificados e uma classificação mais exata (De Silva, Miyasato e Nakatsu, 1997). Devido à necessidade de captação de dados, com vários dispositivos, o objetivo de uma classificação geral e facilitada não foi possível.

De maneira a tentar resolver este problema, esperou-se pelos progressos da tecnologia e da computação, nomeadamente nas áreas de Inteligência Artificial, da Psicologia e da Interação Humana com o Computador. Todo este avanço científico foi a base de uma área dedicada a este tema, a Computação Afetiva (Cambria, 2016). Desta forma deu-se o início do estudo geral da classificação de vários temas, incluindo a classificação de estados emocionais com recurso a texto (Shivhare e Khethawat, 2012). Com o aprofundamento da investigação nesta área, rapidamente se chegou à conclusão que a deteção destes estados emocionais, usando apenas texto, era muito mais complexa do que inicialmente esperado. Efetuando apenas uma análise textual, as diferenças culturais que existem representam uma dificuldade na generalização da solução. Outros aspetos como a existência de termos ambíguos e contraditórios, bem como a existência de textos e expressões irónicas, aumentam ainda mais a complexidade da classificação de emoções com base em texto.

Hoje em dia, com a simplificação da *social media*, e o seu conseqüente crescimento, levou à sua utilização constante e diária por parte dos seus utilizadores (Perrin, 2015). Devido a esta simplificação, criou-se um novo fenómeno denominado por *microblogging* (Java et al., 2007), que consiste na expressão individual, com recurso a pequenos textos. Não tardou a que surgissem novos métodos, capazes de adicionar mais expressividade aos textos, como foi o caso de *hashtags*, dos *emoticons* e dos *emojis*.

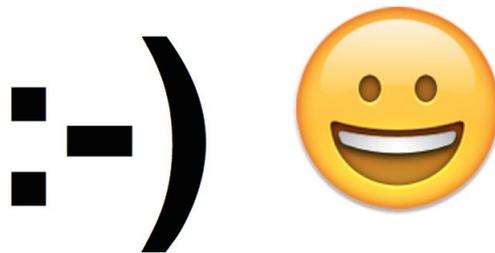


Figura 1.1: Exemplo de um *emoticon* e de um *emoji*

Estes novos métodos acrescentam o valor emotivo, que até agora era necessário equipamentos de captação de vídeo e de áudio para os obter. *Emoticons* e *emojis* fazem uma representação visual para o leitor, dando informação sobre o estado emocional do autor da mensagem. O sentimento/tema geral é providenciado pelos *hashtags*,

representados por uma palavra antecedida pelo carácter cardinal (#).

O *Twitter* e o *Facebook* são duas redes sociais onde a partilha de estados de pequenos textos, vídeos e imagens é possível (Schwartz et al., 2013). No *Twitter* é onde o fenómeno de *microblogging* tem mais impacto. Isto deve-se ao facto de ser uma rede social dedicada exclusivamente à partilha de textos curtos, com um máximo de duzentos e oitenta caracteres denominados por *tweets*. Esta rede conta com mais de 326 milhões de utilizadores, ativos mensalmente.

A quantidade de conteúdo produzido no *Twitter* apresentam uma grande possibilidade para extração de dados. Considerando os avanços da Inteligência Artificial, a classificação dos estados emocionais com base em texto é uma realidade possível. A utilização de modelos de aprendizagem computacional para tarefas de classificação de texto tem levado a resultados surpreendentes. Arquiteturas como Naive Bayes e SVM (Support Vector Machine) e mais recentemente redes neuronais apresentam formas distintas de efetuar a classificação e são dos modelos mais promissores na área.

Atualmente existem sobre o tema, resultados bastante significativos focados para a língua inglesa. Estes trabalhos fornecem uma base para o nosso estudo, bem como uma fonte de inspiração. Para a língua portuguesa, verifica-se o oposto, sendo que existe muito pouco trabalho na área da classificação de emoções em texto. Desta forma espera-se construir um conjunto de modelos de aprendizagem computacional capazes de classificar corretamente a emoção presente num texto de língua portuguesa.

O resto deste capítulo tem como propósito, as motivações pessoais e as abordagens ponderadas, que levaram ao desenvolvimento deste trabalho. Além destas motivações apresentam-se também as principais contribuições, para o progresso da Inteligência Artificial e da abertura do tema para a língua portuguesa.

## 1.1 Motivação e Objetivos

O processo de classificação de emoções transmitidas num texto é complexo mas pode levar a avanços significativos em diversos sistemas. Devido à utilização do inglês como linguagem global, os trabalhos existentes na área são em grande parte dedicados à língua inglesa. Fazendo uma análise às estatísticas sobre a língua portuguesa, verificamos que esta, tem cerca de 221 milhões de falantes nativos<sup>1</sup>. Com este número ocupa o estatuto de sexta língua mais falada mundialmente. Desta forma, o aprofundamento sobre o tema em questão, com foco na língua portuguesa é importante. Esta adaptação seria um grande contributo, devido ao potencial que tem de mudar o dia-a-dia de uma quantidade significativa de pessoas. Outro ponto relevante, é a possibilidade dos modelos resultantes deste trabalho poderem ser integrados em sistemas IoT que vão para além da captação de dados do ambiente, mas que consideram também o estado do utilizador, para melhorar o quotidiano das pessoas.

Um destes sistemas, o ISABELA (Fernandes et al., 2019) tem como objetivo principal a recolha de um conjunto de características físicas e psicológicas de alunos universitários. Exemplos destas características é o número de horas de sono do utilizador, quantidade de mensagens trocadas, utilização de aplicações, entre outras. Desta forma, espera-se relacionar os hábitos de vida dos utilizadores (por exemplo, estudantes), com o seu desempenho (por exemplo, aproveitamento). Assim, o ISABELA ambiciona melhorar a vida dos seus utilizadores, juntamente com o seu desempenho, através de sugestões

<sup>1</sup>Estatísticas disponíveis em <https://www.ethnologue.com/statistics/size>

de melhoramento de certos hábitos diários. O desenvolvimento de modelos para o reconhecimento de emoções possibilitará a integração de características emocionais extraídas de texto produzido pelos utilizadores nos seus *smartphones* e publicado, por exemplo, em redes sociais.

Outra motivação, é a possibilidade do melhoramento de alguns aspetos da vida diária, nomeadamente nas áreas da saúde. Um exemplo concreto, é permitir a psicólogos detetar casos de variações emocionais, de maneira não intrusiva, monitorizando apenas o conteúdo partilhado pelo doente. Para além da saúde, a área de *marketing* é uma realidade, devido à possibilidade de criar campanhas direcionadas, tendo como base, a emoção corrente dos seus clientes.

Juntamente com estes pontos, existe a possibilidade de melhoramento de modelos atualmente implementados, como os sistemas de recomendação. Este melhoramento é possível, devido à não utilização da emoção do utilizador nos sistemas atuais, como ponto importante a considerar.

Para além destes pontos, a pouca existência de bibliografia semelhante dedicada para o português, pode servir como base e referência para futuros trabalhos na área. Temos ainda a constante e crescente utilização da rede social *Twitter*, por parte de falantes da língua portuguesa. Desta forma, a criação de um conjunto de dados de pequenos textos em português é facilitada. Assim, esperamos de uma maneira mais concreta e resumida, atingir os seguintes objetivos:

- Estudo de modelos computacionais utilizados pela comunidade científica para representar emoções.
- Estudo da relação entre emoções e *emojis*, com base em literatura e, se possível, empiricamente.
- Recolha de uma grande quantidade de textos curtos, escritos em português e publicados na rede Twitter, e anotação automática da emoção transmitida, com base na presença de *emojis*.
- Identificação de características relevantes para o reconhecimento de emoções em textos curtos, escritos em português.
- Levantamento de recursos que podem ser utilizados no reconhecimento automático de emoções em textos escritos em português.
- Experimentação de diferentes abordagens de aprendizagem computacional, normalmente aplicados à classificação de texto, que tirem partido das características identificadas para reconhecer a emoção predominante em textos curtos, escritos em português.
- Aplicação das abordagens definidas a diferentes contextos, nomeadamente o reconhecimento da emoção (dada pelo *emoji* originalmente presente) e previsão de *emoji* (identificação do *emoji*), tanto em contexto real como (artificialmente) balanceado.
- Aplicação dos modelos treinados nos dados recolhidos (*tweets* anotados com base na presença de *emojis*) ao reconhecimento de emoções noutros tipos de texto.

## 1.2 Abordagens

O problema de classificação de emoção é um processo bastante complexo, muito devido à dificuldade de qualificar o estado emocional num dado instante. Isto deve-se às diferenças culturais existentes entre povos que tornam o processo difícil de generalizar, que serão abordadas mais à frente no capítulo 3.

Devido à dificuldade de obter conjuntos de dados de grande dimensão para a classificação de emoções, em textos portugueses, é preciso adotar métodos que resolvam este problema. Para o desenvolvimento de qualquer modelo de aprendizagem automática supervisionada, é necessário um grande conjunto de dados e as suas categorias, a explorar num processo de treino. Este processo é crucial para que o modelo consiga generalizar a solução do problema.

Em muitos casos, a construção deste conjunto é bastante dispendiosa em recursos humanos. Por exemplo, a anotação de emoções em texto implica a leitura do texto e a sua interpretação, de forma a identificar o estado emocional que transmite. Para minimizar estes problemas na criação do conjunto de dados, optou-se por um método de extração e anotação automática, onde os textos são retirados de redes sociais. Com recurso a novos métodos de comunicação, neste caso, *emojis*, foi possível efetuar a anotação da emoção correspondente.

Esta anotação automática é possível, devido às características dos *emojis*, como descreve Mohammad e Kiritchenko (2015) e Wood e Ruder (2016) que permitem anotar os dados sem que seja necessária a intervenção humana. Devido à interligação entre uma grande quantidade de *emojis* e o estado emocional, como apresentado por Paiva (2016), iremos verificar se a previsão de um *emoji* num texto será equivalente à previsão da emoção presente nesse mesmo texto. Assim, é possível contornar a problemática relativa à classificação manual da emoção, tornando, ao mesmo tempo, a tarefa do reconhecimento mais objetiva. De forma a confirmar estas relações, iremos classificar a emoção num texto e também prever o *emoji* adequado a este. Dos resultados obtidos destas duas tarefas iremos verificar se estas relações se confirmam. Para construir este conjunto de dados iremos então recorrer às redes sociais, mais concretamente ao *Twitter*<sup>2</sup>, para obter grandes quantidades de textos pessoais. Com os textos obtidos juntamente com estas técnicas de anotação automática de dados conseguiremos obter um conjunto de dados pronto a ser utilizado nas duas tarefas anteriormente referidas.

O resto do trabalho consistirá na utilização do conjunto de dados resultante para efetuar o processo de treino de diversos modelos com arquiteturas distintas. O objetivo principal será a classificação de novos textos numa das seis emoções básicas de Ekman (1999), seguindo a proposta de mapeamento entre *emoji* e emoção de Wood e Ruder (2016), para além de, posteriormente, testarmos uma abordagem semelhante na previsão do *emoji* propriamente dito. Devido ao teor do trabalho consistir na classificação de textos, decidimos explorar três modelos com arquiteturas distintas e adaptadas a este tipo de tarefa, por terem tido bons resultados no passado. Pretendemos assim a verificação de qual a melhor maneira de efetuar a classificação, e tentar apurar se o conteúdo emocional está presente na utilização de termos, expressões ou uma combinação de todas estas particularidades. Para esse efeito adotaram-se três arquiteturas, um modelo baseado em *Naive Bayes*, outro em SVM e o último utilizando uma rede neuronal LSTM. Utilizando o modelo *Naive Bayes* conseguimos verificar se a classificação obtém melhores resultados através da ocorrência sequencial de termos. Através de SVM verificamos se através da

---

<sup>2</sup><https://twitter.com>

representação dos dados num espaço dimensional, consegue-se encontrar fronteiras entre as diferentes emoções. Por fim, com a rede neuronal LSTM exploramos se a capacidade de lembrar certos termos/*emojis* leva a uma boa classificação. Por fim, com os resultados obtidos destes modelos, verificamos qual a melhor maneira de efetuar a classificação de emoções e previsão de *emojis* em textos, bem como a arquitetura que obtém melhores resultados.

### 1.3 Contribuições

As principais contribuições resultantes deste trabalho são as seguintes:

- Três modelos, com arquiteturas diferentes, que permitem a previsão de ocorrência de *emojis* e classificação de emoções. Estes modelos terão como arquiteturas modelos baseados em *Naive Bayes*, SVM e redes neuronais LSTM. A grande parte dos trabalhos realizados nesta área são dedicados à língua inglesa, como demonstrado em Alm, Roth e Sproat (2005) e Binali, Wu e Potdar (2010). A sua adaptação para o português, pode levar a que mais trabalhos apareçam nesta área de estudo. Além disso, promove o desenvolvimento geral da inteligência artificial e da computação afetiva dos países lusófonos.
- Métodos de extração simplificados para a criação de conjuntos de dados, compostos por pequenos textos denominados por *tweets*, escritos em Português. Para além destas características, a adição de mais campos como a localização do autor, número de palavras, data de publicação, entre outras, foi efetuado. Isto pode resultar num grande conjunto de dados categorizados, sem a necessidade da utilização de anotadores humanos e os problemas que estes trazem, para a classificação de texto. Assim é possível a criação de dados, que até agora não estão disponibilizados, e consequentemente permitir que trabalhos futuros não percam demasiado tempo na fase de extração de dados. Para além desta facilidade, existe a possibilidade de recriação da tarefa de *emoji prediction* Barbieri et al. (2018b), mas adaptada para português com este conjunto de dados.
- A produção de um artigo científico relatando parte do trabalho realizado. Devido à escassez de trabalhos semelhantes para a língua portuguesa, a produção de um artigo científico contribuirá para o progresso, na área de classificação de emoção em textos portugueses. Para além desta contribuição, trabalhos futuros terão nesse artigo, uma base inicial. Outro ponto positivo, é que contarão com as abordagens já tomadas para o problema em questão, evitando assim a repetição de trabalho. O artigo será apresentado na 19th EPIA Conference in Artificial Intelligence, entre os dias 3 e 6 de setembro de 2019, e será incluído nas atas dessa mesma conferência, a publicar pela editora Springer, na série Lecture Notes in Computer Science (Duarte, Macedo e Oliveira, 2019).

## 1.4 Estrutura do Documento

Depois desta introdução, este documento encontra-se dividido em seis capítulos, de acordo com os seguintes conteúdos:

- **Capítulo 2** — introduz ao leitor uma série de conhecimentos necessários para melhor entender o trabalho realizado. Aborda temas importantes como o processamento de linguagem natural, o que são modelos de aprendizagem computacional e ainda uma fundamentação teórica sobre emoções.
- **Capítulo 3** — aborda a revisão da literatura bibliográfica relacionada com o trabalho desenvolvido. Incluem-se trabalhos acerca da criação de grandes quantidades de dados categorizados, léxicos existentes sobre o domínio afetivo, os modelos desenvolvidos para a classificação de estados emocionais e previsão de ocorrência de *emojis*.
- **Capítulo 4** — descreve todo o trabalho realizado e as decisões tomadas, incluindo a construção dos processos de extração e de processamento de dados, efetuados para a criação do conjunto de dados. Também é descrita a criação e arquitetura dos modelos de aprendizagem computacional e o processo de otimização dos vários parâmetros que são necessários de definir.
- **Capítulo 5** — apresenta todos os resultados obtidos relativos aos modelos desenvolvidos nas tarefas de classificação de emoção e previsão de *emojis*. Também é efetuado uma discussão dos resultados obtidos das várias experiências efetuadas com os modelos desenvolvidos.
- **Capítulo 6** — resume as principais decisões tomadas e os seus resultados, para além de abordar as principais conclusões retiradas do trabalho realizado e discutir possíveis direções para trabalho futuro.

Esta página foi intencionalmente deixada em branco.

## Capítulo 2

# Fundamentação Teórica

Neste capítulo são abordados, de forma resumida, alguns temas necessários para que o restante trabalho descrito neste documento seja entendido em toda a sua plenitude. Estes temas dizem respeito a alguns pontos sobre Inteligência Artificial como técnicas de Processamento da Linguagem Natural, os diversos modelos de aprendizagem supervisionada e as métricas existentes de avaliação. Também aborda temas da área da psicologia nomeadamente os modelos de emoção mais comuns.

### 2.1 Modelos de emoções

A definição de uma emoção ou de um estado emocional, é um assunto não-consensual. Na área da psicologia, nunca se chegou a um consenso sobre a própria definição, sendo este problema abordado em Kleinginna e Kleinginna (1981). Reuniram-se 92 conceitos de emoção e nunca se chegou a nenhuma definição geral. Se a definição do termo “emoção” é complicada, a categorização dos seus diferentes estados levantam ainda mais problemas. Estas indefinições não ocorrem só no seio da comunidade científica, mas também na sua utilização diária pelo ser humano. Um caso concreto deste ponto são expressões comuns do dia a dia como, por exemplo “Não sei o que sinto”, realçando essa dificuldade de categorização.

De acordo com Russell (1991), a categorização que fazemos das nossas emoções são muito influenciadas por dois fatores que se encontram interligados, a língua nativa e a cultura do indivíduo. Como analisado e resumido por Wu (1982), as sete emoções básicas chinesas são categorizadas de maneira diferente, devido a serem influenciadas pela conceção tradicional e cultural chinesa. O mesmo acontece em nós portugueses, retratados como um povo “melancólico” e “sonhador”, categorizamos estados emotivos únicos, como o caso da “saudade”. O trabalho de Silva (2012) sugere que este estado não tem tradução para outros idiomas, nem uma categorização direta para outras culturas.

Devido à existência destas diferenças, é impossível o desenvolvimento de um modelo, capaz de abranger todos os estados emotivos, capturando ao mesmo tempo estas diferenças culturais. Mesmo com a diversificação das emoções, existem estados emotivos constantes, independentes da cultura ou percurso individual da pessoa (Ekman e Friesen, 1971). Logo, mesmo existindo um elevado número de divergências, é possível a criação de modelos comuns, que possam ser aplicados a qualquer ser humano, sem a necessidade de uma análise ao seu contexto sócio-cultural.



Figura 2.1: Seis emoções básicas de Ekman<sup>1</sup>

Devido a esta possibilidade, foram propostos vários modelos, cujo conteúdo pode ser aplicado de maneira generalizada. Destas novas propostas, surgiu o aparecimento dos modelos categóricos. Estes modelos consistem na categorização do estado emocional em níveis ou etiquetas. Dos primeiros modelos categóricos a surgir e a ter mais impacto, foi o modelo proposto por Ekman (1999), que descreve a categorização do estado emocional em seis emoções básicas do ser humano. Estas ficaram definidas em **felicidade**, **tristeza**, **surpresa**, **medo**, **nojo** e **irritação**. Ekman desenvolveu este modelo com base na leitura de expressões faciais de pessoas (Ekman, 1993). Este conjunto de indivíduos tinham diferentes percursos sócio-culturais, de maneira a gerar um modelo capaz de generalizar as emoções contidas nele.

Alguns trabalhos sobre a classificação de emoções com base em texto recorrem a este modelo para a sua categorização. Com recurso a um grande conjunto de dados provenientes de pequenos contos infantis, Alm, Roth e Sproat (2005) desenvolveram modelos baseados em redes neuronais para a classificação e reconhecimento de emoções em texto, seguindo as categorias propostas por Ekman.

Outro modelo categórico muito usado para a classificação de emoções é o modelo proposto por Plutchik (2001). Este modelo inclui as emoções presentes no modelo de Ekman (1999) e acrescenta as emoções de **antecipação** e de **confiança**. Cada uma destas emoções constitui um conjunto de opostos, criando assim os pares **felicidade-tristeza**, **medo-irritação**, **confiança-nojo**, **antecipação-surpresa**. A representação do modelo encontra-se na figura 2.2.

<sup>1</sup>Adaptado de <https://www.ibmastery.com/blog/emotion%2Dnotes%2Dtheory%2Dof%2Dknowledge>

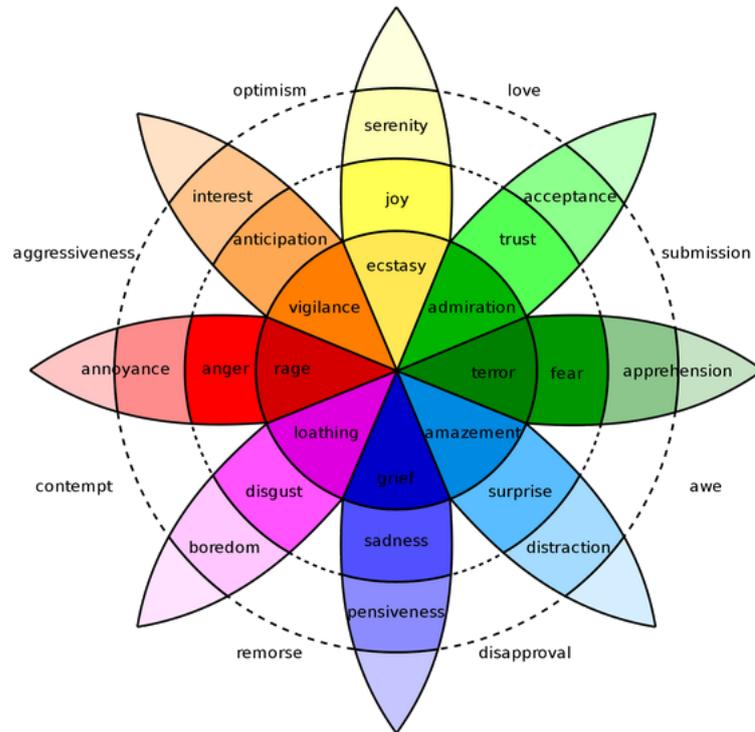


Figura 2.2: Roda de emoções de Plutchik<sup>2</sup>

Suttles e Ide (2013) decidiram não seguir o modelo apresentado por Ekman. Utilizando o modelo de emoções proposto por Plutchik, foi desenvolvido uma série de modelos para uma classificação binária de emoções, tendo como base *tweets*. Esta classificação é efetuada através da comparação dos pares de emoções opostas existentes no modelo.

A possibilidade que os modelos categóricos oferecem é a facilidade de se poderem adaptar a um objetivo pretendido. Esta adaptabilidade deve-se à capacidade de desenvolver um modelo específico para um grupo com características culturais diferentes. Um exemplo deste caso, é o modelo Navarasa (Sreeja e Mahalakshmi, 2016, 2017; Sreeja e Mahalakshmi, 2015), sendo baseado num texto indiano (Gomes, 2005) sobre os aspetos da arte, que é um reflexo da cultura indiana e uma representação emotiva do povo Hindu.

Para além dos modelos categóricos, surgiram também trabalhos baseados na localização espacial do estado emocional numa determinada dimensão. Aqui o modelo mais popular tem por base as investigações de Russell (1979, 1991). O seu modelo de classificação do estado emocional consegue fazer uma categorização generalizada independente da cultura da pessoa (Russell, Lewicka e Niit, 1989). O modelo de Russel descreve que o estado emocional da pessoa pode ser categorizado através de dois valores: a valência e a excitação<sup>3</sup> de um indivíduo naquele instante. A categorização numa etiqueta, como nos modelos categóricos, pode ser efetuada atribuindo a emoção definida nestes modelos, mais próxima. Este método de categorização ficou designado por modelo VA, sendo mais habitual a definição como modelo bidimensional que é possível observar na figura 2.3.

<sup>2</sup>Retirado de [https://www.researchgate.net/figure/Wheel%2Dof%2DEmotion%2Dby%2DRobert%2DPlutchik%2D1958%2DImage%2DSource%2DWikimedia%2DCommons\\_fig2\\_315612031](https://www.researchgate.net/figure/Wheel%2Dof%2DEmotion%2Dby%2DRobert%2DPlutchik%2D1958%2DImage%2DSource%2DWikimedia%2DCommons_fig2_315612031)

<sup>3</sup>Provenientes dos termos em inglês *Valence* e *Arousal*

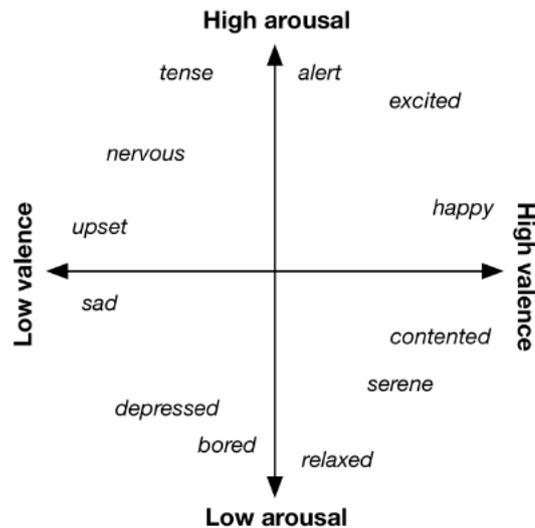


Figura 2.3: Modelo bidimensional com a localização categórica de emoções<sup>4</sup>

Russel propôs ainda um modelo adicional, mas categorizado por três indicadores, adicionando aos valores de *valence* e *arousal* uma dimensão extra, correspondente à dominância<sup>5</sup> da palavra. Assim é construído o modelo VAD ou tridimensional, observável na figura 2.4.

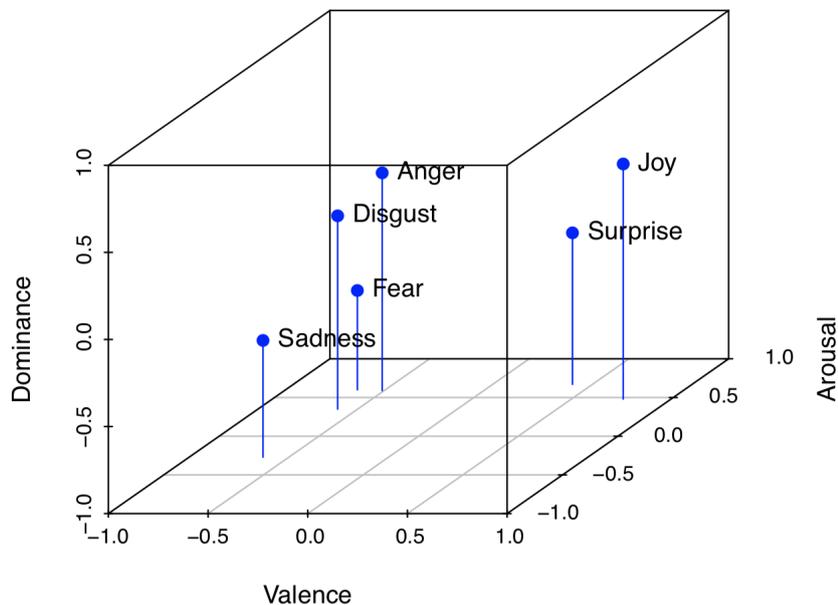


Figura 2.4: Modelo tridimensional com a localização espacial das seis emoções básicas de Ekman<sup>6</sup>

Ambos os modelos apresentados por Russel, são usados em diversos trabalhos de

<sup>4</sup>Retirado de [https://www.researchgate.net/figure/A-PAD-model-of-valence-and-arousal-and-2Dexamples%2Dof-related-discrete-affects\\_fig1\\_280105023](https://www.researchgate.net/figure/A-PAD-model-of-valence-and-arousal-and-2Dexamples%2Dof-related-discrete-affects_fig1_280105023)

<sup>5</sup>Derivado do termo Inglês *Dominance*

<sup>6</sup>Retirado de [https://www.researchgate.net/figure/The-emotional-space-spanned-by-the-2DValence%2DArousal%2DDominance-model-For-illustration-the\\_fig1\\_316546282](https://www.researchgate.net/figure/The-emotional-space-spanned-by-the-2DValence%2DArousal%2DDominance-model-For-illustration-the_fig1_316546282)

classificação de emoções. Para a aplicação destes modelos, foi efetuada a adaptação de léxicos. Os léxicos computacionais são conjuntos de palavras a que pode estar associada informação adicional sobre o seu significado (e.g., categoria gramatical, divisão silábica, sentimento), sendo que o seu conteúdo varia de léxico para léxico. Nos chamados léxicos de emoções estão associadas a cada palavra os valores de *valence*, *arousal* e *dominance*, o que torna possível a aplicação de métodos de classificação com base nestes valores.

Recentemente, também surgiram métodos para efetuar o mapeamento de um modelo categórico para dimensional. O trabalho retratado de Buechel e Hahn (2016, 2018b) descreve que, através de regressões, é possível verificar quais as características que estão relacionadas entre os dois tipos de modelo. Desta forma, é possível efetuar a transformação de valores de um modelo VAD numa emoção presente num modelo categórico e vice versa. Com este processo, é possível evitar erros provenientes do mapeamento através da distância.

Tal como o que acontece nos modelos categóricos, em que é possível definir um modelo com base nas categorias desejadas, o modelo proposto por Thayer (1990) é semelhante, mas para os casos dimensionais. Em vez de usar os valores de *valence* e *arousal*, este modelo utiliza valores de stress e de energia para definir a localização espacial do estado emocional. O mapeamento para um estado emocional categórico é efetuado como nos modelos anteriores, verificando a emoção categorizada mais próxima de si.

## 2.2 Aprendizagem Computacional

A Aprendizagem Computacional é uma área científica que estuda formas de desenvolver sistemas computacionais capazes de realizar tarefas sem que lhes sejam dadas instruções explícitas. Normalmente ligada à Inteligência Artificial, os sistemas “aprendem” a realizar tarefas com base nas características de dados fornecidos. A abordagem mais comum é fornecer a estes sistemas dados já categorizados, de maneira que estes consigam encontrar uma forma de generalizar a solução do problema. O processo de fornecer a estes sistemas conjuntos de dados para que encontrem estes padrões é denominado como treino. Para além disso estes sistemas são muitas vezes designados como modelos de aprendizagem computacional. Caso não seja possível fornecer estes dados de forma categorizada, existem abordagens que são capazes de resolver o problema, criando grupos (*clusters*), dos dados mais semelhantes e assim encontrar categorias e parecenças nos dados. Existem inúmeras abordagens e diversas arquiteturas para efetuar a classificação de dados no qual existem algumas que se destacam mais pelos resultados que obtêm. Uma tarefa muito aplicada a estes modelos de aprendizagem é a classificação de textos em categorias. Devido à popularidade desta tarefa e tendo em conta a sua importância, não tardou em aparecer modelos com arquiteturas mais adequadas para efetuar a classificação de textos.

Uma destas arquiteturas são as redes Bayesianas. Estas redes são modelos gráficos probabilísticos, onde cada variável introduzida representa conhecimento de um domínio incerto. O resultado final da rede, é um gráfico onde cada entrada corresponde a uma variável. Para essa variável as entradas que lhe antecedem são ações condicionantes e as de saídas as variáveis que esta condiciona. Desta forma é criado um conhecimento probabilístico do problema e do domínio em questão, baseando-se para tal na fórmula de probabilidade condicionada de Bayes, que destaca a probabilidade do evento  $x$  ocorrer sabendo que  $y$  é verdade.

$$P(x|y) = \frac{P(y|x)P(x)}{P(y)}$$

Figura 2.5: Formula da probabilidade condicionada

Apesar de o modelo ter como base a fórmula de Bayes e ser relativamente simples conseguimos obter resultados significativos para tarefas de classificação. Para além da classificação de textos, as redes bayesianas são muito usadas para tarefas como processamento de linguagem natural, reconhecimento de voz, diagnósticos médicos, previsão de tempo entre outras. As redes bayesianas de uma forma muito simplificada, permitem classificar um novo texto através da representação desse para uma forma numérica e da probabilidade de ocorrência que esta representação tem para uma determinada categoria (classe).

Para além das redes Bayesianas existem outras arquiteturas para o qual as tarefas de classificação de textos foram aplicadas como é o caso das *Support Vector Machine* (SVM). As SVM são modelos pertencentes à categoria de aprendizagem supervisionada, onde recebem um conjunto de dados e a categoria a que estes pertencem. As SVM usam estes dados, para efetuar uma regressão de previsão ou classificá-los de acordo com as categorias existentes. O que estes modelos fazem, é uma representação dos dados em pontos num espaço dimensional, e tentam encontrar uma margem de divisão, (o termo usado é o de fronteira). Para além de encontrar esta margem tenta-se sempre que as categorias existentes sejam separadas com a maior distância possível. Novos dados que sejam introduzidos, são classificados de acordo com a localização espacial a que estes se encontram, da fronteira calculada. As SVM são consideradas por *Black Box Systems*, devido a não se ter um conhecimento aprofundado sobre o processamento dos dados, dentro do modelo. Este desconhecimento constitui um problema, pois precisamos de dar explicações e ter fundamentos para a solução encontrada (Gunning, 2017).

Estes modelos também podem ser desenvolvidos com base em aprendizagem não supervisionada, onde se tenta agrupar dados com características semelhantes. Deste processo tenta-se encontrar uma fronteira, de maneira que separe os grupos criados uns dos outros. A classificação de novos dados é feito da mesma forma, com base na proximidade que estes têm dos grupos existentes. Esta proximidade pode ser calculada de diversas formas, sendo a mais comum a utilização da distância Euclidiana. As SVM são também muito aplicadas em tarefas como bioinformática, deteção de faces, entre outras.

Com o aumento da capacidade de processamento bem como a evolução da aprendizagem computacional, a aplicação de redes neuronais para tarefas de classificação começou a tomar alguma popularidade. As redes neuronais são representações artificiais de um cérebro biológico e do seu funcionamento. Tal como as SVM as redes neuronais são modelos de aprendizagem supervisionada, que aceitam um conjunto de dados juntamente com as categorias que lhes pertencem. Destes dados, tenta-se encontrar uma maneira de generalizar a solução, fazendo a divisão entre os dados introduzidos. As redes neuronais tal como as SVM são *Black Box Systems* e muito dificilmente conseguimos obter os passos que levaram a rede a tomar certas decisões no que diz respeito à classificação de novos dados.

As redes neuronais permitem modificar a sua arquitetura, sendo que cada uma deve ser adaptada para o problema em questão, pois, é sempre necessário afinar certos parâmetros. As redes neuronais são constituídas por múltiplas unidades individuais interligadas entre

si, denominadas por neurónios, que como o nome indica funcionam de forma semelhante aos existentes no cérebro humano. Também são constituídas por pesos, de maneira a dar mais importância a certas ligações entre neurónios do que outras. Por fim, são utilizadas funções específicas entre camadas para ser possível efetuar uma classificação dos dados introduzidos. O processo de aprendizagem nas redes neuronais é feito através de retro-propagação, onde é feita uma análise da diferença entre o resultado obtido e o resultado esperado. Desta diferença são atualizados os pesos existentes, de maneira a darem o valor pretendido e assim obter-se uma aproximação ao resultado desejado. A figura 2.6 representa um neurónio de uma rede neuronal.

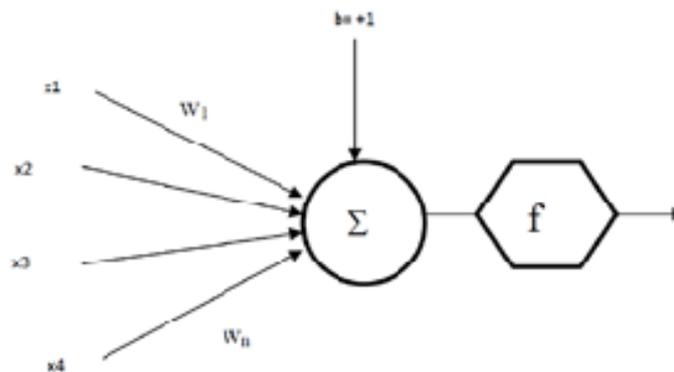


Figura 2.6: Representação de um neurónio<sup>7</sup>

O processamento dos dados ao passar no neurónio da figura 2.6 é feito através da multiplicação dos valores de entrada ( $x_i$ ) e dos seus pesos ( $w_i$ ). Os resultados obtidos através destas multiplicações são então somados, juntamente com um peso adicional ( $b$ ) para depois ser aplicado à função  $f$ . Normalmente estas funções são do tipo lineares, logarítmicas ou sigmóides. Uma rede composta por diversos neurónios é denominada por rede neuronal simples<sup>8</sup>. As redes neuronais são cada vez mais utilizadas em tarefas como reconhecimento de imagens, dígitos e previsões de inúmeros temas, mas podem ser adaptadas a vários problemas.

Das diversas arquiteturas das redes neuronais existem algumas que se destacam mais, devido a serem utilizadas com mais frequência. Exemplos destas são as redes neuronais *Convolutional Neural Network*, focadas para o reconhecimento de objetos em imagens. Outra arquitetura importante, as redes neuronais *Long Short-term Memory* são evoluções das *Convolutional Neural Network*. Estas redes tentam simular de forma detalhada o funcionamento de um cérebro biológico, pois, armazenam informação entre camadas. Desta forma é possível simular o conceito de memória a curto e longo prazo dos seres humanos, e obter resultados que anteriormente não eram possíveis. Exemplo de uma célula LSTM é visível na figura 2.7

<sup>7</sup>Retirado de [https://www.researchgate.net/figure/Basic-Neural-Network-Cell\\_fig6\\_318680143](https://www.researchgate.net/figure/Basic-Neural-Network-Cell_fig6_318680143)

<sup>8</sup>O mais comum é o termo em inglês *shallow neural network*

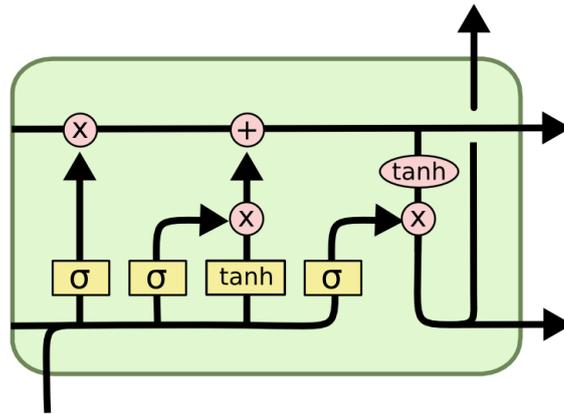


Figura 2.7: Representação de uma célula LSTM<sup>9</sup>

A junção de várias células LSTM criam uma rede LSTM. Para além disso estas células LSTM podem ser combinadas com outras mais simples de forma a dar mais flexibilidade e assim conseguir obter uma solução. Pela figura verifica-se que o processo de transformação de dados dentro de cada célula é bastante complexa. De uma forma muito simplificada, os dados sofrem transformações com base na informação retida de outras células e passam informação importante para as células seguintes. Assim cria-se um conceito de memória, onde a informação é passada ou esquecida conforme o processamento efetuado. Na figura 2.7 a parte superior da célula diz respeito ao tratamento dos dados que são passados à célula, e a parte inferior respetiva ao conceito de memória. A informação é passada de células anteriores para células posteriores, onde esta é aplicada para fazer alterações nos dados que passam na parte superior.

Um dos principais problemas do desenvolvimento de modelos de aprendizagem computacionais é que existe uma elevada quantidade de parâmetros a definir. Para além de ser preciso definir os valores para estes parâmetros durante o desenvolvimento do modelo, estes terão grande influência no desempenho e funcionamento nos resultados finais. Para tal é necessário verificar qual a melhor combinação de valores para os diversos parâmetros do modelo a desenvolver. Para obter esta combinação surgiu uma subárea da aprendizagem computacional denominada como hiper-parametrização<sup>10</sup>. O *grid search* é um de diversos métodos existentes para efetuar a hiper-parametrização de modelos de aprendizagem computacional. O *grid search* é também o método mais simples de executar a hiper-parametrização, pois consiste numa simples pesquisa exaustiva de um subconjunto especificado manualmente. O processo vai consistir no desenvolvimento de vários modelos, onde cada um conta com uma das várias combinações possíveis para os parâmetros a definir sendo de seguida treinado e avaliado. Este método deve ser guiado por alguma métrica de desempenho de forma a poder ser avaliado e devolver a melhor combinação de valores para os parâmetros a definir. Apesar de ser um processo que gasta muitos recursos físicos, pois efetua todas as combinações possíveis, ganha na sua simplicidade levando assim ao melhor modelo para a tarefa estipulada.

No entanto para a avaliação de um modelo é preciso definir um conjunto de métricas para que seja possível determinar qual a melhor opção para a tarefa em questão. Para efetuar a avaliação é preciso efetuar o processo de treino do modelo em questão e de

<sup>9</sup>Retirado de <https://colah.github.io/posts/2015-08-Understanding-LSTMs>

<sup>10</sup>Traduzido do termo em inglês *Hyperparameter*

seguida, fornecer novos dados no qual sabemos as classes a que estes pertencem. De seguida é efetuada a classificação destes novos dados e com os resultados obtidos podemos comparar os valores obtidos com aqueles que eram esperados. Normalmente é usada uma percentagem do conjunto de dados inicial para efetuar o treino do modelo sendo o resto aplicado para a avaliação do modelo. Esta avaliação pode ser efetuada de diversas formas e indicar problemas que por vezes não são detetados numa simples avaliação correto/incorrecto. De forma a calcular estas métricas existem quatro tipos de valores que podem ocorrer quando avaliamos um modelo. Estes valores são dados pela tabela 2.1.

		Classe Prevista	
		Classe = Sim	Classe = Não
Classe Verdadeira	Classe = Sim	Verdadeiro Positivo	Falso Postivo
	Classe = Não	Falso Negativo	Verdadeiro Negativo

Tabela 2.1: Resultados possíveis durante a avaliação de modelos computacionais

A tabela 2.1 representa as combinações que podem ocorrer na avaliação de um modelo de classificação binária. Um verdadeiro positivo é quando a classe é classificada como pertencente à classe 1, quando no conjunto de dados é atribuída essa mesma classe. A ocorrência de um verdadeiro negativo é o mesmo mas para a classe 0. Os falsos positivos e falsos negativos é quando ocorre uma má classificação da classe. Apesar da representação estar desenhada para uma classificação binária, no caso de existir uma classificação com múltiplas classes o conceito será o mesmo. Com estes valores conseguimos calcular métricas que nos permite avaliar corretamente o modelo desenvolvido. Essas métricas são apresentadas de seguida.

### Accuracy

A *accuracy*<sup>11</sup> é a medida de desempenho mais intuitiva e é a proporção das observações corretamente previstas com o total de observações. Esta métrica leva a conclusões que quanto mais alta a *accuracy*, melhor é o nosso modelo. Isto é o caso quando temos conjuntos de dados simétricos em que os valores de falso positivo e falso negativo são quase iguais, o que muito raramente representam a realidade. O cálculo desta métrica é feito através da seguinte formula:

$$accuracy = \frac{VP + VN}{VP + VN + FP + FN} \quad (2.1)$$

Apesar do problema do seu valor ser enganador é normalmente apresentado em casos de avaliação de modelos.

### Precision

A *precision*<sup>12</sup> é a proporção de observações positivas previstas corretamente com o total de observações positivas previstas. A *precision* é calculada da seguinte forma:

$$precision = \frac{VP}{VP + FP} \quad (2.2)$$

<sup>11</sup>Em português “exatidão”

<sup>12</sup>Em português “precisão”

É uma métrica mais adequada do que a *accuracy* pois em caso de predominância de uma classe os valores de *precision* das restantes será bastante baixo. A ocorrência destes casos indica que o modelo classifica muito bem um caso em particular mas a classificação das restantes classes não são muito boas.

### Recall

O *recall*<sup>13</sup> é a proporção de observações positivas previstas corretamente para todas as observações da classe real. Também é descrita por vezes como *sensitivity*. É obtida através da seguinte formula:

$$recall = \frac{VP}{VP + FN} \quad (2.3)$$

### Specificity

A *specificity*<sup>14</sup> é considerada como a medida dos falsos verdadeiros. Consiste nas observações negativas que são classificadas corretamente. É calculada a partir da seguinte formula:

$$specificity = \frac{VN}{VN + FN} \quad (2.4)$$

### F1-Score

O *F1 Score* é a média harmónica da *Precision* e do *Recall*. Logo este valor tem em conta tanto os falsos positivos como os falsos negativos. É um valor que não é tão fácil de entender quanto a *accuracy*, mas o *F1* é geralmente mais útil que a *accuracy*, especialmente se tivermos num caso em que a distribuição de classe não é balanceada. O valor de *F1* é dado pela seguinte maneira.

$$F1 = 2 \times \frac{precision \times recall}{precision + recall} \quad (2.5)$$

Normalmente é a métrica mais usada para comparar o desempenho de modelos de aprendizagem automática.

A separação do conjunto de dados, onde uma parte é usada para o treino do modelo e outra para teste, é uma prática bastante comum mas no entanto existem diversas formas de efetuar a divisão dos dados. O método de *k-fold cross validation* é um procedimento de amostragem usado para avaliar modelos de aprendizagem com uma amostra de dados limitada. O procedimento tem um único parâmetro, *k* que se refere ao número de grupos para os quais uma determinada amostra de dados deve ser dividida. Normalmente após a escolha deste valor *k* atribui-se ao nome do método esse mesmo valor, como *k = 10*, denominamos o processo *10-fold cross validation*.

---

<sup>13</sup>Em português “recordação”

<sup>14</sup>Em português “especificidade ”

Este método é usado principalmente no treino de modelos computacionais para estimar a habilidade de aprendizagem deste em dados não vistos. Ou seja, usar uma amostra limitada para estimar o desempenho esperado do modelo em geral quando usado para fazer previsões ou classificações sobre dados não utilizados durante o treinamento do modelo. É um método bastante comum porque resulta numa estimativa menos tendenciosa ou menos otimista da habilidade do modelo do que os métodos de divisão de treino e teste. Os valores de  $k$  são escolhidos de acordo com o problema sendo o valor de 10 muito comum de se utilizar. A figura 2.8 demonstra este método de forma mais simplificada e intuitiva.

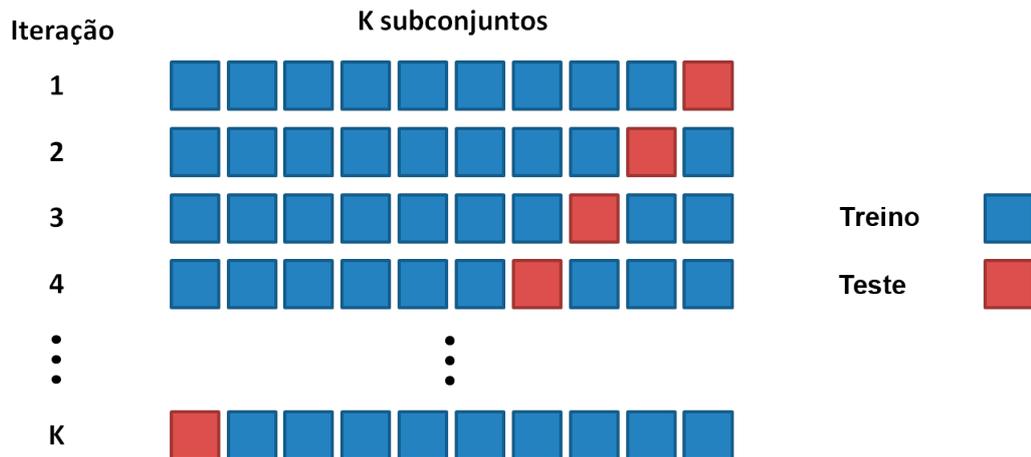


Figura 2.8: Exemplo do método de k-cross validation<sup>15</sup>

Os valores obtidos deste processo são as médias das métricas estipuladas, obtidas após cada iteração.

Devido à quantidade elevada de valores existentes nos conjuntos de dados aplicados aos modelos de aprendizagem, é necessário encontrar formas de reduzir o seu tamanho. De forma a não eliminar valores importantes é preciso avaliar as características presentes nestes dados de forma a eliminar aquelas que não acrescentam grande informação. Para este objetivo, existem diversos testes que podem ser aplicados às várias características presentes nos dados. O qui-quadrado ( $\chi^2$ ) é um teste estatístico que permite determinar a dependência de duas variáveis em estudo. O teste de qui-quadrado só é aplicável a dados categóricos ou nominais, logo é muitas vezes aplicado em trabalhos relacionados com aprendizagem computacional. Nestes casos o teste de qui-quadrado é utilizado para procurar as características (*features*) menos dependentes com as classes especificadas para depois remove-las do conjunto de dados. A este processo é dado o nome de redução de características<sup>16</sup> reduzindo a quantidade de dados para processar sem que este leve a perdas de informação.

O qui-quadrado é calculado através da seguinte forma:

$$\chi^2 = \sum_{j=1}^n \frac{(O_j - E_j)^2}{E_j} \quad (2.6)$$

Onde  $O_j$  corresponde à frequência observada da classe  $j$  e  $E_j$  a frequência esperada. No caso da classificação de emoções temos uma variável para cada uma das classes (uma

<sup>15</sup>Adaptado de [https://ericcouth.files.wordpress.com/2013/07/cross\\_validation.png](https://ericcouth.files.wordpress.com/2013/07/cross_validation.png)

<sup>16</sup>Traduzido do termo em inglês *feature reduction*

das seis emoções básicas) e variáveis que descrevam cada uma destas classes. Neste caso é calculado as estatísticas de qui-quadrado entre cada variável das características e a variável de classe e observamos a existência de uma relação entre si. Se as variáveis forem independentes podemos descartar essa característica pois não são relacionadas, caso contrario, demonstra uma dependência onde quanto maior for o valor obtido maior é relação entre as duas variáveis.

## 2.3 Processamento de Linguagem Natural

A área denominada por Processamento de Linguagem Natural (Martin e Jurafsky, 2009), é uma sub-área da Inteligência Artificial que tem como objetivo final desenvolver sistemas que compreendam a língua humana e consigam comunicar através dela. Para além da extração de informação é crucial que um sistema NLP consiga lidar com os vários fenómenos da língua humana, onde se destaca a ambiguidade (i.e., a mesma palavra ou expressão podem ter dois sentidos diferentes) ou a variabilidade linguística (i.e. o mesmo conceito ou intenção podem ser transmitidos através de palavras ou expressões diferentes). Mesmo entre humanos, muitos desses fenómenos são difíceis de lidar, o que aumenta a complexidade do NLP. Um desses exemplos é a deteção de ironia, dando como exemplo a crítica “A sopa estava uma delícia: fria e sem tempero”, alguns sistemas de NLP classificarão este caso como positivo, quando na realidade ele é bastante negativo.

Exemplos de uma aplicação relativamente completa de NLP são os *chatbots*, programas que respondem a perguntas do utilizador, a quem também podem fazer perguntas, simulando uma conversa que se assemelha à de dois humanos. Assistentes pessoais, como o caso da Siri<sup>17</sup>, Alexa<sup>18</sup> ou Cortana<sup>19</sup> presentes em diversos dispositivos, são capazes de reconhecer comandos por voz, interpretar o que é pretendido e, se possível, executar a ação desejada.

Para resolver estes problemas, o NLP recorre a várias técnicas e processos para desenvolver sistemas capazes de extrair informação num texto, e que aproximem a interação entre a máquina e o ser humano. Alguns destes processos estão na base do NLP, ou seja, é difícil de lhes fugir. Por exemplo, a *tokenização* consiste na transformação de frases em listas de átomos (*tokens*), simplificando assim as análises seguintes. Partindo da frase “O rato roeu a rolha”, o processo de *tokenização* resulta numa lista com os termos “O”, “rato”, “roeu”, “a”, “rolha”. Apesar de ser um processo intuitivo, a *tokenização* é bastante importante no âmbito do processamento de linguagem natural. Para além da separação através de palavras pode ser necessário efetuar o tratamento de casos com pontuações e contrações. Nos casos em que abreviaturas são ocorrentes a *tokenização* terá de efetuar o tratamento correto, de forma a que a palavra não seja separada. A *tokenização* também efetua o tratamento de contrações separando termos como “na” em “em” e “a” caso seja necessário.

O *stemming* é um processo de normalização do texto que consiste em retirar a terminação de uma palavra, facilitando assim a associação de palavras com a mesma raiz (singular e plural, diferentes conjugações do mesmo verbo). É um processo relativamente simples que não precisa de muita informação, e é normalmente aplicado em tarefas de classificação ou recuperação de informação. Mas o resultado do *stemming* nem sempre é uma palavra, e daí podem surgir problemas. Por exemplo, é um processo que nem

---

<sup>17</sup><https://www.apple.com/siri/>

<sup>18</sup><https://developer.amazon.com/alexa>

<sup>19</sup><https://www.microsoft.com/pt-br/windows/cortana>

sempre resulta numa palavra válida, podendo dificultar a interpretação do texto. Outro problema é a dificuldade em tratar palavras irregulares no qual não se irá obter nenhuma raiz de palavra e assim resultar num processo ineficiente para o ganho de informação. Um exemplo que engloba estes dois casos é a frase “O gato e as gatas” que resulta em “O gat e as gat” onde não se obtém nenhum ganho. Uma maneira de combater estes problemas é aplicar a lematização, que consiste em deflexionar uma palavra para determinar o seu lema.

Outro processo bastante comum é o *Part of Speech Tagging* (PoS) que consiste em atribuir a cada palavra uma etiqueta correspondente à sua função gramatical na frase. Identificar o PoS é muito mais complicado do que simplesmente mapear as palavras. Em diferentes frases, a mesma palavra pode ter um PoS diferente quando aplicado a frases ou textos distintos. É por isso que é impossível ter um mapeamento generalizado para o PoS e daí o surgimento de diversos algoritmos para efetuar este método. Estes algoritmos também efetuam o PoS usando diversas técnicas, sendo mais comum métodos probabilísticos ou baseados em regras. No caso dos métodos baseados em regras alguns destes consistem em verificar a terminação da palavra e atribuir se é um verbo, preposição ou nome. Os métodos probabilísticos, tem como base a probabilidade de uma determinada sequência de palavras de ocorrer. Devido à existência de vários algoritmos com técnicas diferentes de efetuar o PoS, é comum preposições serem consideradas como determinantes, ou verbos serem classificados como adjetivos. Na figura 2.9 estão dispostos exemplos dos processos de NLP abordados.

## O rato roeu a rolha da garrafa do rei da Russia

Tokenização: [O, rato, roeu, a, rolha, da, garrafa, do, rei, da, Russia]  
 Stemming: [o rat roeu a rolh da garraf do rei da russ]  
 PoS: [(O, Det), (rato, Nome), (roeu, Verbo), (a, Det), (rolha, Nome),  
 (da, Prep), (garrafa, Nome), (do, Prep) (rei, Nome), (da, Prep),  
 (Russia, Nome)]

Figura 2.9: Exemplos de processo de NLP

Temos ainda processos de análise semântica, que procuram identificar conceitos relevantes numa frase (por exemplo, entidades mencionadas) e relações entre eles (por exemplo, é-um, localizado-em), recorrendo à análise de padrões e também a léxicos dedicados, que fornecem informação acerca de palavras ou conceitos. Existe uma grande variedade de léxicos, cada um para um objetivo em concreto. Como será abordado na secção 3.2 alguns destes léxicos foram desenvolvidos para fornecer informação afetiva sobre a palavra em questão. Estes processos são apenas alguns que a NLP utiliza para extrair grande parte do significado da frase ou de um texto. Desta forma, tenta-se simular uma interação mais natural, entre a máquina e o seu utilizador.

Como descrito na secção 2.2, os modelos de aprendizagem computacional são mais adequados para conseguirem processar dados na forma numérica. Nos casos em que a tarefa consiste em utilizar dados de forma textual, é normal efetuar uma transformação do texto para uma representação numérica. Esta transformação será através duma representação *space vector model* dos nossos dados. Os *space vector models* consiste num processo onde os documentos são representados na forma de vetores, onde cada termo

corresponde a uma dimensão. A abordagem mais simples de construir estes vetores é utilizando um método de *bag-of-words* que consiste em criar um vocabulário de todos os termos únicos presente no conjunto de dados, sem ordem específica. Também se pode construir este vocabulário com n-gramas, que consiste na sequência de  $n$  palavras seguidas. Ao utilizar n-gramas estaremos a obter informação sobre conjuntos de palavras (e.g. “San Francisco”, “Os três porquinhos”) que antes seriam consideradas como termos individuais. Usando unigramas ( $n = 1$ ), cada posição da lista obtida diz respeito ao termo  $i$  do vocabulário criado. Após efetuar esta representação é normalmente calculada a ocorrência dos termos presentes na lista, para cada texto/documento no conjunto de dados. O resultado final de todo este processo é uma representação em matriz denominada *document-term* do conjunto de dados. Nesta matriz cada linha corresponde a um texto/documento específico do conjunto de dados e cada coluna a ocorrência do termo  $i$  nesse mesmo texto/documento. Ao analisar a matriz cada linha representa um vetor de dimensão correspondente ao número de colunas. De forma a ilustrar o processo a figura 2.10 demonstra o método *bag-of-words* e a matriz *term-frequency* resultante.

Texto 1 = Ele foi o melhor aluno

Texto 2 = Ele foi o melhor aluno atleta

Texto 3 = Ele foi o pior atirador

Vocabulo = ['Ele'; 'foi'; 'o'; 'melhor'; 'aluno'; 'atleta'; 'pior'; 'atirador']

Bag of words =

	Ele	foi	o	melhor	aluno	atleta	pior	atirador
Texto 1	1	1	1	1	1	0	0	0
Texto 2	1	1	1	1	1	1	0	0
Texto 3	1	1	1	0	0	0	1	1

Figura 2.10: Exemplos do método de *bag-of-words*

A contabilização da ocorrência de cada termo é definido como *term frequency*. Este método era usado inicialmente na pesquisa de documentos onde procurava dentro de uma coleção de documentos, aqueles que continham ocorrências dos termos que desejamos obter. No caso de querermos procurar documentos sobre a economia de Portugal, a abordagem inicial era procurar todos os documentos que contenham pelo menos um dos termos “economia” e “Portugal”. A ideia principal era que documentos semelhantes terão termos semelhantes, algo que se mostrou que nem sempre representava a realidade. O problema com o *term frequency* é que por vezes termos com elevado número de ocorrências não transmitem muita informação. Exemplo destes termos são as denominadas palavras vazias<sup>20</sup> (e.g. “a”, “o”, “é”) que só acrescentam mais conteúdo para processar, sem obter nenhum tipo de informação da palavra.

De maneira a combater este tipo de ocorrências surgiu o método de *inverse document frequency*. O valor de *inverse document frequency* indica a importância que o termo terá num documento, com base nas suas ocorrências em toda a coleção de documentos. Desta forma, mesmo que um termo tenha uma frequência elevada num documento, se ele for utilizado em muitos documentos, será menos importante para caracterizar o documento em

<sup>20</sup>traduzido do termo inglês *stopwords*

questão, e deverá ter um peso inferior. A junção dos métodos *term frequency* e *inverse document frequency* ficou designada por *Term Frequency Inverse Document Frequency* TF-IDF (Amati e Van Rijsbergen, 2002). O valor de TF-IDF para um determinado termo  $i$  de uma coleção de documentos  $d$  é calculado da seguinte forma:

$$\text{tfidf}_{i,j} = \text{tf}_{i,j} \times \log\left(\frac{N}{df_i}\right) \quad (2.7)$$

Onde  $\text{tf}_{i,j}$  corresponde ao total de ocorrências do termo  $i$  em  $d$ , e  $df_i$  o número total de textos que contêm o termo  $i$ . Por fim  $N$  corresponde ao número total de documentos em  $d$ .

O TF-IDF é muito utilizado nas representações *space vector model* para além do *term frequency*. Neste processamento é calculado o valor de TF-IDF para cada termo  $i$  do vetor resultante do *bag-of-words* em vez da frequência de ocorrência. Para lidar com casos em que o número total de termos chega à casa de milhões, pouco praticáveis, é comum procurar formas para reduzir a dimensão e assim tornar o processo mais eficiente. Isso pode passar pela limitação mínima de ocorrência de termos e/ou valor de IDF. Desta forma estipulamos um valor para que só sejam considerados termos que ocorram mais do que essa constante, ou que tenham um valor IDF superior ao estipulado.

De forma a obter a similaridade entre documentos é preciso definir uma métrica para aplicar aos vetores resultantes dos vários documentos processados. A medida mais usada é calcular o coseno dos ângulos formado pelos dois vetores que desejamos comparar. Neste caso valores de coseno perto de 1 indicam muita similaridade dos dois vetores que estamos a comparar, e valores perto do 0 indica que os documentos não são nada semelhantes.

Outra forma de efetuar a representação de termos numa forma numérica é através do *word2vec* (Mikolov et al., 2013). O *word2vec* resulta da aplicação de uma rede neuronal com duas camadas, que são treinadas para representar e reconstruir os contextos dos termos do conjunto de dados, em *word embeddings*. As *word embeddings* são representações de termos como pontos num espaço dimensional.

Para o *word2vec* é necessário um grande conjunto de textos, onde é produzido um espaço vetorial, de  $n$  dimensões definidas na sua criação. Desta forma o *word2vec* atribui a cada termo único no conjunto de dados um vetor correspondente no espaço. Estes vetores são depois posicionados num espaço vetorial de modo que as palavras que compartilham contextos comuns no conjunto de dados estejam localizadas próximos uns dos outros.

Resultados interessantes que os modelos *word2vec* produzem é o caso de responderem a analogias, onde o modelo permite-nos obter respostas que “Rei - Homem + Mulher” é igual a “Rainha”. O *word2vec* efetua uma série de cálculos através do coseno dos ângulos, pois tal como os documentos baseia-se numa representação vetorial, que de forma resumida vai devolver o vetor mais semelhante ao de rei e mulher e menos semelhante a homem. Os modelos *word2vec* também são muito utilizados em motores de busca, permitindo obter termos semelhantes com base na similaridade do vetor resultante, da *query* efetuada.

Esta página foi intencionalmente deixada em branco.

## Capítulo 3

# Trabalho Relacionado

Nos últimos anos, com o desenvolvimento das áreas da Computação Afetiva (Lisetti, 1998) e da Inteligência Artificial, o estudo das emoções foi explorado em diversos trabalhos, com resultados importantes para a comunidade científica. Este capítulo aborda a literatura existente sobre o tema da classificação de emoções com base em texto. Descreve-se inicialmente as várias maneiras de construir grandes quantidades de dados anotados, relativos ao domínio afetivo. Por fim, descrevem-se os trabalhos que utilizam léxicos existentes para este domínio, juntamente com métodos e arquiteturas relevantes para o contexto da classificação de emoções em texto.

### 3.1 Recolha Automática de Textos e Emoções Associadas

A crescente quantidade de dados produzidos e disponibilizados diariamente, nomeadamente na *Social Media*, abrem uma oportunidade para diversas entidades conduzirem processos de *data mining*. Assim, conseguem satisfazer as necessidades dos seus clientes (Barbier e Liu, 2011) e encontrar potenciais mercados e soluções, que anteriormente desconheciam.

De maneira a efetuar uma aprendizagem supervisionada (secção 2.2), é necessário um grande volume de dados e a categoria a que estes pertencem. Num processo normal, estes são divididos em conjuntos mais pequenos, para efetuar os processos de treino, teste e validação do modelo a desenvolver. No problema de deteção de emoções em texto e de previsão de *emojis*, o problema é o mesmo. É necessário a existência de grandes quantidades de textos com a anotação correta da emoção associada, para serem aplicados ao processo de treino do modelo. Contudo, para além do custo e morosidade da tarefa, devido à ambiguidade que, por vezes, é sentida em relação à emoção presente num texto, é difícil efetuar esta criação e catalogação de dados com anotadores humanos. Neste contexto, surge a opção de tirar partido da presença de *emojis*. Através do relacionamento existente entre a representação gráfica do *emoji* e a emoção consentida na altura da escrita, é possível fazer-se a associação de emoções, e assim simplificar este problema de anotação.

A rede social *Twitter*<sup>1</sup> fornece uma grande quantidade de dados sobre diversos domínios de opinião pública, comunicação, notícias entre outros. Estes dados são utilizados por investigadores, analistas e empresas para estudos de novas campanhas ou métodos de investigação (Driscoll e Walker, 2014). O *Twitter* é um dos *websites* mais utilizados diariamente, muito devido à maneira simplificada do seu uso. Existem cerca de 326 milhões

---

<sup>1</sup><https://twitter.com>

de utilizadores ativos mensalmente, sendo previsto um crescimento para 400 milhões de utilizadores até 2020. Estes utilizadores trocam e partilham milhões de pequenos textos por dia, e seguem os seus utilizadores ou temas favoritos. Estes textos denominam-se por *tweets* e têm como limite máximo 280 caracteres, podendo conter algum tipo de imagem, URL externo e a localização de onde foi partilhado. Os *tweets* também se caracterizam por serem escritos informalmente, e nem sempre seguem o rigor de uma escrita correta. Por exemplo, utilizam muitas abreviaturas, não usam pontuações de forma adequada, nem sempre utilizam maiúsculas e minúsculas de forma apropriada, e é regular conterem erros ortográficos. É também comum a identificação de outro utilizador da rede (com recurso ao carácter '@' seguido do nome de utilizador destinado), a utilização de *hashtags*, isto é palavras tipicamente associadas ao tema do texto, precedido pelo carácter '#' e ainda a utilização de *emojis* (😊), pequenas imagens que transmitem a ideia de um sentimento ou palavra. Os *emojis* são normalmente descritos pela imagem de uma cara, onde o estado é alterado de acordo com as variantes existentes, mas, nos últimos anos, surgiram vários tipos de *emoji* que fogem a esta norma (e.g., mãos, bandeiras, animais e outros conceitos). A rede social conta ainda com métodos de privacidade, permitindo ao utilizador partilhar os seus *tweets* apenas com os seus seguidores ou disponibilizá-los publicamente (*standard*).

Os primeiros passos a dar no domínio afetivo sobre dados extraídos da rede social *Twitter*, foram relativos à extração de sentimento (positiva, negativa ou neutra) dos *tweets* de língua inglesa (Agarwal et al., 2011). Com um grande conjunto de textos extraídos, seguido de um processo de anotação do respetivo sentimento com anotadores humanos, são depois construídos modelos para efetuar a classificação de novos dados. Este método manual mostrou-se demorado e pouco eficaz. Para além disso, resultados obtidos pelos classificadores não foram os esperados, devido ao quanto problemático e ambíguo é o processo de anotação de dados. O interesse na análise de sentimentos em textos nas redes sociais levou mesmo à realização de várias tarefas de avaliação conjunta, onde sistemas competem na atribuição automática de sentimento a *tweets* (ver, por exemplo, (Nakov et al., 2013; Rosenthal, Farra e Nakov, 2017))

O estudo de análise do sentimento não foi realizado apenas para a língua inglesa, nem reservado a um pequeno conjunto de modelos, sendo este trabalho expandido para diversos idiomas. No trabalho de Mozetič, Grčar e Smailović (2016) verifica-se a abordagem comum de extrair grandes quantidades de dados do *Twitter* mas para vários idiomas. Mas desta forma fica a necessidade da criação de métodos que permitam a catalogação destes dados de forma automática. Para português, a investigação na área de análise de sentimentos também foi abordada (Souza e Vieira, 2012). O objetivo do trabalho anterior consistiu numa análise do sentimento presente num *tweet*, através de indicadores como a existência de *hashtags*. Mais concretamente as *hashtags* **#vencer** ou **#falhar**. Devido ao significado das *hashtags*, é possível inferir se o *tweet* em que estas se encontram tenha o sentimento associado, como positivo ou negativo respetivamente.

A opção de recolher dados do *Twitter* tornou-se, então, uma tendência para a criação de grandes quantidades de dados automaticamente anotados. Esta tendência deve-se muito ao carácter emocional associado com a partilha de conteúdos na *social media*. Para além do conteúdo partilhado, a vasta utilização de novas formas de comunicação como *emoticons*, *hashtags* e, mais recentemente, os *emojis*, permitem a criação de dados anotados automaticamente. Para a recolha destes dados, o *Twitter*, conta com uma API<sup>2</sup> no qual permite efetuar a comunicação com a rede social. De forma a proteger a privacidade dos seus utilizadores, é necessário obter uma série de credenciais do *Twitter* de forma a poder usar esta API. Com a automatização deste processo, evita-se assim despesas e

---

<sup>2</sup><https://developer.twitter.com>

complicações que ocorrem, com o recurso a pessoas diretamente e especificamente para a anotação dos dados.

No trabalho Mohammad e Kiritchenko (2015) usou-se um método para a criação de um grande conjunto de dados anotados com recurso aos *hashtags*, onde é assumido que estes representam a categoria correspondente ao *tweet* em questão. Com este método, construiu-se um léxico denominado por *Hashtag Emotion Lexicon*, que tem por base a análise de *hashtags*, correspondentes às emoções básicas (*#Happy*, *#Sad*, *#Surprised*, *#Angry*, *#Disgusting*, *#Fear*) propostas por Ekman (1999).

Em Dosciatti, Ferreira e Paraiso (2015) efetuou-se a construção de um corpus de notícias anotado manualmente para as seis emoções de Ekman (1999). Este corpus é composto por 1500 pequenos textos, em Português do Brasil, sendo estes textos constituídos pelo título e subtítulo de notícias extraídas de jornais online e anotados manualmente. O trabalho relata a construção deste corpus, de onde é possível retirar algumas conclusões. Uma das conclusões obtidas é que textos jornalísticos não têm grande variabilidade de emoções, no que resulta em discordâncias da classe a atribuir entre os anotadores. O resultado deste trabalho está disponível gratuitamente, no qual serão efetuadas experiências usando este corpus para os modelos desenvolvidos.

Hasan, Agu e Rundensteiner (2014) executa um processo semelhante, para detetar casos de depressão em universidades americanas. Para atingir esse objetivo, é criado um grande conjunto de dados categorizados da língua inglesa, para a roda de emoções de Plutchik (2001). No trabalho Hasan, associa o texto à classe respetiva, com recurso ao *hashtag* presente no *tweet*. Os *tweets* extraídos foram depois aplicados ao treino de um modelo, de onde se tentou efetuar a classificação da emoção presente em novos *tweets*.

Apesar de usar a *hashtag* como etiqueta<sup>3</sup> de modo a criar grandes conjuntos de dados, tentou-se sempre procurar por mais métodos de automatizar a categorização de textos. Desta forma tentou-se identificar melhores indicadores, que expressem um valor mais claro e mais forte da emoção que o texto transmite. Desta necessidade juntou-se outra maneira de comunicação dentro da *social media*, os *emojis*. A investigação de Wood e Ruder (2016) propõe um método semelhante aos apresentados por Hasan, Agu e Rundensteiner (2014) e Mohammad e Kiritchenko (2015), mas baseado nos *emojis* presentes nos *tweets*, para categorizar o texto dentro de uma emoção. Aqui, é feito um mapeamento dos *emojis* mais utilizados nos textos da rede social para uma das seis emoções básicas de Ekman. Este mapeamento é feito com recurso à descrição textual do próprio *emoji* bem como à sua representação gráfica.

A investigação de Hussien et al. (2016) procura verificar se estes métodos de extração e anotação automatizada fornecem melhores resultados do que se forem categorizados por anotadores humanos. Esta necessidade surge para verificar se o método automático de categorização resulta em dados de pior qualidade. Tal como as abordagens descritas em Hasan, Agu e Rundensteiner (2014), foram criados dois conjuntos de dados, ambos retirados com *emojis* e *emoticons* contidos nos *tweets*. Num desses conjuntos, uma de quatro emoções definidas, alegria, tristeza, raiva e nojo, é categorizada baseando-se nos *emojis* e *emoticons* presentes no texto. No outro conjunto, foram distribuídos os textos a anotadores humanos para fazerem a categorização manualmente. Com ambos os conjuntos de dados categorizados, seguiu-se o desenvolvimento de dois modelos iguais, cada um treinado com os dados obtidos. Desta forma tenta-se avaliar qual o método que atinge melhores resultados. Após o processo de treino e validação dos dados, verificou-se que no âmbito da experiência realizada, o processo automático obtém melhores valores.

---

<sup>3</sup>Tradução do termo inglês *label*

Este resultado deve-se, em grande parte, ao processo automático ignorar problemas de ambiguidade, e avaliação pessoal. Quando os dados eram submetidos a anotadores humanos, em muitos casos havia discordâncias, pois cada um interpretava o texto da sua forma. Assim, criava-se confusão, já que o mesmo texto teria, segundo diferentes anotadores, diversas emoções distintas. Esta conclusão foi obtida através dos resultados, quando submetidos com novos dados para classificar, dentro das quatro categorias definidas.

O trabalho de Hasan, Agu e Rundensteiner (2014) demonstra a problemática que é a anotação de uma emoção presente num texto. Tal como decorrido em Hussien et al. (2016), devido às experiências pessoais de cada indivíduo, a categorização de texto irá variar conforme a pessoa que está a efetuar a anotação. Isto constitui um problema, pois a categoria atribuída dificilmente será a correta, sendo que a pessoa mais indicada para fazer esta anotação seria o próprio autor do texto. Por outro lado quando o autor está a escrever, este não estará a pensar explicitamente na emoção que quer transmitir, mas que, apesar de tudo, deixa pistas, e que, entre elas, as pistas mais concretas acabam por ser as *hashtags* e os *emojis*.

O trabalho realizado por Suttles e Ide (2013) acaba por combinar todos estes novos métodos de comunicação. Esta junção permite assim a criação de dados ainda maiores e com mais qualidade, respetivos à categoria atribuída ao *tweet*. A junção de *hashtags*, *emojis* e *emoticons* permite uma categorização de uma das emoções presentes no modelo proposto por Plutchik (2001), mais precisa. Verificou-se ainda um aumento significativo no desempenho dos modelos desenvolvidos e treinados, com estes dados resultantes do processo automático.

Uma análise importante é efetuada por Wang et al. (2012), onde se utiliza estes métodos para criar um grande conjunto de dados. Verifica-se que a existência de um *hashtag* de categoria afetiva como **#love**, **#happy** ou de um *emoji* no fim do texto ou da frase, é um forte indicativo do estado emocional associado. Assim, o grande foco, foi o treino com dados correspondentes a esta característica, para serem aplicados no treino de modelos. Obtiveram-se resultados superiores aos de modelos treinados com base em *emojis* e *hashtags*, mas sem a verificação da posição de onde estes se encontram. Estes resultados demonstram que *hashtags* ou *emojis* presentes no fim de um *tweet* estão altamente relacionados com o teor emocional do texto.

Em Kunneman, Liebrecht e Bosch (2014) usaram-se métodos de extração de grandes quantidades de *tweets* para serem aplicados no treino de modelos. Estes dados baseiam-se na presença de cerca de 20 dos *hashtags* mais usados na língua neerlandesa. Desta forma, tentou-se efetuar uma previsão de novos dados, tendo como base a própria *hashtag*. Os resultados obtidos demonstraram que é possível prever um conjunto considerável destas *hashtags* e reforçam o conteúdo emocional dos textos. Também foi possível verificar que a maneira de criação de dados da forma automática, pode ser generalizada para qualquer idioma, ainda que considerando *hashtags* específicas e mais populares da língua alvo.

Para além da utilização destas novas formas de comunicar para a classificação do estado emocional, Van Hee, Lefever e Hoste (2018) focam-se em *tweets* irónicos. Este processo é feito através da presença das *hashtags* indicativas de ironia, como **#irony**, **#sarcasm**, **#not**. Os dados obtidos foram aplicados no treinos de dois modelos, um para classificar o texto presente em irónico ou não irónico, e outro para detetar o tipo de ironia presente. Uma abordagem diferente das anteriores é a dupla verificação dos dados retirados. Após a extração de dados, recorre-se a anotadores humanos para verificar se o texto retirado com base nos *hashtags* presentes, é efetivamente de carácter irónico ou não. Esta dupla verificação é importante, pois assim obtêm-se dados mais limpos e mais significativos para

aplicar ao processo de treino.

Tal como os problemas de análise de sentimento e classificação de emoções, a classificação de textos irónicos já foi abordada para a língua portuguesa. Vanin et al. (2013) focam-se em *tweets* publicados durante a semana de 22 de dezembro de 2012. A data foi escolhida devido à teoria que o mundo iria acabar de acordo com a previsão Maia, logo um tópico amplamente falado com tom irónico. Os autores recorreram à *hashtag* **#fimdumundo** para verificar se o texto seguia uma de quinze categorias definidas para ironia. Esta verificação tinha como base verificar se a frase obtida correspondia a uma das expressões regulares definidas. Através da forma da expressão regular era possível fazer a classificação de uma das quinze categorias existentes. Um trabalho semelhante (Carvalho et al., 2009), procura por dados irónicos em comentários ou menções de acordo com os termos da rede social, em notícias de tema político.

Olhando para os trabalhos anteriores, a criação de um grande conjunto de dados textuais com emoções associadas é um processo que pode ser construído e abordado de forma automática. Existem inúmeras abordagens para fazer esta categorização, e podem ser divididas em dois métodos principais: manualmente ou com recurso aos novos métodos de comunicação existentes. Na tabela 3.1 são resumidas as abordagens dos trabalhos analisados para a recolha automática de textos e emoções associadas.

Bibliografia	Língua	Anotação	Base	Previsão
Agarwal et al. (2011)	Inglês	Manual	Tweets	Sentimento
Pak e Paroubek (2010)	Inglês	Automático	Emoticons	Sentimento
Mozetič, Grčar e Smalović (2016)	18 Idiomas	Manual	Quantidade enorme de Tweets	Sentimento
Alm, Roth e Sproat (2005)	Inglês	Manual	Textos	Emoções Ekman
Barbieri et al. (2018b)	Inglês, Castelhana	Automático	Emojis	Emoji
Nascimento et al. (2012)	Português	Manual	Notícias	Sentimento
Araújo et al. (2013)	Português	Manual	Notícias	Sentimento
Souza e Vieira (2012)	Português	Automático	Hashtags	Sentimento
Mohammad e Kiritchenko (2015)	Inglês	Automático	Hashtags	Emoções Ekman
Wood e Ruder (2016)	Inglês	Automático	Emojis	Emoções Ekman
Hasan, Agu e Rundensteiner (2014)	Inglês	Automático	Hashtags	Valência e Ativação
Suttles e Ide (2013)	Inglês	Automático	Hashtag, Emojis, Emoticons	Emoções Plutchik
Arifin et al. (2014)	Indonésio	Manual	Hashtag, Emojis, Emoticons	Emoções Ekman
Wang et al. (2012)	Inglês	Automático	Hashtag	Emoções Plutchik
Hussien et al. (2016)	Arábico	Automático	Hashtag, Emojis	Feliz, Triste, Raiva, Nojo
Barbieri, Ballesteros e Saggion (2017)	Inglês	Automático	Emojis	Emojis
Felbo et al. (2017)	Inglês	Automático	Emojis	Emojis
Kunneman, Liebrecht e Bosch (2014)	Neerlandês	Automático	Hashtags	Hashtags
Vanin et al. (2013)	Português	Automático	Hashtags	Ironia
Dosciatti, Ferreira e Paraiso (2015)	Português	Manual	Notícias	

Tabela 3.1: Tabela da bibliografia existente sobre a recolha automática de dados

## 3.2 Léxicos Afetivos

Um léxico, por vezes denominado por dicionário, consiste num conjunto de vocábulos relevantes para um dado domínio, ordenados, juntamente com informação relacionada com o seu significado (e.g., definição, informação morfológica, afetiva). Na *social media*, termos em calão e abreviações, são usadas extensivamente, o que levou a certos termos serem considerados como expressões comuns do dia a dia. Estas considerações levaram a que, conseqüentemente tenham sido inseridos nos dicionários. Relativamente ao domínio

afetivo, o surgimento de palavras específicas relacionadas com o estado emocional da pessoa, começaram a ser implementadas nos léxicos existentes. Esta implementação deve-se à necessidade de classificar o estado emocional com base em texto, de onde surgiram léxicos dedicados a estas palavras. Estas expressões estão em grande parte relacionadas com a cultura do povo de onde são retiradas. A expressão “*Foi um bate-volta*” denota para um indivíduo português, uma emoção de carácter negativo como raiva, muito devido ao significado da palavra “bate”. Mas para alguém proveniente do Brasil, é simplesmente uma expressão a indicar que uma ação feita foi rápida. Exemplos destes termos, mas introduzidas pela *social media* é a expressão “*lol*”, vinda da abreviação da língua inglesa *laughing out loud*. A tradução direta para o português consiste em “Rir alto” e não tem muito nexos fora do contexto das redes sociais. No entanto este termo é utilizado diariamente em contextos de expressão de felicidade ou de riso.

Ao nível do próprio significado das palavras, a **WordNet** desenvolvida por Princeton é um dos léxicos mais utilizados para o inglês, mas com adaptações para diversas línguas. Este léxico é um dos recursos usados por Ma, Prendinger e Ishizuka (2005), onde efetua uma comparação das palavras nos textos que quer classificar, com aquelas presentes no léxico. Caso sejam iguais, usa os valores presentes no léxico para obter mais características sobre o texto em análise, que de outra forma não era possível.

Outros trabalhos basearam-se no conteúdo do **WordNet** e desenvolveram novos léxicos para a classificação de sentimento. Para tal, a cada palavra anotaram o sentimento associado, de maneira a ajudar o processo de classificação. Um destes léxicos é o **SentiWordNet** (Esuli e Sebastiani, 2007). Entre outros léxicos deste tipo encontram-se, dedicados para a associação do sentimento, o **SenticNet** (Cambria et al., 2010), **LIWC** (Tausczik e Pennebaker, 2010) e o **OpinionLexicon** (Ding, Liu e Yu, 2008). Todos estes léxicos, construídos utilizando abordagens diferentes, fornecem um conjunto de palavras e o sentimento associado a estas.

Para português, um dos léxicos mais conhecidos para a classificação de sentimento é o **SentiLex-PT** (Carvalho e Silva, 2015). É constituído por cerca de 18 000 palavras, verbos e adjetivos portugueses, a que está associado o valor de sentimento. Grande parte destes valores foram atribuídos manualmente, sendo que uma parte foi obtida de forma automática, através de algoritmos com base na distância entre as palavras. Outro léxico de sentimentos para o português é o **ReLi** (Freitas, 2013), construído com recurso a vários livros de autores portugueses, com recurso a anotadores humanos, onde para cada palavra indica o sentimento associada a este. Este sentimento foi obtido, verificando o carácter sentimental de cada palavra existente nos textos extraídos, sendo que tem a particularidade de esta classificação ser binária (positivo ou negativo). Importante realçar que esta classificação é binária, pois não contabilizam palavras com carácter sentimento neutro.

Os léxicos são ferramentas poderosas para ajudar na classificação de sentimento, tal como relatam Souza e Vieira (2012). Com recurso a um léxico, o **OpLexicon** juntamente com dados extraídos do *Twitter*, desenvolveram um modelo para a classificação de sentimento para pequenos textos. Este léxico é constituído por cerca de 15 000 palavras, onde para cada uma indica o sentimento associado. Os resultados obtidos não foram muito bons pois, os valores finais foram influenciados por uma má extração de dados, onde incluía alguns *tweets* em castelhano. Deste modo o modelo desenvolvido, não conseguia obter os valores de sentimento das palavras contidas nestes *tweets*, pois não existiam no conteúdo do léxico.

O processo de criação ou adaptação de léxicos, para efetuar a ligação entre palavra e estado emocional, foi também abordado de maneira a ajudar no problema da classificação

de emoção em texto. Um destes léxicos foi o **ANEW** (*Affective Norms for English Words*), que veio efetuar essa ponte entre palavra e emoção. Assim, a construção de modelos que permita a classificação de emoções, ficaram menos complexos de desenvolver. Isto deve-se ao **ANEW**, para cada palavra de língua inglesa, conter os valores medianos, de valência (*valence*) excitação (*arousal*) e dominância (*dominance*). Um trabalho que aplica este léxico foi o de Hasan, Agu e Rundensteiner (2014), que com recurso ao conteúdo existente no **ANEW**, treinam um modelo SVM para efetuar a classificação da emoção presente num texto. Analisando os resultados obtidos, os autores chegaram a conclusão que aplicando métodos de classificação de emoção com léxicos, estes são mais rápidos e mais eficientes, mas ignoram em grande parte o sentido de negação originando resultados contraditórios.

Adaptações do **ANEW** começaram a ser desenvolvidas para diversos idiomas (Montefinese et al., 2014; Redondo et al., 2007; Schmidtke et al., 2014) sendo também adaptada para o português Europeu, surgindo assim o **ANEW-PT** (Soares et al., 2012). Tal como o original, inclui um conjunto de palavras portuguesas e os valores de *valence*, *arousal* e *dominance* associados. O trabalho de Buechel e Hahn (2018b) descreve a construção do **EmoMap**, um léxico que abrange um total de oito idiomas, onde está incluído o português. No **EmoMap**, cada palavra está associada a um valor numérico relativo a cinco emoções (felicidade, raiva, tristeza, medo, nojo). Este valor é obtido através de um método desenvolvido em Buechel e Hahn (2018a), onde é efetuado um mapeamento entre vários léxicos existentes para valores VAD.

A crescente utilização de *emojis* e de *emoticons* em textos publicados nas redes sociais, levou à necessidade de também os considerar na tarefa de reconhecimento de emoções. Os *emojis* e *emoticons* são designados por pictogramas, imagens que transmitem a ideia de uma palavra ou frase e têm normalmente um valor emotivo associado, sendo que o último usa a conjugação de caracteres para fazer esse pictograma (e.g. 😊 : ( :D ) (Gustafsson, 2017). Assim surgiu a necessidade de construir um léxico dedicado à associação de *emojis* e *emoticons* com a respetiva carga emotiva. O trabalho apresentado em Rodrigues et al. (2018) relata a construção do **LEED** (*Lisbon Emoji and Emoticon Database*), que com um questionário efetuado a mais de 500 pessoas de nacionalidade portuguesa, conseguiram associar valores de VA, para cada *emoji* ou *emoticon*.

A tabela 3.2 resume todos os léxicos de sentimento/emoções identificados na fase de análise da bibliografia existente, sendo que alguns não foram explorados devido a serem bastante semelhantes, aos abordados neste capítulo.

Referência bibliográfica	Nome do léxico	Valores Contidos
Carvalho e Silva (2015)	SentiLex-PT	Palavra, Categoria, sentimento (Positiva, Neutra, Negativa)
Freitas (2013)	ReLi	Palavra, Categoria sentimento (Positiva, Negativa)
Pennebaker, Francis e Booth (2001)	LIWC	Palavra, sentimento (Positiva, Neutra, Negativa)
Costa (2012)	EmoTaix.PT	Palavra, Valence
Bradley e Lang (1999)	ANEW	Palavra, Valence, Arousal, Dominance
Soares et al. (2012)	ANEW PT	Palavra, Valence, Arousal, Dominance
Rodrigues et al. (2018)	LEED	Emoji, Emoção
Buechel e Hahn (2018b)	EmoMap	Palavra, Emoção
Mohammad e Turney (2013)	NRC Emotion	Palavra, Sentimento, Emoção
Mohammad e Kiritchenko (2015)	Hashtag Emotion Lexicon	Hashtag, Sentimento, Emoção

Tabela 3.2: Tabela da bibliografia encontrada referente aos léxicos existentes

### 3.3 Modelos Utilizados para o Reconhecimento de Emoções

Com os avanços da Inteligência Artificial, houve um consequente crescimento no número de modelos de aprendizagem, seguindo diferentes arquiteturas. Estes modelos têm sido aplicados a diversos problemas tais como a classificação de itens em classes, de acordo com as suas características. O problema da classificação de emoções, com base em texto não é diferente. Ao longo do tempo, têm sido apresentados modelos com diversos resultados na tarefa de classificar emoções. Assim uma referência a alguns destes a estes modelos, é importante para o trabalho a realizar.

Os primeiros modelos para a classificação no domínio da computação afetiva foram dedicados à classificação do sentimento associado, em textos na língua inglesa. O trabalho de Pak e Paroubek (2010) começa por apresentar o problema da classificação de sentimento e as suas aplicações. De seguida descreve como recolhe dados da rede social *Twitter* de maneira automática, com base nos *emoticons* presentes nos textos extraídos, para aplicar no treino do modelo desenvolvido. O problema da classificação de sentimento raramente é binário. Por normalmente haver três classes positivo, negativo e neutro, é necessário desenvolver classificadores multi-classe. O autor desenvolveu dois modelos, um baseado em SVM e outro baseado em Naive Bayes. Para treinar os modelos, usou como características (*features*)  $n$ -gramas. Devido à utilização destas *features*, começaram a surgir os primeiros desafios, devido ao valor de  $n$  a atribuir. Foi verificado que o treino usando unigramas é interessante, pois consegue cobrir todos os dados existentes, mas bigramas e trigramas detetam melhor expressões sentimentais. Para os modelos desenvolvidos, os melhores resultados foram obtidos usando bigramas num classificador Naive Bayes. Este resultado é explicado, pois usando bigramas aborda-se o texto muito semelhante a uma análise termo

a termo, e mesmo assim conseguir detetar expressões sentimentais presentes no texto.

As investigações efetuadas por Nascimento et al. (2012) foram semelhantes, mas já com o foco na Língua Portuguesa, mais concretamente o português do Brasil. Em vez de se focar na análise de sentimento em textos sem especificação do tema, foi efetuada a extração de *Tweets* de três categorias distintas, **Política**, **Entretenimento** e **Policial** (corresponde a eventos relacionados com ações da polícia brasileira). Esta extração foi efetuada automaticamente e os dados foram depois categorizados manualmente. Após a extração de dados, tentou-se perceber quais os melhores modelos para classificar um texto em sentimento positivo ou negativo. Aplicados a um modelo probabilístico de Naive Bayes, analisaram-se quais as melhores *features* a explorar. Ficou definida a utilização de n-gramas, nomeadamente unigramas e octogramas. A utilização de n-gramas com distancia considerável deve-se ao facto de tentarem verificar se o conteúdo sentimental se localiza, em termos individuais ou na totalidade de cada frase. Os resultados não foram muito conclusivos, mas em termos de valores, aqueles obtidos com base na utilização de octogramas, foram ligeiramente superiores do que aplicando unigramas.

Em Martinazzo, Dosciatti e Paraiso (2011) efetua-se a classificação de emoções mas usando um algoritmo baseado em *Latent Semantic Analysis* (LSA). Usando um corpus de notícias anotado com as seis emoções básicas é efetuado uma série de transformações aos textos antes de ser submetido ao algoritmo. As primeiras transformações são operações normais de NLP, sendo os textos passados todos a palavras minúsculas bem como pontuações, *stopwords* e números removidos. Depois os textos são aplicados a um processo de *stemming* de forma a manter as palavras semelhantes juntas. Com os termos resultantes dos textos é então efetuado uma transformação do texto para forma vetorial usando o método de *term frequency* seguido de uma técnica de *Singular Value Decomposition* (SVD). O SVD é uma técnica que reduz a dimensionalidade dos vetores obtidos com *term frequency*, e que dá mais peso a termos relacionados dentro do conjunto de textos de forma a manterem essa similaridade. Desta forma os autores obtêm um conjunto de coordenadas, que indicam os centros dos grupos obtidos para as seis emoções de Ekman. A classificação de novos dados é feita através do cálculo do coseno do ângulo, do texto que desejamos classificar com os grupos obtidos da técnica de SVD, de onde se obtiveram resultados com uma média de 71% de *accuracy*.

Também para o Português do Brasil surgiu o trabalho de Dosciatti, Ferreira e Paraiso (2013), que efetuam a classificação da emoção de um texto usando modelos SVM, Naive Bayes e K-NN. Este trabalho também usa um corpus de notícias em português do Brasil onde treina os modelos para classificar novos textos numa das seis emoções de Ekman. Os textos presentes no corpus são representados através de um vetor TF-IDF e consequentemente aplicados ao treino dos modelos. Os resultados obtidos para a classificação usando o modelo SVM foi aproximadamente de 60% para a métrica F1, que foi o melhor dos três modelos desenvolvidos. Um ponto que o autor aborda é relativo à inexistência de literatura para a língua portuguesa, sobre a classificação das seis emoções básicas de Ekman para textos. Esta falta de literatura levou a que fossem desenvolvidos os três modelos, de forma a poder efetuar comparações.

O desenvolvimento de modelos para a classificação do estado emocional associado a um texto, foi um pequeno salto dos modelos desenvolvidos para a classificação de sentimento. No trabalho de Wood e Ruder (2016), a extração de dados para aplicar no treino de um modelo de aprendizagem supervisionada, foi efetuada com recurso ao *Twitter*. Com estes dados foram desenvolvidos seis modelos SVM binários (um para cada uma das seis emoções básicas de Ekman). Para a fase de treino dos modelos, foram usadas como *features* n-gramas, indo até 5-gramas. Os resultados obtidos foram o reflexo dos dados de acordo

com o autor. O desbalanceamento dos dados resultou na existência de muitos dados correspondentes à classe de felicidade e de tristeza. Desta forma, os modelos resultantes tiveram um bom desempenho a detetar estas duas emoções, mas relativamente fraco para as restantes, levando a baixos resultados a nível da generalização do modelo.

Em Danisman e Alpkocak (2008), a classificação de emoções no texto é abordada de maneira diferente das até agora apresentadas, pois apresenta uma classificação com base em *Vector Space Model* (VSM). O conjunto de dados aplicados no trabalho foi obtido de diversos léxicos, e consequentemente processados, aplicando a transformação de formas negativas em *tokens* específicos. Outro processo aplicado foi a remoção de *stop words*, palavras listadas que normalmente não acrescentam informação ao texto, com o objetivo de obterem dados sem conteúdo desnecessário. De maneira a poder efetuar comparações entre os classificadores normalmente aplicados, foram construídos dois modelos, um com base em SVM e outro Naive Bayes. No final da experiência foram comparados os três modelos onde se verificou, que os modelos desenvolvidos obtiveram todos resultados semelhantes. Assim, não se concluiu a existência de um modelo melhor que os restantes, mas provou-se que um modelo VSM, consegue ter o mesmo desempenho que os modelos de arquiteturas mais aplicadas.

Devido à associação de *emojis* a emoções ser bastante significativa, e verificando que são usados em grande parte para a criação de dados categorizados, a sua exploração na classificação de emoções foi natural. A investigação feita por Barbieri, Ballesteros e Saggion (2017), aborda esse tema de uma maneira bastante superficial, onde constrói um conjunto de dados com base nos vinte *emojis* mais utilizados no *Twitter*. Após a etapa de processamento dos dados, são selecionados os *tweets* que contenham apenas um *emoji* no final. O autor impõe esta restrição, devido ao facto que desta forma, a relação entre o texto e o *emoji* é mais forte. Para tentar prever um *emoji*, foi criada uma rede neuronal (*Feed Forward Neural Network*), no qual obteve resultados significativos, para os dados introduzidos inicialmente. Para comparar os resultados com o desempenho humano, analisaram-se os valores obtidos da rede criada com aqueles obtidos do modelo, mas os dados aplicados ao treino foram categorizados por seres humanos. Chegou-se a várias conclusões significativas. Devido à forma como selecionamos os *emojis* em que são iguais, diferindo apenas na cor como exemplo os corações (❤️🧡💜), serem quase sempre selecionados de forma aleatória, essa seleção é transposta para o modelo de previsão do *emoji*. Assim o modelo não consegue fazer a sua previsão corretamente, e prevê a variante do *emoji* aleatoriamente. Outro resultado importante e inesperado, é a capacidade de fazer associações com poucos dados, associando *emojis* da época natalícia como a árvore de Natal ou do Pai Natal à palavra Natal.

Outros trabalhos como Felbo et al. (2017), abordaram a questão de previsão de *emojis* usando redes neuronais com arquitetura LSTM, no qual o autor deu o nome de *Deepemoji*. A rede desenvolvida consta com parâmetros diferentes das redes LSTM normalmente aplicadas, e apresenta os detalhes mais concretos da rede. Descreve a utilização e descrição de uma *embedding layer*, com 256 dimensões e uma função de ativação de tangente hiperbólica, para converter os dados num intervalo compreendido entre -1 e 1 na criação da rede. De maneira a interpretar os contextos de cada palavra presente para classificação, são definidas duas camadas com o total de 1024 unidades. Cada uma destas camadas, conta com 512 unidades para cada direção, em relação ao sentido da frase (sentimento positivo ou negativo). Depois deste passo, são transformadas por outra camada, designada por camada de atenção. De maneira a verificar os resultados obtidos, efetuou-se a comparação entre os valores resultantes do modelo desenvolvido, com os dados conseguidos de anotadores humanos, no qual se verificou que o modelo obteve resultados mais precisos que os anotadores humanos.

Também Barbieri et al. (2018a), trabalhou no problema de previsão de *emojis*, seguindo uma abordagem semelhante. Uma maneira de conseguir ultrapassar o problema de certas associações palavra/*emoji*, como casa e *emoji* de casa, foi melhorar os modelos baseados em LSTM. As abordagens desenvolvidas até agora, não tinham em conta da probabilidade de ocorrência de mais do que um *emoji*, ocorrer no resto do *tweet*. Aqui a abordagem efetuada para resolver este problema, foi a criação de uma rede neuronal LSTM sendo que, quando é feita uma análise aos n-gramas, tem-se a atenção quando existe a possibilidade de ocorrência de vários *emojis* significativos. Um exemplo dado foi o caso do *tweet* “Rezo para que caía neve amanhã”. Os modelos desenvolvidos até agora iriam prever a ocorrência de um *emoji* associado com rezar (🙏) (dando como exemplo o caso das redes com a arquitetura *Deepemoji*). Isto deve-se ao facto de, ser mais frequente a sua utilização e a sua existência, nos dados aplicados para treino. A rede desenvolvida diverge das restantes, pois dá a possibilidade de ocorrência tanto do *emoji* de rezar, como o *emoji* do floco de neve (❄️), devido à ocorrência do verbo “nevar”, prevendo assim a ocorrência de *emojis* que os modelos desenvolvidos até agora dificilmente seriam capazes de detetar.

A arquitetura dos modelos desenvolvidos, costuma variar de acordo com o problema em questão, e na generalidade, apesar de haver modelos com utilização mais frequente, não significa que outras abordagens não obtenham melhores resultados. Como tal, na tabela 3.3 apresenta-se, de forma resumida, um enquadramento da bibliografia relacionada com o problema da classificação de emoções.

<b>Bibliografia</b>	<b>Features Utilizadas</b>	<b>Modelo Aplicado</b>	<b>Classes</b>
Barbieri et al. (2018b)	N-gramas, Emojis, Diferente Idiomas	SVM, Random Forests	Emojis
Nascimento et al. (2012)	Unigramas, Octogramas	Naive Bayes	Sentimento
Pak e Paroubek (2010)	Unigramas, Bigramas	Naive Bayes, SVM	Sentimento
Buechel e Hahn (2018b)	Valence, Arousal Dominance	K-NN	Emoções
Wood e Ruder (2016)	Unigramas, Bigramas Trigramas, 4-gramas, 5-gramas	SVM	Emoções
Hasan, Agu e Rundensteiner (2014)	Valence, Arousal, Léxicos	SVM, K-NN	Emoções
Arifin et al. (2014)	Hashtags, Emojis, Adjetivos, Emoticons	K-NN	Emoções
Wang et al. (2012)	Unigramas, Bigramas Trigramas, Léxicos, Hashtags	Naive Bayes	Emoções
Binali, Wu e Potdar (2010)	Unigramas	SVM	Emoções
Hussien et al. (2016)	Emojis, Emoticons	SVM, Naive Bayes	Emoções
Danisman e Alpkocak (2008)	Unigramas	SVM, VSM, Naive Bayes	Emoções
Barbieri, Ballesteros e Saggion (2017)	Emojis	Recurrent Neural Network, Bag of Words, Kip-Gram Vector Average	Emojis
Barbieri et al. (2018a)	Unigramas, Bigramas, Emojis	LSTM Neural Network	Emojos
Felbo et al. (2017)	N-gramas, Emojis	LSTM Neural Network	Emojis
Kunneman, Liebrecht e Bosch (2014)	Unigramas, Bigramas Trigramas	Ballanced Winnow	Hastags, Emoções
Suttles e Ide (2013)	N-gramas, Emojis, Hashtags	Naive Bayes, Maximum Entropy	Emoções
Tato, Nkambou e Frasson (2018)	ECG	LSTM Neural Network	Emoções
Martinazzo, Dosciatti e Paraiso (2011)	Unigramas Term Frequency	LSA	Emoções
Dosciatti, Ferreira e Paraiso (2013)	Unigramas TF-IDF	Naive Bayes, SVM, K-NN	Emoções

Tabela 3.3: Tabela da bibliografia encontrada referente aos modelos aplicados

Esta página foi intencionalmente deixada em branco.

# Capítulo 4

## Abordagem

Neste capítulo é descrito o trabalho realizado, isto é, o desenvolvimento dos modelos para efetuar a classificação de emoções com base em texto para a língua portuguesa. De um modo geral, o trabalho envolveu uma recolha de dados, a sua análise, o treino de modelos de aprendizagem e, finalmente, a sua avaliação. As secções seguintes irão descrever o trabalho realizado, bem como explicar os motivos das decisões tomadas que resultaram no conjunto de dados e nos modelos desenvolvidos. Para os modelos estão apresentados todos os parâmetros importantes, e ainda uma breve explicação de como estes foram otimizados.

### 4.1 Criação do conjunto de dados

Para qualquer processo de aprendizagem supervisionada, os dados aplicados no treino do modelo desenvolvido são cruciais para que este consiga obter um bom desempenho. Para além disso, é sempre desejado uma boa capacidade de generalização. Assim, o processo de criação do conjunto de dados foi aplicado com o máximo cuidado possível de maneira a tentar obter um resultado final com qualidade e com a maior diversidade possível. Esta diversificação foi efetuada para tentar abranger o máximo de textos possíveis, escritos em português, tanto na sua variante europeia como brasileira. Assim, tenta-se encontrar expressões de carácter emocional aplicáveis a qualquer uma das variantes, independentemente das diferenças culturais significativas que existem entre as duas que permita identificar mais facilmente a emoção associada ao texto.

Todos os dados recolhidos e usados neste trabalho foram obtidos a partir do *Twitter*, sendo que para cada texto (*tweet*) foram extraídas várias características (*features*). Esta extração foi efetuada de maneira a tentar recolher o máximo de informação possível sobre cada texto. Os *tweets* foram recolhidos com recurso a uma biblioteca na linguagem *Python*, o *Tweepy*<sup>1</sup>, que com as credenciais de *developer* do *Twitter*, efetua uma ligação à *TwitterAPI*<sup>2</sup> de forma facilitada.

Com o *Tweepy* desenvolveu-se um método, que permite a monitorização e extração dos *tweets* em tempo real, com base numa série de condições. O texto dos *tweets* tem de estar na língua portuguesa, sendo que aqui a variante não é especificada porque a *Twitter API* não consegue efetuar a distinção entre as duas. O *tweet* tem de conter pelo menos uma *hashtag* ou um dos *emojis* da lista definida no trabalho de Wood e Ruder (2016), visível na tabela 4.1.

---

<sup>1</sup><http://www.tweepy.org>

<sup>2</sup><https://developer.twitter.com>

Hashtags	Emojis
#irritado	
#nojo	
#medo	
#feliz	
#triste	
#surpresa	

Tabela 4.1: Lista de *emojis* e de *hashtags* aceites para armazenamento dos *tweets*

Caso o *tweet* contenha pelo menos uma destas características, a seguinte informação será armazenada:

- ID do *Tweet*
- Autor do *Tweet* (Nome do utilizador)
- Data de criação
- Texto, onde estão incluídos *emojis* e/ou *hashtags*
- Emoção associada, com base no emoji ou hashtag utilizados
- Geo-localização associada ao *tweet*

No caso de o *tweet* conter uma das *hashtags* que correspondem às 6 emoções básicas de Ekman (1999), nomeadamente #irritado, #nojo, #medo, #feliz, #triste e #surpresa, é efetuado o seguinte. Para o campo da emoção, a categoria atribuída é a correspondente ao *hashtag* sem o carácter #. Esta decisão teve como base o trabalho de Mohammad e Kiritchenko (2015), onde reproduzimos a forma de categorização da emoção do texto, com o *hashtag* presente.

No caso do *tweet* conter um dos *emojis* assinalados no texto, a categorização da emoção é feita de acordo com a divisão definida no trabalho de Wood e Ruder (2016). Desta forma, a classe a que o *emoji* pertence corresponde à emoção atribuída ao *tweet*. Para além das classes definidas, foram adicionados alguns *emojis* que durante o desenvolvimento do trabalho de Wood e Ruder (2016) ainda não estavam implementados. Estes foram os *emojis* de vômito (🤢 🤮), que transmite o carácter emocional de nojo e não era considerado. A divisão dos *emojis* em classes é apresentada na tabela 4.2.

A atribuição de uma emoção a um *emoji* é feita com base no que a sua representação gráfica transmite, bem como o seu nome e descrição<sup>3</sup>. Devido ao facto de cada *emoji* ter uma representação diferente para cada sistema operativo, marca ou plataforma de comunicação (e.g., *WhatsApp*, *Facebook messenger*, *Twitter*), a tabela 4.3 apresenta a mesma divisão de *emojis*, mas utilizando a sua representação em *Unicode* em vez da sua representação gráfica.

<sup>3</sup>Lista de *emojis*, nome e descrição de cada um, disponível em <https://emojipedia.org/people/>

Emoção	Emojis
Felicidade	
Raiva	
Nojo	
Medo	
Tristeza	
Surpresa	

Tabela 4.2: Divisão dos *emojis* em emoções

Emoção	Emojis
Felicidade	U+1F600, U+1F602, U+1F603, U+1F604, U+1F606, U+1F607, U+1F609, U+1F60AU+1F60B, U+1F60C, U+1F60D, U+1F60E, U+1F60F, U+1F31E, U+263A, U+1F618 U+1F61C, U+1F61D, U+1F61B, U+1F63A, U+1F638, U+1F639, U+1F63B, U+1F63C U+2764, U+1F496, U+1F495, U+1F601, U+2665
Raiva	U+1F62C, U+1F620, U+1F610, U+1F611, U+1F620, U+1F621, U+1F616, U+1F624 U+1F63E
Nojo	U+1F4A9, U+1F92E, U+1F922
Medo	U+1F605, U+1F626, U+1F627, U+1F631, U+1F628, U+1F630, U+1F640
Tristeza	U+1F614, U+1F615, U+2639, U+1F62B, U+1F629, U+1F622, U+1F625, U+1F62AU+1F613, U+1F62D, U+1F63F, U+1F494
Surpresa	U+1F633, U+1F62F, U+1F635, U+1F632

Tabela 4.3: Divisão dos códigos *unicode* dos *emojis* em emoções

Atribuída a emoção ao *tweet*, toda a informação descrita anteriormente é armazenada para uso posterior. A extração de dados foi efetuada no período correspondente desde **outubro de 2018** até **março de 2019**, de maneira a ter um tamanho considerável em termos de conteúdo. Para além deste ponto, é evitado a ocorrência de demasiados *tweets* sobre um determinado tema ou evento, devido ao período de extração ser efetuado durante vários meses. Outro ponto importante é que a extração de dados desta forma permitirá a criação de diferentes conjunto de dados, considerando o período em que foram publicados.

## 4.2 Limpeza de dados

Após efetuar a extração de dados é necessário proceder à sua limpeza, pois o texto contém várias ocorrências que precisam de ser removidas ou normalizadas, contribuindo assim para a redução de dados não relevantes. Estes dados consistem em textos ou termos que não acrescentam nenhuma informação para o conjunto geral e levam a que seja preciso mais tempo para processar todo o conjunto. A remoção destes dados leva a que os modelos finais sejam mais precisos e demorem menos tempo no processo de treino.

Por exemplo, o texto dos *tweets* contém expressões com significado apenas dentro da rede social *Twitter*. Uma destas expressões é a menção/identificação de um utilizador da rede, sendo esta representação feita através do identificador “@” seguindo do nome de utilizador. Outra ocorrência comum nesta rede social é a partilha de *tweets*, neste contexto designados por *retweets*, iniciados pelo marcador @RT. Além destas ocorrências, a partilha de URL com imagens ou vídeos também é bastante comum. Este é um ponto bastante importante, porque o conteúdo do URL pode contribuir para o teor emocional do *tweet*, o que pode influenciar o desempenho final do modelo. Estas ocorrências terão de ser tratadas de forma a conseguir obter um conjunto de dados para atingir os objetivos pretendidos.

Outro passo importante é remover do conjunto de dados *tweets* mal formados ou “emocionalmente ambíguos”. Devido ao método como os dados são extraídos e tratados (descrito na secção 4.1), os *tweets* presentes neste conjunto podem conter *emojis* ou *hashtags* que pertençam a emoções distintas. Um exemplo será a ocorrência de um *emoji* sorridente (😊), que classificaria o texto com a emoção feliz, e no mesmo *tweet* a presença de um *emoji* de cor vermelha (😡) que, por sua vez, indicaria uma expressividade de raiva. Num *tweet* já considerado como feliz, isto representa uma contradição. Este conjunto específico de *tweets* também representa alguns casos de ironia, tornando assim o texto difícil de classificar. Exemplos de casos de *tweets* contraditórios são visíveis na tabela 4.4.

Vou fazer mais uma tatoo... unica coisa boa desse mês 😞😍
bolsominions q pagam pau pros EUA 🤔 É FAKE NEWS ❌ TRUMP MELHOR PRESIDENTE 😍
ah que feriadinho bom... em casa e com chuva 😍 só faltou moção 😞

Tabela 4.4: Exemplos de textos contraditórios

*Tweets* com pouco conteúdo também não apresentam grandes benefícios, sendo que um conjunto de dados com este tipo de *tweets* poderá levar a um mau desempenho do modelo, nomeadamente na sua capacidade de generalização. Muitos deste *tweets* consistem numa palavra curta, onde a sua ocorrência levará a que o modelo classifique novos textos contendo esta palavra de forma errada pois, estão muito replicados no conjunto de dados. Este tipo de textos também levará a que os modelos desenvolvidos demorem mais tempo para efetuar o seu treino.

Tendo em conta estes problemas, o tratamento destes casos é crucial para que o conjunto de dados final tenha a maior qualidade possível. Isto é importante para que futuramente, quando aplicado ao processo de treino, os modelos para classificar a emoção obtenham um bom desempenho. Para isso, foi desenvolvido um processo que remove estas ocorrências, de maneira a deixar o conjunto final mais compacto e com mais qualidade.

O primeiro passo foi a remoção de *tweets* ambíguos do conjunto de dados. O mesmo é feito para a existência de *retweets*, que por serem duplicados só estão a replicar informação já existente e a contribuir para um enviesamento dos dados. Por fim, *tweets* com pouco conteúdo são removidos, sendo que aqui são considerados aqueles onde o texto contenha menos de 5 palavras na sua totalidade. Grande parte destes *tweets* são designadas por mensagens correntes, dentro da rede social e são casos que não transmitem grande teor emocional. Por fim, no caso da identificação de utilizadores e a existência de URL o texto é alterado substituindo os identificadores da rede social pelos termos “@USERNAME” e “@URL” respetivamente.

O conjunto de dados resultante deste processo foram 1.422.158 *tweets* de um total de 2.004.129 *tweets*. Pela redução dos números, verificamos que o conjunto de dados original inclui muitos *retweets* e uma quantidade significativa de *tweets* ambíguos. Também é

importante ter uma noção de como o conjunto de dados final se encontra distribuído, relativamente às classes de emoções. Esta distribuição é visível na figura 4.1.

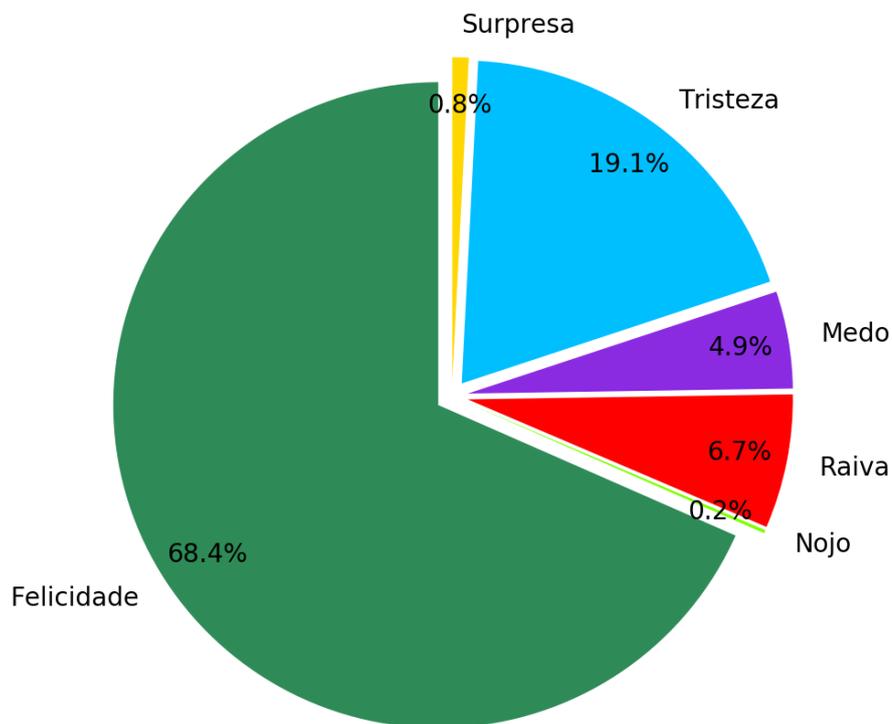


Figura 4.1: Distribuição de *tweets* por emoção

Constituindo mais de metade do conjunto de dados final, a predominância da emoção de felicidade é bem visível. Isto deve-se a diversos fatores, sendo o principal a quantidade de *emojis* classificados para esta classe ser maior que as restantes. Outro aspeto, relacionado com o ponto anterior, deve-se à popularidade de utilização que os *emojis* pertencentes à classe de felicidade ser também superior à dos restantes<sup>4</sup>. Uma última situação está relacionada com o conteúdo partilhado nas redes sociais, onde os utilizadores têm maior tendência em partilhar conteúdo alegre.

Esta relação popularidade/*emoji* também é verificada analisando a divisão dos *tweets* presentes no conjunto de dados final, visível na figura 4.2.

---

<sup>4</sup>A utilização destes *emojis* no *Twitter* pode ser verificada em <http://emojitracker.com>



fornecida pela biblioteca *Natural Language Toolkit*<sup>5</sup>.

- A outra foi remover do conjunto total de termos, aqueles que só têm apenas uma ocorrência. Estes termos correspondem em quase toda a sua totalidade palavras com erros ortográficos, ou termos sem nexos, como é o caso das palavras “*tempurada*” ou “*descaregando*”, que correspondem a erros de ortografia.

Para além da aplicação do TF-IDF a termos singulares, este também foi aplicado a bi-gramas e a tri-gramas. O principal objetivo aqui será uma tentativa de incluir expressões que em muitos casos indicam a emoção existente. Exemplos destas expressões são os casos de “*gosto muito*” e “*queria tanto*” que ocorrem com alguma frequência e encontram-se, em quase toda a sua totalidade, associadas às emoções de felicidade e tristeza respetivamente. Destas expressões podemos verificar que alguns termos específicos estão muito relacionados com a emoção que lhe é atribuída, pelo processo de anotação dos dados. De forma a verificar estes pequenos padrões entre termos e emoções foi efetuado um teste de  $\chi^2$ . Este teste permite associar a cada termo um valor numérico indicativo da relevância que este tem com a classe (emoção) selecionada. Para o conjunto de dados foi então efetuado o teste de  $\chi^2$  relacionando os termos provenientes do TF-IDF com uma das seis emoções básicas de Ekman (1999). Assim tenta-se obter respostas concretas sobre a ligação entre os termos existentes e as emoções definidas. Para cada emoção, a tabela 4.5 mostra os termos mais relevantes para cada emoção (identificada através do *emoji* usado), de acordo com o teste do  $\chi^2$ .

Emoção	Termos mais relevantes
Felicidade	linda, adoro, amo
Raiva	fdp, odeio, porra
Nojo	lixo, Bolsonaro, merda
Medo	mentira, medo, temo
Tristeza	chorar, queria, dor
Surpresa	sério, amo, gostei

Tabela 4.5: Termos mais relevantes por emoção

A tabela 4.5 mostra que os termos assinalados são adequados às emoções em questão. Um termo que se destaca é “Bolsonaro”, referente ao presidente em vigor do Brasil. Este termo tem uma grande relevância para a emoção de nojo devido ao período em que ocorreu a extração de dados. A começar pelo facto da extração ter sido feita durante as eleições presidenciais e uma grande percentagem de utilizadores no conjunto de dados serem do Brasil, bem como todas as polémicas que ocorreram com o candidato e atual presidente Bolsonaro. Assim, a partilha de textos de teor negativo e repulsivo tomaram uma quantidade significativa do conjunto total de *tweets*. A isto também não será alheio o facto da emoção de nojo ser a classe menos representada no conjunto de dados, o que aumenta o impacto do termo em questão.

Analisando os restantes termos, verificamos que correspondem às emoções atribuídas, denotando que o termo “fdp” corresponde a uma abreviação de uma expressão ofensiva de carácter negativo. Desta forma verifica-se a existência de relações que podem ser exploradas para efetuar a classificação de emoções de forma mais facilitada.

<sup>5</sup><https://www.nltk.org>

## 4.4 Extração de características

Os vetores TF-IDF permitiram-nos representar o nosso conjunto de dados numa forma numérica. No entanto, para além destes vetores, existe uma quantidade de características adicionais que podemos explorar, com o objetivo de retirar mais informação dos textos obtidos e, idealmente, contribuir para uma melhor classificação. Aqui incluímos algumas características adicionais do tweets, não capturadas pelos vetores TF-IDF, nomeadamente:

- Quantidade de palavras negadas (“não”, “nem”, “nunca”, “jamais”, “nenhum”) do *tweet*
- Quantidade total de palavras do *tweet*
- Quantidade de letras da maior palavra do *tweet*
- Valor médio de letras nas palavras do *tweet*
- Quantidade de pontos de exclamação do *tweet*
- Quantidade de pontos de interrogação do *tweet*
- Quantidade de palavras em maiúsculas do *tweet*

Procuramos ainda tirar partido de recursos externos, nomeadamente de léxicos de sentimento e emoção. Recorda-se que estes léxicos associam palavras em português a valores associados à emoção que tipicamente transmitem. Extraímos então as seguintes características deste tipo:

- Os seguintes valores dos léxicos NRC-VAD e ANEW-PT:
  - Valência média de todas as palavras do *tweet*
  - Excitação média de todas as palavras do *tweet*
  - Dominância média de todas as palavras do *tweet*
- O seguinte valor do léxico SentiLex-PT:
  - Sentimento médio de todas as palavras do *tweet*

Os primeiros pontos dizem respeito a várias contabilizações que fornecem alguma informação adicional, sobre o texto em questão. A contabilização do número total de palavras indica que textos maiores têm tendência a ser opiniões mais formadas sobre determinado assunto. Em contraste, *tweets* pequenos são mais relacionados com a interação entre dois ou mais utilizadores da rede social. A contabilização dos caracteres da maior palavra também é um indicativo do teor do texto. Por fim, acreditamos que o total de pontos de exclamação e de interrogação bem como a quantidade de palavras onde os caracteres sejam todos maiúsculos são um forte indicativo de teor expressivo e emocional no texto. Exemplos de tais textos encontram-se presentes na tabela 4.6.

Em relação aos segundos pontos, todos os *tweets* presentes no conjunto de dados são submetidos a uma análise para cada palavra presente. Esta análise consiste em verificar se a palavra existe nos léxicos NRC-VAD<sup>6</sup> e ANEW-PT<sup>7</sup> e caso se encontre, são guardados os

<sup>6</sup>Disponível em <https://saifmohammad.com/WebPages/nrc-vad.html>

<sup>7</sup>Disponível em <http://p-pal.di.uminho.pt/about/databases>

valores de valência, excitação e dominância presentes. Caso a palavra não esteja contida no léxico é atribuído o valor 0. Após processar todas as palavras do *tweet* é efetuada uma média dos valores que será o valor de cada *feature*.

O mesmo processo é efetuado mas para a verificação do sentimento usando o léxico SentiLex-PT<sup>8</sup>, onde o valor presente é 1 (positivo), 0 (neutro), -1 (negativo) denotando o sentimento da palavra. Após processar todas as palavras, a característica terá o valor do sentimento com maior ocorrência. Em caso de empate, a característica terá o valor 0 (neutro). Tanto para os valores de emoção como de sentimento, sempre que uma palavra com valores associadas é precedida de uma negação, esse valor é substituído pelo seu simétrico. As palavras consideradas para as negações são as mesmas consideradas para a característica relacionada.

Texto	Emoção Atribuída
@USERNAME Namorar de manhã e tão gostoso, nossa!!!!!! 😍😍	Felicidade
Que ventania é essa????? 🤔🤔	Medo
AI MEU DEUS VCS VOLTARAM AHHHHHHH TÔ TÃO FELIZ — VOLTAMOS AAA 💖 @URL	Felicidade

Tabela 4.6: Exemplos de ocorrência de pontuação expressiva

Verifica-se que a alta quantidade de pontos de exclamação e de palavras maiúsculas realçam ainda mais o teor emocional relativo à felicidade que os utilizadores transmitem no texto. Já os pontos de interrogação estão muito relacionados com a emoção de medo, realçando o sentimento de uma pergunta lançada aos seguidores do autor do *tweet*. A ocorrência de muitas palavras com letras maiúsculas parece relacionada com a emoção de raiva. Nestes casos os utilizadores expressam muitas vezes a sua frustração, recorrendo a este tipo de palavras.

## 4.5 Modelos para o Reconhecimento Automático de Emoções

De forma a atingir o objetivo de classificação da emoção presente num texto na língua portuguesa, e ainda para efetuar a previsão de *emojis*, foi necessário a criação de diversos modelos. Alguns destes modelos deverão processar o texto e decidir a que classe este pertence, enquanto outros deverão ir a um nível mais profundo e prever o *emoji* mais adequado. Como referido na secção 1.3, a criação de um ou mais modelos e a comparação de arquiteturas distintas é uma das contribuições esperadas. Destas arquiteturas decidiu-se utilizar três modelos bastante utilizados em tarefas de classificação de texto, nomeadamente modelos baseados em Naive Bayes, Support Vector Machine e numa rede neuronal do tipo Long Short Term Memory (LSTM). A escolha destas arquiteturas deve-se, em grande parte, à existência de trabalhos com resultados significativos na área da classificação de sentimento e emoção noutras línguas, como descrito no capítulo 3. Utilizando a transformação dos textos obtidos no vetor de TF-IDF, bem como as características adicionais processadas, conseguimos, representar os textos através de números, cobrindo o máximo de informação possível.

Os três modelos a desenvolver apresentam arquiteturas bastante distintas e de

<sup>8</sup>Disponível em <https://github.com/daviddias/METI-EADW/tree/master/src/sentimentAnalisis>

complexidade crescente, pelo que será abordado cada um destes detalhadamente. Os modelos Naive Bayes e SVM foram desenvolvidos usando a linguagem de programação *Python*, com recurso à biblioteca *scikit-learn*<sup>9</sup>. A rede neuronal LSTM foi desenvolvida com recurso à biblioteca *Keras*<sup>10</sup>. Para que estes modelos possam ser capazes de efetuar a classificação e previsão, é necessário fornecer-lhes dados onde eles possam reconhecer padrões. Para este efeito, o conjunto de dados composto pelo vetor TF-IDF e pelas características adicionais foram usadas para o treino dos modelos. Os dados foram divididos em duas partes: uma com 80% dos tweets e outra com 20% sendo a primeira parte usada para treino dos modelos. A segunda parte, independente da primeira, foi usada apenas para avaliar os modelos treinados. Devido a este método apresentar vantagens e desvantagens optou-se pela aplicação de outro método. Esta escolha teve como fundamento a elevada quantidade de dados, pois pode haver grandes discrepâncias nos valores obtidos usando o conjunto de dados para testar o modelo. Assim ficou definido que os modelos seriam treinados e validados com duas abordagens, sendo que a segunda foi efetuar o treino e teste com *10-fold cross validation*. Desta forma consegue-se obter dados mais precisos e com menos discrepâncias.

A classificação destes modelos irá ser para efetuar as duas tarefas distintas com base nos dados extraídos. A primeira consiste na classificação do texto numa das seis emoções básicas de Ekman, ou seja classificar um texto numa das seguintes emoções felicidade, tristeza, surpresa, medo, nojo ou irritação. A outra tarefa é a previsão do *emoji* mais adequado para um dado texto, sendo esta previsão muito mais complexa. Esta complexidade deve-se ao facto de ser necessário atribuir um *emoji* dentro de 62 definidos. Estes 62 *emojis* estão presentes na tabela 4.1, sendo que entre si existem variações gráficas muito próximas.

#### 4.5.1 Naive Bayes

Os modelos *Naive Bayes* são aplicados a tarefas de classificação de textos desde a década de 70. Mesmo com o aparecimento de outras arquiteturas, estes modelos computacionais ainda são bastante populares e utilizados para o mesmo tipo de tarefas na atualidade. Como abordado na secção 2.2 estes modelos são classificadores probabilísticos tendo como base o teorema de Bayes. Para o desenvolvimento é preciso estipular alguns parâmetros, no qual é necessário otimizar. Os parâmetros mais importantes são os seguintes.

- **Alpha:** Constante numérica denominada por *Laplace smoothing*. Valor que permite nos casos em que a probabilidade de ocorrência ser  $\simeq 0$  atribuir um valor numérico diferente de 0.
- **Fit prior:** Indicação se o modelo deve calcular a probabilidade de ocorrência das classes antes do processo de treino.

O valor mais crítico é o parâmetro de *alpha* pois terá influência no tratamento de casos raros. Devido ao elevado número de características e uma grande quantidade de textos, a ocorrência de certos termos ou mesmo de certas classes será relativamente rara. Devido a estas ocorrências e devido à natureza da formula da probabilidade condicionada, estes termos terão atribuído um valor muito perto de 0. O valor de *alpha* permite que estas ocorrências não tomem um valor tão baixo, e consequentemente termos ou características com grande probabilidade de ocorrência, terão valores ligeiramente inferiores. Desta forma

<sup>9</sup><https://scikit-learn.org/stable/index.html>

<sup>10</sup><https://keras.io>

evita-se ligeiramente a predominância de classes, sendo que o combate desta situação é bastante complexo e difícil.

A opção de *fit prior* indica ao modelo se este deve aprender a probabilidade de ocorrência das classes do conjunto de dados. Neste caso em concreto, o modelo tem em conta a probabilidade da ocorrência das diversas emoções no conjunto de dados que foram aplicados. Para ambos os parâmetros foram definidos os valores na tabela 4.7

Parâmetro	Escolha tomada
Alpha	0.98
Fit Prior	True

Tabela 4.7: Definição dos parâmetros para o modelo Naive Bayes

O valor de *alpha* ficou definido a 0.98, não muito distante do valor *default* que é de 1.0. Este valor foi obtido através de um processo de *grid search*. Também se verificou empiricamente que os resultados do modelo são melhores se as probabilidades das classes foram previamente conhecidas.

#### 4.5.2 Support Vector Machine

Quando comparadas com modelos baseados em Naive Bayes, as SVM têm diferenças de funcionamento que podem levar a resultados muito distintos. A classificação de um modelo SVM é feita através da representação dos dados de teste num espaço dimensional, onde é calculada uma fronteira de decisão de tal forma que esta tenha o maior espaçamento entre as diversas classes. Um dos principais problemas do desenvolvimento de modelos de aprendizagem é a escolha de parâmetros, de forma a que estes levem a melhores resultados. Devido a este problema, é necessário efetuar testes que indiquem qual a melhor combinação de parâmetros. Os principais parâmetros a otimizar para estes modelos são os seguintes.

- **Loss:** Corresponde à função de perda usada pelo classificador. A função de perda é um método de avaliação para o nosso modelo, no qual é responsável pela própria aprendizagem deste. Se as previsões obtidas se desviarem muito dos resultados reais, a função de perda obterá um número relativamente grande. Com a ajuda de alguma função de otimização, a função de perda aprende a reduzir o erro na previsão e assim melhorar o desempenho final. As funções fornecidas vão desde *hinge* até *log*, sendo constituídas por funções lineares e logarítmicas respetivamente.
- **Penalty:** O termo de regularização utilizado. Corresponde ao método de penalização, usado pelo modelo durante o processo de treino.
- **Alpha:** Constante numérica que é aplicada ao método de penalização. Mais conhecido como parâmetro de aprendizagem.
- **Epochs:** Número total de passagens pelos dados. Por cada passagem, o modelo aprende a efetuar melhor a separação das classes existentes nos dados.
- **Stop Criteria:** Indicação de paragem do modelo. Variável booleana que indica se o modelo deve finalizar o processo de aprendizagem, porque o valor de aprendizagem não está a aumentar.

Devido a todos estes fatores terem por si imensas opções, foi efetuado um processo de *grid search* de forma a obter a melhor combinação de valores. Como método de avaliação foi utilizado o resultado de *accuracy* obtido pelo modelo, quando aplicado o conjunto de teste. A melhor combinação de parâmetros obtida é apresentada na tabela 4.8.

Parâmetro	Escolha tomada
Loss	hinge
Penalty	L2
Alpha	$1 \times e^{-3}$
Epochs	1000
Stop Criteria	Yes

Tabela 4.8: Definição dos parâmetros para o modelo SVM

O método de *hinge* para a *loss*, é aquela que apresenta melhores resultados do conjunto de opções apresentadas. Ao usar o método *hinge*, o modelo SVM, consegue lidar mais adequadamente com problemas multi-classe bem como, processar melhor grandes conjuntos de dados. Também é a função de perda mais utilizada nos modelos SVM. O método de penalização utilizado foi o *L2*, que são os regularizadores *standard* de modelos SVM. Este método foi escolhido devido à condicionante do método de *loss* escolhido. Os restantes valores foram os melhores, dentro de um conjunto de valores estipulados anteriormente. Importante realçar que o *stop criteria* não veio proveniente do método de *grid search* mas sim, uma decisão tomada para diminuir o tempo de treino do modelo sem que isso tivesse uma implicação no seu desempenho

### 4.5.3 Long Short Term Memory

As redes neuronais Long Short Term Memory apresentam imensos parâmetros por definir. Em primeira instância, para qualquer tipo de redes neuronais, é preciso definir o número de camadas, funções de ativação e número de neurónios. As redes LSTM não são diferentes sendo que é preciso definir imensos parâmetros, de forma a construir um modelo eficaz e capaz de efetuar classificações o mais preciso possível. Como mencionado anteriormente as redes LSTM, foram desenvolvidas com recurso à biblioteca *Keras*, onde é preciso construir as diversas camadas que constituem a rede neuronal.

Para a construção de uma rede LSTM é necessário aprofundar um pouco mais, os conhecimentos obtidos no capítulo 2.2. A construção de uma rede LSTM através do *Keras* é efetuada através de um construtor fornecido pela biblioteca de redes neuronais sequencial, com várias camadas onde existe pelo menos uma camada LSTM. Para entender o processo de treino e classificação da rede é preciso especificar cada uma delas pois, cada uma tem uma tarefa específica e essencial. Estas camadas são as seguintes.

- **Sequential:** Não representa camada mas sim o método de construção da rede. Este método, permite que seja construída uma rede com camadas sequenciais, onde o *output* proveniente duma camada será o *input* da camada seguinte. Assim obtemos uma maior flexibilidade em termos de arquitetura da rede e conseguimos adaptá-la ao problema em questão.
- **Embedding Layer:** Camada dedicada para o tratamento de texto em redes neuronais. Aceita como parâmetros os dados textuais representados de forma numérica, como por exemplo o *Term Frequency*, e transforma cada palavra em

números reais únicos, com relação às restantes palavras do conjunto de dados. Desta forma podemos obter um vetor de várias dimensões com as relações entre estas palavras. Este vetor pode ser guardado e ser aplicado noutros modelos, ou pode ser utilizado no processo de treino. Caso tenhamos *word embeddings* resultantes de outros modelos, podem ser utilizadas no modelo, simplificando o processo. É preciso especificar 3 valores para esta camada.

- **Dimensão de *input*:** O número de termos a serem aplicados à camada.
  - **Dimensão de *output*:** O tamanho do vetor resultante do processo efetuado pela camada, para os termos introduzidos.
  - **Tamanho do *input*:** A quantidade de dados aplicados à camada.
- **Spatial Dropout:** Camada responsável por efetuar o processo de *dropout*. O *dropout* é característico das redes LSTM pois é aqui que certos *inputs* são passados a 0, ou seja, esquecidos, para tornar a rede mais eficiente.
  - **LSTM Layer:** Camada constituída por células LSTM, onde é efetuado o processamento dos dados provenientes da camada anterior. Tem muitos parâmetros para otimizar, sendo os principais:
    - **Unidades:** Número inteiro indicando a dimensionalidade do vetor obtido pelo processamento da camada.
    - **Função de ativação:** Função utilizada para aplicar aos dados de entrada da camada. É costume utilizarem-se funções de tangente hiperbólica (*tanh*), mas também podem ser aplicadas funções lineares ou sigmoide.
    - **Dropout:** Valor correspondente à percentagem de dados que passarão pelo processo de *dropout* durante a entrada dos dados na camada.
    - **Recurrent Dropout:** Valor correspondente à percentagem de dados a passar pelo processo de *dropout* mas durante o processo de passagem temporal.
  - **Dense Layer:** Camada típica de rede neuronais. Recebe valores de entrada onde efetua várias transformações através da multiplicação dos seus pesos e de funções específicas. Tem os seguintes parâmetros a serem especificados:
    - **Unidades:** Número inteiro indicando a dimensionalidade do vetor obtido pelo processamento da camada. Devido a ser a última camada, este número será igual ao número de classes do problema.
    - **Função de ativação:** Função utilizada para aplicar aos dados de entrada da camada.

Para além da definição das camadas da rede, é necessário ainda definir alguns parâmetros gerais, responsáveis pelo desempenho em geral da rede. Os parâmetros a otimizar são os seguintes:

- **Loss:** Função objetivo que o modelo irá tentar minimizar.
- **Optimizador:** Escolha do otimizador a aplicar durante o processo de dados.
- **Epochs:** Número total de passagens pelos dados. Por cada passagem o modelo aprende a efetuar melhor a separação das classes existentes nos dados.

Uma das principais razões para as redes neurais terem obtido resultados muito significativos, é em grande parte dedicado à sua flexibilidade em construir arquiteturas específicas para o problema em questão. A arquitetura da rede neuronal desenvolvida foi baseada nos trabalhos já abordados na seção 3, bem como em trabalhos existentes para a classificação de texto usando redes LSTM como Wang, Huang e Zhao, 2016. A combinação das diversas camadas, bem como a utilização de outras funções de ativação, poderão sempre levar a outro tipo de resultados. Devido à infinidade de combinações possíveis para a criação da rede neuronal LSTM decidiu-se optar pela arquitetura definida e representada na figura 4.3.

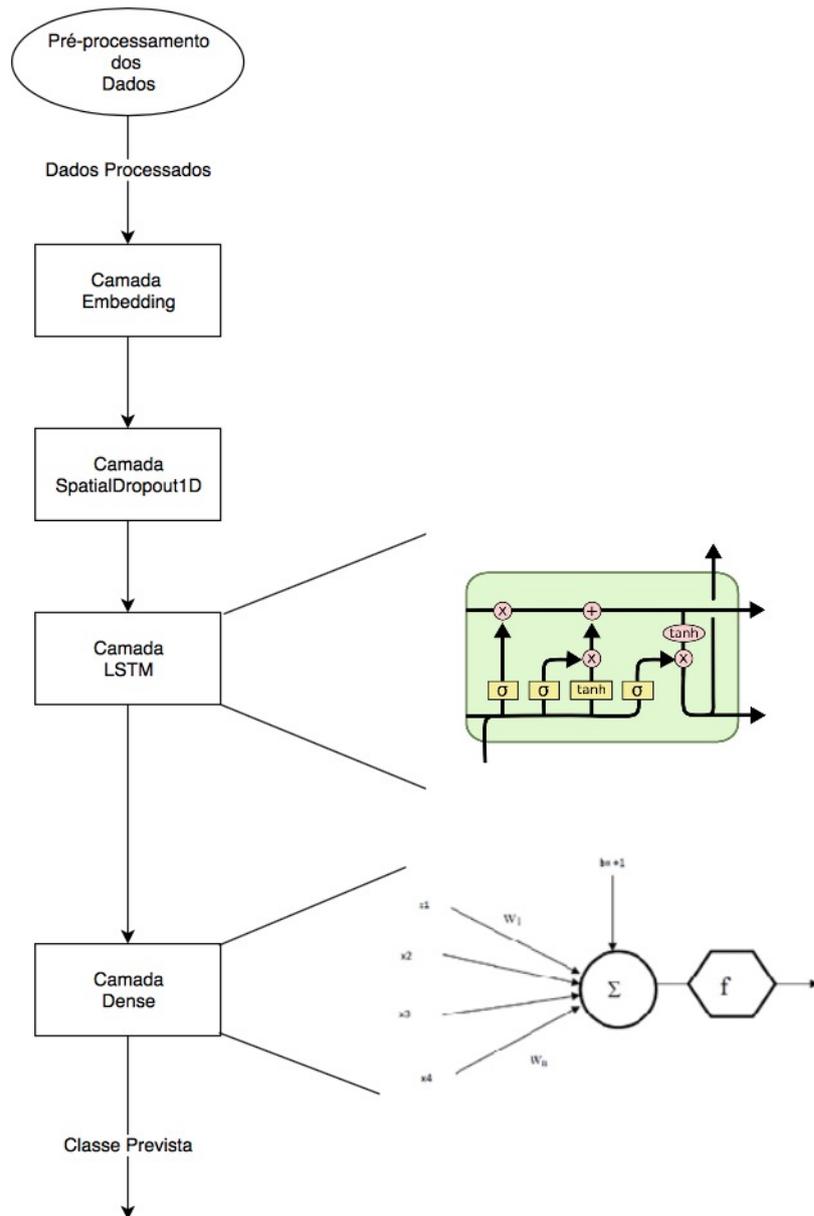


Figura 4.3: Arquitetura da rede LSTM desenvolvida

Um ponto importante de referir é que o resultado obtido da rede neuronal não é uma classe em concreto, mas sim a probabilidade de pertencer à dita classe. De forma a aplicar as métricas definidas para a avaliação dos modelos, considera-se que a classe atribuída

será sempre aquela com a maior probabilidade. Tal como os restantes modelos, de forma a obter os melhores resultados foi necessário otimizar e procurar, neste caso de forma exaustiva, a melhor combinação de parâmetros. Optou-se pelo método de *grid search* para obter tais combinações, no qual resultou os valores presentes na tabela 4.9.

Camada	Parametro	Escolha tomada
Embedding	Dimensão de input	150000
	Dimensão de output	100
	Tamanho de input	Tamanho dos dados
SpatialDropout1D	Percentagem de dropout	20%
LSTM	Unidades	100
	Função de ativação	Tanh (Tangente hiperbólica)
	Percentagem de dropout	10%
	Dropout recorrente	15%
Dense	Unidades	6 ou 62
	Função de ativação	Lin (Linear)

Tabela 4.9: Definição dos parâmetros para a rede LSTM

Do resultado obtido pelo *grid search* é necessário clarificar alguns valores obtidos. Em primeiro lugar, o tamanho do *input* da camada de *embedding* e o valor das unidades da camada *dense* dizem respeito à tarefa de classificação. Este valor é relativo ao número total de dados a serem aplicados à rede e a quantidade de classes em que estes dados irão efetuar a classificação, respetivamente. Os valores de dimensão de *output* e unidades da camada LSTM terão sempre de serem iguais, devido à saída da camada de *embedding* ter de ser da mesma dimensão da entrada da camada LSTM.

## 4.6 Disposição de emojis

Para além da criação destes três modelos, foi criado um modelo adicional dedicado a um dos objetivos principais deste trabalho, os *emojis*. De modo a facilitar a visualização dos emojis de acordo com o contexto da sua utilização, e assim identificar relações entre os variados emojis estudados e também entre emojis e texto, decidimos explorar modelos de similaridade semântica. Uma das respostas que se pretende obter é a disposição dos emojis sugere efetivamente um agrupamento por emoções. Mais concretamente, para o caso do Twitter português, essa disposição confirma os agrupamentos sugeridos pelo trabalho de Wood e Ruder, 2016.

Tendo em conta que os emojis são representados através de sequências de caracteres *unicode*, para atingir este objetivo optou-se pela criação de um modelo *word2vec* usando o conjunto de dados. Este modelo foi construído com recurso à biblioteca *Gensim*<sup>11</sup>, depois de um pré-processamento adicional aos dados já transformados até esta fase. Para aplicar o conjunto de dados ao treino do modelo *word2vec* é necessário transformar cada texto

<sup>11</sup><https://radimrehurek.com/gensim/>

numa lista de termos. O resultado final é uma lista composta por várias listas, onde cada uma corresponde aos termos de cada texto.

Tal como nos modelos anteriores, a criação de um modelo *word2vec* também necessita a definição de vários parâmetros. No entanto, ao contrário dos restantes modelos, estes parâmetros não necessitam de ser otimizados, pois dizem respeito às características dos dados introduzidos. Alguns destes parâmetros também dizem respeito ao processo de treino e como queremos que este se efetue. Segue-se então a definição destes parâmetros:

- **Contagem Mínima:** Número inteiro que indica a frequência mínima de um termo para ser incluído no modelo. Por outras palavras, todos os termos que ocorrem menos vezes do que este número são ignorados.
- **Janela:** Número inteiro que indica o intervalo de palavras relacionadas com o termo em questão. Se o modelo estiver a processar o termo  $i$  com tamanho de janela  $n$ , o modelo irá relacionar as  $n$  palavras anteriores e as  $n$  posteriores ao termo como relacionadas.
- **Tamanho:** Número que indica a dimensão do vetor de *output*, ou seja o tamanho do vetor resultante do modelo.
- **Epochs:** Número total de passagens pelos dados. Por cada passagem o modelo aprende a efetuar melhor a separação das classes existentes nos dados.
- **Workers:** Número de *threads* a usar para o treino do modelo e tornar o processo de treino mais rápido e eficiente.
- **Alpha:** Constante numérica que é aplicada ao método de penalização. Mais conhecido como parâmetro de aprendizagem.

Todos estes parâmetros tiveram de ser escolhidos com base no conhecimento dos dados, pois um método de *grid search* não é o mais indicado para obter os melhores valores para este tipo de modelo. As métricas usadas são utilizadas em modelos para tarefas de classificação, no qual o modelo *word2vec* não realiza. Definiram-se então os valores presentes na tabela 4.10 como parâmetros do *word2vec*.

Parâmetro	Valor
Contagem Mínima	20
Janela	5
Tamanho	300
Epochs	700
Workers	8
Alpha	0.03

Tabela 4.10: Definição dos parâmetros para o modelo *word2vec*

Dos valores definidos, aquele que pode influenciar mais os resultados finais é o parâmetro da janela. Este valor foi definido com base nos dados estatísticos efetuados ao nosso conjunto de dados. Uma característica adicional que é extraída dos textos é o número total de palavras. Desta característica foi obtida uma média de  $\approx 9$  palavras por *tweet* de onde se definiu o valor do parâmetro de janela para  $n = 5$ . Com este valor é possível abranger o texto em quase toda a sua totalidade independentemente da posição do termo a analisar.

A contagem mínima ficou definida com um valor de 20 de forma a não incluir termos que são apenas erros ortográficos. Os restantes valores foram obtidos através de variações com os valores *standart*. Os parâmetros de tamanho e *epochs* são difíceis de estimar à primeira vista, no qual foram obtidos através de tentativa/erro. O valor de tamanho utilizado foi o de *standart*.

Com o modelo resultante após o processo de treino, conseguimos representar cada termo do conjunto de dados, incluindo *emojis*, através de um vetor numérico. É este vetor que nos vai permitir representar os emojis no espaço, para além de permitir calcular a similaridade entre os diversos termos presentes no conjunto de dados, com base no coseno dos seus vetores. Os resultados obtidos bem como a sua discussão estão presentes no capítulo seguinte.

Esta página foi intencionalmente deixada em branco.

## Capítulo 5

# Resultados Finais

Este capítulo aborda os resultados obtidos das diversas experiências efetuados, aos três modelos desenvolvidos. Para além dos resultados dos modelos de classificação, também são apresentados as experiências com o modelo *word2vec* e usando uma abordagem simples com recurso a léxicos para efetuar a classificação da emoção num texto. Na secção 2.2 foi apresentado um conjunto de métricas utilizadas para avaliar classificadores, e que serão aplicadas para medir o desempenho dos modelos desenvolvidos. Para além destas métricas é importante realçar que muitos dos valores obtidos, são o reflexo do trabalho descrito na secção 4.1. Com o trabalho desenvolvido e com algumas das decisões tomadas sobre a criação do conjunto de dados, verificamos que estas afetaram de forma significativa o resultado final.

De forma a avaliar os modelos desenvolvidos foi efetuado o processo de treino usando 80% do conjunto de dados obtido. Os restantes 20% do conjunto de dados foram utilizados para a avaliação dos modelos, usando as métricas definidas. Para além desta abordagem, também é efetuado um processo de *10 cross validation* ao conjunto de dados de forma a obter resultados mais precisos.

### 5.1 Abordagem Baseline

De forma a verificar se os modelos desenvolvidos são uma solução viável para o problema da classificação de emoções de textos em português, e que demonstram ganhos relativamente a métodos mais simples, é importante compará-los com um método simples e fácil de aplicar, normalmente chamado de *baseline*. A nossa abordagem *baseline* consiste na utilização do léxico NRC Emotion<sup>1</sup> (Mohammad e Turney, 2013), onde cada palavra está categorizada numa das seis emoções básicas de Ekman. Desta forma verifica-se qual a emoção que ocorre com mais frequência num texto e compara-se com a emoção atribuída no conjunto de dados. No caso de existir empates a emoção atribuída é de “neutro” pois não existe nenhuma emoção predominante.

Ao efetuar esta abordagem aos textos presentes no conjunto de dados, obteve-se os seguintes resultados da tabela 5.1.

Verificamos que usando esta abordagem com recurso ao léxico NRC os resultados obtidos são muito baixos e alguns próximos de 0%, um indicativo de que limitarmo-nos a usar léxicos para o reconhecimento de emoções em *tweets* está longe de ser suficiente,

---

<sup>1</sup>Disponível em <http://sentiment.nrc.ca/lexicons-for-research>

<b>Emoção</b>	<b>Accuracy</b>	<b>Precision</b>	<b>Recall</b>	<b>F1</b>
Felicidade	2.0%	1.1%	1.3%	0.019
Raiva	0.0%	0.0%	0.0%	0.000
Nojo	0.1%	0.1%	0.1%	0.001
Medo	0.0%	0.0%	0.0%	0.000
Tristeza	0.7%	0.5%	0.4%	0.004
Surpresa	0.0%	0.0%	0.0%	0.000
<b>Total</b>	<b>1%</b>	<b>0.3%</b>	<b>0.3%</b>	<b>0.007</b>

Tabela 5.1: Resultados obtidos para a tarefa de classificação de emoção usando o léxico NRC

e será sempre necessário recorrer a uma abordagem mais complexa. Isto deve-se a vários motivos, sendo o principal o tipo de escrita que está presente nos textos que constituem o conjunto de dados. Como a escrita das redes sociais é bastante informal, contendo muitas abreviações, uso de calão e erros ortográficos, grande parte das palavras presentes nos textos não se encontram no léxico NRC. Isto por sua vez, resulta em muitos textos sem uma emoção atribuída. Para além deste ponto, também existem casos em que apenas um termo é que está presente no léxico, o que leva a que todo o texto seja classificado com a emoção respetiva a esse termo. A não contabilização de características como os *emojis* também torna o processo mais complicado pois não acrescentam nenhum tipo de informação.

Outra razão para estes maus resultados deve-se à construção do léxico NRC, mais concretamente às emoções atribuídas a cada palavra, pois dizem respeito à sua origem inglesa. O léxico NRC contém palavras em diversos idiomas, no qual o português se encontra, mas resultam da tradução direta do conteúdo do léxico de língua inglesa. Devido a este aspeto a emoção atribuída não tem em conta o contexto cultural do idioma para qual foram traduzidas, atribuindo assim uma emoção que não é a mais adequada. Exemplos deste tipo de ocorrências é a palavra “fogo” à qual foi atribuída a emoção de medo, mas em Barbieri et al. (2018b) esse termo em castelhano e o seu respetivo *emoji* são muito associados a paixão sendo classificado com uma emoção distinta.

Com esta abordagem demonstramos a dificuldade que é a classificação de emoção em textos, e que mesmo utilizando recursos especificamente desenvolvidos para estes casos muitas das vezes são insuficientes. Nesta abordagem não é produzido nenhuma melhoria, sendo que uma escolha aleatória da emoção de cada texto produziria, em teoria, melhores resultados. Devido a esta ineficiência é necessário produzir métodos que consigam analisar o texto como um todo, e não termo a termo como esta abordagem. Estes resultados sugerem que será necessário desenvolver modelos computacionais mais avançados, que foi precisamente o próximo passo.

## 5.2 Naive Bayes

O modelo Naive Bayes, de um ponto de expectativas iniciais será aquele que apresentará piores resultados. Não porque o modelo é inferior aos restantes, mas sim porque efetua a classificação com base na probabilidade de ocorrência de cada termo/texto. Juntamente com este ponto, vem também o tratamento de casos raros. Em diversos trabalhos usando modelos Naive Bayes estes casos são tendencialmente classificados como pertencentes à classe maioritária. Este tipo de ocorrência é algo não desejado, dando por exemplo a

deteção de casos de cancro. Devido à existência de poucos dados de casos de cancro, um classificador treinado com estes dados, ao ser apresentado com um novo caso terá tendência a classificar como pertencente à classe negativa. Assim, o modelo está a classificar uma pessoa como saudável quando na realidade ela não está. No caso deste trabalho a existência de casos raros será de cinco classes, pois nos dados extraídos existe a predominância da classe de felicidade. Apesar de não ter a gravidade de classificar um falso positivo no processo de classificação de cancro, continua a ser um problema não desejado.

A primeira experiência efetuada pelo modelo Naive Bayes, foi precisamente o cálculo das métricas apresentadas na secção 2.2, após o treino do modelo com a parte do conjunto de dados destinado para esse propósito. De forma a validar ao máximo os resultados obtidos, também foi aplicado o processo de *10-fold cross validation* onde foram calculados os valores médios de *accuracy* juntamente com o seu desvio padrão. Como abordado na secção 4.5, ao aplicar o processo de *grid search*, houve parâmetros, onde foi necessário testar manualmente, nomeadamente a escolha dos n-gramas a aplicar ao vetor TF-IDF. É importante efetuar esta experiência pois pode-nos ajudar a localizar o teor emocional dos textos. Ou seja, se o processo de classificação da emoção do textos é mais fácil fornecendo termos, expressões ou uma combinação de ambos. Por isso foram efetuados testes, correndo o modelo tendo como dados de treino o vetor TF-IDF com diferentes n-gramas, e que depois é avaliado seguindo as métricas definidas com os dados de teste.

Também se verificou através do processo de *grid search* que as características adicionais extraídas (descrito na secção 4.4) introduzem alguma confusão no modelo. Isto deve-se em grande parte a muitas destas características terem um valor 0 atribuído pois as palavras presentes nos textos não se encontram nos léxicos. Este problema já foi verificado na abordagem *baseline* devido à informalidade dos textos presentes no conjunto de dados, e devido a textos de diferentes emoções terem estas características com o mesmo valor introduz aos modelos maior confusão. Desta forma optou-se por usar apenas os vetores TF-IDF resultantes sem estas características extraídas.

É importante clarificar que nesta experiência foi realizada uma classificação e uma previsão. A classificação consiste em atribuir aos textos introduzidos no modelo, uma das seis emoções de Ekman (1999). A previsão consiste em prever a utilização de um dos 62 *emojis* definidos no trabalho de Wood e Ruder (2016). Os resultados obtidos destas experiências estão dispostos nas tabelas 5.2 e 5.3 respetivamente.

N-grams	10 Cross Validation	Treino e Teste			
	Accuracy	Precision	Recall	Specificity	F1
1	73.2%±2.1%	70.1%	74.3%	59.8%	0.706
2	73.1%±2.5%	74.3%	74.1%	54.5%	0.696
3	70.4%±1.2%	74.6%	72.0%	43.2%	0.652
<b>1-2</b>	<b>74.5%±2.6%</b>	<b>76.2%</b>	<b>75.7%</b>	<b>59.5%</b>	<b>0.708</b>
2-3	73.2%±2.4%	75.8%	75.2%	54.6%	0.706
1-3	74.0%±2.0%	77.3%	75.6%	53.8%	0.699

Tabela 5.2: Resultados obtidos para a tarefa de classificação de emoção do modelo Naive Bayes

Os valores obtidos para a tarefa de classificação de emoção são satisfatórios. Também demonstra que o valor de n-gramas tem alguma influência nos resultados finais do modelo, sendo que uma combinação de unigramas e bigramas aparenta ser a melhor solução para esta tarefa. Apesar de os valores de F1, métrica principal de comparação pois é um equilíbrio entre as métricas de *precision* e *recall*, serem bastante semelhantes entre si,

há alguns resultados curiosos. Uma utilização de trigramas mostra sempre ser inferior aos restantes n-gramas, logo podemos presumir que expressões longas não acrescentam o mesmo valor emocional que expressões curtas ou termos individuais.

Ao aprofundar este caso analisando os textos existentes no conjunto de dados é possível verificar esta realidade. Já é conhecido pela tabela 4.5, um conjunto de termos singulares altamente relacionados com as seis emoções em específico. Além destes termos, também é possível verificar expressões relativamente comuns no conjunto de dados, com essa mesma relação. Exemplos destas expressões, são os casos de “*Gosto muito*”, “*Odeio isto*” e “*Amo imenso*” que estão muito associadas às emoções de felicidade e de raiva. A existência de termos semelhantes, mas com trigramas já não é muito recorrente no conjunto de dados, o que reforça os resultados da tabela 5.2.

Analisando os restantes valores, verificamos que a *accuracy* do método de *10-cross validation* tem, para todos os casos, valores muito próximos. Apesar disso, a combinação de unigramas e bigramas também se mostrou ser a melhor opção, obtendo os valores de *accuracy* mais altos. Por outro lado, também apresenta os valores de variância mais elevados no qual se torna difícil classificar como o melhor valor para aplicar ao vetor TF-IDF. Outro valor menos positivo é o valor de *specificity*, à volta de 50%, um indicador que é preciso efetuar uma análise mais aprofundada dos resultados obtidos. De seguida apresenta-se os resultados para a tarefa de previsão de *emojis*.

N-grams	10 Cross Validation	Treino e Teste			
	Accuracy	Precision	Recall	Specificity	F1
1	26.0%±2.0%	30.3%	27.7%	90.3%	0.195
2	23.9%±2.4%	49.8%	27.2%	89.1%	0.205
3	18.9%±2.2%	57.1%	24.0%	87.7%	0.193
1-2	25.9%±2.0%	47.3%	27.6%	89.2%	0.189
<b>2-3</b>	<b>24.1%±2.4%</b>	<b>50.1%</b>	<b>29.3%</b>	<b>90.8%</b>	<b>0.237</b>
1-3	25.8%±2.3%	53.7%	28.9%	89.3%	0.206

Tabela 5.3: Resultados obtidos para a tarefa de previsão de *emojis* do modelo Naive Bayes

Verificamos que os resultados da tarefa de previsão de *emoji* foram bastante mais baixos. Apesar de serem inferiores, é algo esperado devido à dificuldade da tarefa em questão. Apesar de tudo, os valores de *accuracy* encontram-se bem acima de uma escolha aleatória, onde com seis classes, teria o valor de 1.61% e não estão muito abaixo dos resultados publicados para a tarefa de *emoji prediction* (Barbieri et al., 2018b) realizada recentemente para o inglês e castelhano. No entanto, estes resultados não se podem considerar bons, a começar logo pelo elevado valor de *specificity* que indica, à partida, uma má classificação no geral. Os valores de *precision* e *recall* suportam novamente essa teoria, sendo que a métrica com que se avalia o modelo, o F1, tem em média um valor de 20%. Também verifica-se que a utilização de bigramas e trigramas é a melhor escolha, ao contrário da tarefa de classificação de emoção. Este valor indica que grande parte da utilização de *emojis* está relacionado com expressões e não com termos singulares. Exemplos desta ocorrência encontram-se em expressões como aquelas apresentadas na figura 5.1.

Uma das principais razões para estes resultados, deve-se ao número de classes no qual temos de prever o *emoji*. O facto de ter de optar por um de 62 *emojis*, adiciona uma elevada complexidade ao problema que não é fácil de resolver. Adicionando a este ponto que a utilização destes *emojis* pode ser arbitrária, verifica-se que é muito difícil a previsão

Adoro muito esse cabelo @URL 🥰🥰🥰  
 Adoro muito esse lugar querida @USERNAME 😊  
 Muito linda @USERNAME ❤️❤️❤️❤️❤️  
 vc está muito linda @USERNAME ❤️❤️❤️

Figura 5.1: Exemplos de expressões que usem o mesmo *emoji*

de *emojis*

Devido ao valores de *specificity* mais elevados que o esperado para a tarefa de classificação de emoção procedeu-se a uma análise mais aprofundada, relativamente ao resultados obtidos por classe. Como abordado na secção 4.2, o nosso conjunto de dados tem uma maior quantidade de dados relativos à classe de felicidade do que às restantes. Este não balanceamento de dados pode levar a uma classificação bastante boa para uma classe e má para as restantes. De maneira a verificar se este problema está a decorrer com os nossos modelos, foi efetuado uma análise ao melhor modelo obtido e feita uma avaliação com base nas métricas estabelecidas. Os resultados desta experiência encontram-se na tabela 5.4.

Emoção	Precision	Recall	Specificity	F1
Felicidade	77%	96%	41%	0.866
Raiva	82%	12%	63%	0.153
Nojo	93%	5%	68%	0.078
Medo	98%	2%	67%	0.019
Tristeza	64%	47%	51%	0.527
Surpresa	87%	3%	67%	0.021
<b>Total</b>	<b>76%</b>	<b>75%</b>	<b>59%</b>	<b>0.708</b>

Tabela 5.4: Resultados obtidos para a classificação de emoção do melhor modelo Naive Bayes

Analisando os resultados obtidos, verificamos que os elevados valores de F1 se devem em grande parte ao não balanceamento dos dados. Mais propriamente, tal como descrito na secção 4.2, a maior parte dos tweets pertence à classe de felicidade. Como esta classe constitui quase 70% do nosso conjunto de dados, o modelo consegue classificar corretamente essa mesma percentagem de forma correta. De forma a obter mais conclusões vamos recorrer à matriz de confusão obtida pelo modelo. A matriz de confusão consiste numa visualização da *recall* entre as várias classes do modelo. Uma matriz de confusão boa é aquela cuja diagonal tem valores muito próximos de 100% e as restantes posições perto de 0%. A matriz de confusão para este caso encontra-se na figura 5.2.

Analisando a matriz de confusão, vemos os valores estão todos concentrados na coluna correspondente à emoção de felicidade, ou seja, longe do pretendido. Para além da classe de felicidade, vemos que a classe correspondente à tristeza, a segunda com mais instâncias, tem um valor de F1 próximo dos 50%. As restantes classes, que têm uma proporção baixa de instâncias, são considerados como casos raros e muito pouco prováveis de acontecer. Apesar de negativo tendo em conta o objetivo principal do trabalho, este resultado abre portas a uma discussão já retratada na área da aprendizagem computacional. A discussão destes resultados, bem como as conclusões significativas serão abordadas mais à frente no capítulo 6.

Para a tarefa de previsão de *emoji* os resultados por classe obtidos são difíceis de apresentar. Esta dificuldade deve-se ao facto de termos 62 classes, o que ocupa uma dimensão considerável. Sendo assim, na tabela 5.5 estão apresentados os *emojis* com os

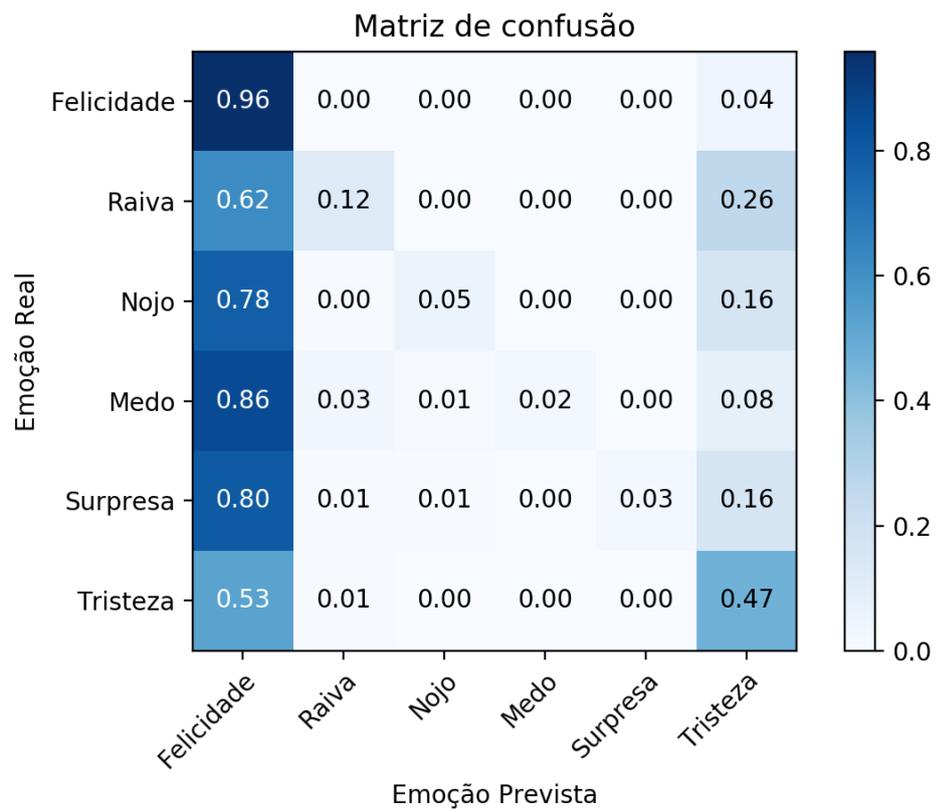


Figura 5.2: Matriz de confusão obtida para a classificação de emoção do modelo Naive Bayes

melhores valores de F1.

Emoji	Proporção	Emoção Associada	Precision	Recall	Specificity	F1
😄	1%	Felicidade	31%	52%	84%	0.392
😞	2%	Raiva	24%	73%	68%	0.364
😏	1%	Raiva	28%	67%	70%	0.407
😁	3%	Felicidade	47%	18%	99%	0.266
😃	1%	Felicidade	26%	18%	98%	0.216
😮	1%	Surpresa	96%	9%	99%	0.171
😱	0.8%	Medo	96%	6%	99%	0.128

Tabela 5.5: Melhores resultados obtidos da previsão de *emojis* do modelo Naive Bayes

Verificando os valores mais elevados de F1 vemos que estes não são muito altos. De referir que muitos dos *emojis* não presentes na tabela apresentam valores de  $F1 \approx 0$ . Para além destes valores verificamos que a emoção à qual foi atribuída pertence às classes maioritárias, nomeadamente felicidade e tristeza. Outro ponto curioso é o facto de nenhum destes *emojis* fazer parte dos mais comuns ao longo do conjunto de dados. Este facto reforça que *emojis* mais comuns são utilizados em imensos contextos e assim mais difícil de serem previstos. De forma a verificar o desempenho para estes *emojis*, na tabela 5.6 estão apresentados os resultados obtidos mas para os mais utilizados, por emoção.

Emoji	Proporção	Emoção Associada	Precision	Recall	Specificity	F1
😍	17%	Felicidade	98%	3%	99%	0.061
😡	1%	Raiva	100%	0%	100%	0.000
😭	4%	Tristeza	100%	0%	100%	0.000
😬	2%	Surpresa	100%	0%	100%	0.000
😨	1%	Medo	98%	6%	98%	0.123
🤢	1%	Nojo	83%	1%	98%	0.012

Tabela 5.6: Resultados obtidos da previsão dos *emojis* mais usados do modelo Naive Bayes

Verificamos que quanto maior a utilização, mais difícil se torna a sua previsão. Desta forma, adiciona-se mais força à teoria que a previsão de *emojis* é mais difícil, pois são utilizados em diversos contextos. O caso do *emoji* pertencente à classe de medo corresponde, dentro deste grupo, aquele com menos utilização. Também por sua vez verifica-se que é este o que obtém melhores resultados. Toda esta discussão é aprofundada no capítulo 6.

Devido ao problema da classe de felicidade ter muitos mais exemplos que as restantes, optou-se por efetuar mais experiências. A primeira foi efetuar um balanceamento dos dados. O processo de balanceamento de dados é por si bastante controverso no âmbito da aprendizagem computacional, como discutido em Kitchenham (1998), onde é abordado os prós e os contras deste método. Para além da discussão entre efetuar ou não o balanceamento, existe ainda a seleção do melhor método a aplicar de forma a igualar o número de classes. De uma forma resumida, o balanceamento de dados pode ser efetuado com dois métodos, o *undersampling* e o *oversampling* (Chawla et al., 2002; Kubat e Matwin, 1997). O *undersampling* consiste na forma mais intuitiva de efetuar o balanceamento, pois consiste em retirar dados da classe maioritária até esta ter o mesmo

tamanho que a classe minoritária. A principal desvantagem é a perda de informação (Liu, Wu e Zhou, 2008). No caso deste trabalho, efetuar o método de *undersampling* ao conjunto de dados corresponde a retirar elementos de todas as classes até ficarem equivalentes à classe de nojo. O método de *oversampling* implica a replicação dos dados da classe minoritária até esta ter a mesma quantidade que a classe majoritária. No caso deste trabalho implicaria replicar os dados pertencentes de todas as classes com quantidade inferior à classe de felicidade, até estas atingirem o mesmo tamanho. Este método por si apresenta alguns contras como abordado em Kubat e Matwin (1997).

Analisando cada um destes métodos optou-se por efetuar *undersampling* ao conjunto de dados. Isto vai resultar numa perda significativa de informação relativa às classes majoritárias, nomeadamente felicidade e tristeza. Apesar de artificial, é uma situação a experimentar para analisar o impacto do balanceamento. Após obter um conjunto de dados balanceado, onde todas as classes têm o mesmo número de exemplos, realizou-se novamente ao processo de classificação de emoções. Os resultados obtidos encontram-se na tabela 5.7.

Emoção	Precision	Recall	Specificity	F1
Felicidade	88%	44%	90%	0.591
Raiva	26%	45%	91%	0.337
Nojo	24%	27%	89%	0.264
Medo	3%	61%	87%	0.052
Tristeza	42%	47%	86%	0.448
Surpresa	4%	30%	89%	0.071
<b>Total</b>	<b>67%</b>	<b>43%</b>	<b>89%</b>	<b>0.507</b>

Tabela 5.7: Resultados obtidos com dados balanceados para a classificação de emoção do modelo Naive Bayes

Verificamos que o balanceamento de dados reduz o desempenho do modelo das classes de felicidade e tristeza. Por outro lado, verificamos valores mais elevados de F1 para as classes de raiva e nojo. As classes de medo e surpresa continuam com valores de F1 baixos, mas superiores aos resultados obtidos sem o balanceamento dos dados. Isto demonstra que o modelo não tem a tendência da experiência anterior de classificar grande parte dos dados como pertencentes à classe de felicidade. De forma a analisar com mais detalhe o processo de classificação do modelo, vamos analisar a matriz de confusão visível na figura 5.3.

Esta já se aproxima mais do pretendido. Verificamos que a classe de felicidade continua a ser aquela a que o classificador tem mais tendência a atribuir quando é presente a novos dados. No entanto, verificamos que os restantes valores da diagonal estão muito mais elevados. Outro aspeto interessante é que as classes com piores resultados, medo e surpresa, são mais vezes confundidas com as restantes classes. Analisando o modelo obtido com o balanceamento de dados, verificamos que este se aproxima mais do pretendido para a classificação de emoção. No entanto, continua a apresentar valores relativamente baixos, que precisam de ser mais trabalhados. Vamos então perceber se os restantes modelos têm desempenhos semelhantes ou não.

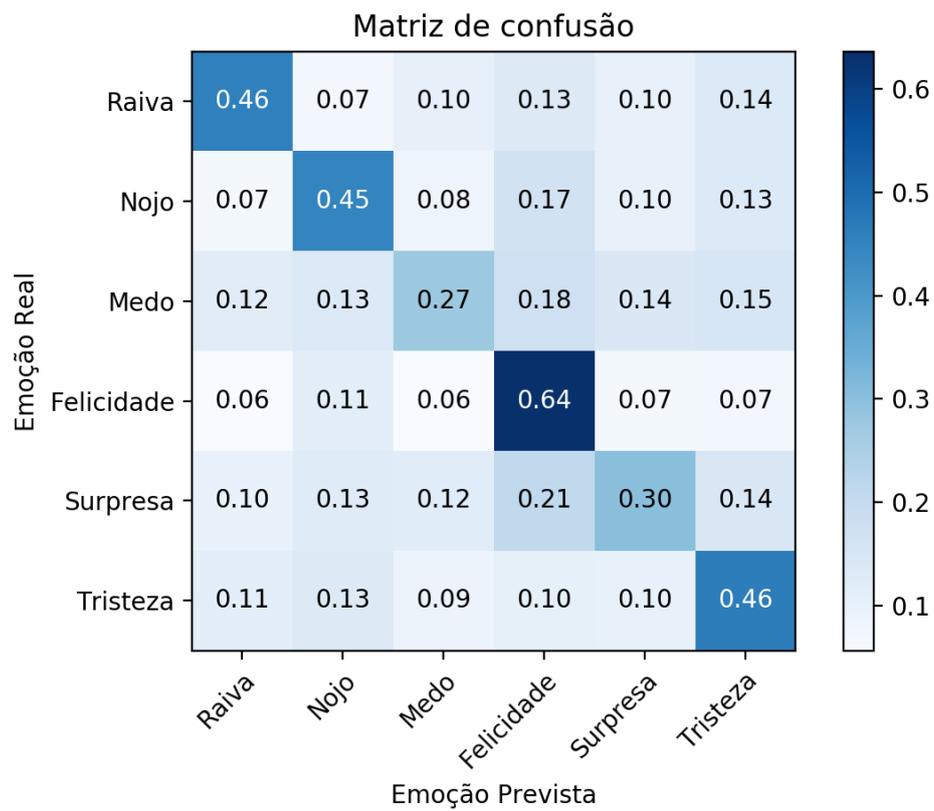


Figura 5.3: Matriz de confusão para a classificação de emoção com dados balanceados do modelo Naive Bayes

### 5.3 SVM

O modelo SVM é muito utilizado para diversos problemas sendo a classificação de textos, um deles. Ao contrário do modelo Naive Bayes, o funcionamento deste modelo é totalmente distinto. O desenvolvimento e avaliação deste modelo permite perceber se a classificação de emoções é possível através da localização espacial dos seus termos/textos. Também diferente do Naive Bayes, com o modelo SVM conseguimos trabalhar com o termo em concreto em vez da sua probabilidade de ocorrência. Apesar de parecer uma diferença pouco significativa, diz imenso sobre os dados que estamos a analisar. Para além destas diferenças, os resultados obtidos pelo modelo SVM permitem-nos verificar se é possível agrupar os textos com base na emoção e no *emoji* atribuído, durante a fase de extração de dados. Para verificar estas hipóteses, tal como no modelo Naive Bayes, precisamos de verificar o intervalo de n-gramas mais adequado para o vetor TF-IDF. Apesar de verificarmos que a combinação usando unigramas e bigramas é a que obtém melhores resultados para o modelo anterior, é necessário efetuar novamente a experiência pois são modelos com arquiteturas distintas.

Para este efeito foi realizada a mesma experiência com os n-gramas para as tarefas de classificação de emoção e previsão *emojis*, para o conjunto de dados de teste. Os resultados para classificação de emoção estão presentes na tabela 5.8.

N-grams	10 Cross Validation	Treino e Teste			
	Accuracy	Precision	Recall	Specificity	F1
1	70.4%±1.4%	64.2%	71.7%	40.3%	0.625
2	68.7%±2.3%	66.1%	69.3%	33.8%	0.576
3	68.4%±1.7%	52.9%	68.4%	32.0%	0.568
<b>1-2</b>	<b>70.6%±1.4%</b>	<b>70.2%</b>	<b>71.9%</b>	<b>41.6%</b>	<b>0.626</b>
2-3	68.7%±5.6%	67.4%	69.2%	33.6%	0.572
1-3	70.6%±1.3%	70.3%	71.7%	41.5%	0.623

Tabela 5.8: Resultados obtidos para a tarefa de classificação de emoção do modelo SVM

Os resultados obtidos inicialmente pelo modelo SVM demonstram que este obtém piores resultados que o modelo Naive Bayes. Ao comparar os resultados obtidos com aqueles disponíveis na tabela 5.2 vemos que o modelo SVM tem um desempenho inferior de aproximadamente 10%. No entanto, um ponto equivalente de ambos os modelos é a seleção dos n-gramas, onde a utilização de unigramas e bigramas provou ser a melhor combinação. Este resultado acrescenta mais força à teoria de o conteúdo emocional estar concentrado em termos individuais e pequenas expressões. Comparando os restantes resultados, verificamos que os valores de *precision* e de *recall* são semelhantes com os valores de *specificity* a serem ligeiramente superiores. Estes resultados são um indicador que o modelo de SVM efetua pior a tarefa de classificação de emoção, que o modelo Naive Bayes. Comparamos também os dois modelos na tarefa de previsão de *emoji*. Na tabela 5.9 estão disponibilizados os resultados do modelo SVM, mas para a previsão de *emoji*.

Tal como para a classificação de emoção, o modelo SVM tem um desempenho inferior para a previsão de *emojis*, quando comparado com o modelo Naive Bayes. Analisando os valores de F1, verificamos que estes são muito baixos, mesmo comparados com aqueles obtidos pelo modelo Naive Bayes. Outro ponto negativo é que não existe concordância a nível da seleção de n-gramas, como aconteceu para a tarefa de classificação de emoção. O modelo SVM obtém melhores resultados utilizando combinações de unigramas, bigramas e trigramas. Apesar de não ser a mesma escolha que o modelo Naive Bayes, verificamos

N-grams	10 Cross Validation	Treino e Teste			
	Accuracy	Precision	Recall	Specificity	F1
1	23.0%±1.4%	25.3%	23.8%	89.2%	0.158
2	19.3%±1.2%	30.1%	17.9%	86.5%	0.085
3	18.7%±1.8%	32.7%	15.3%	85.3%	0.056
1-2	21.2%±2.1%	28.9%	24.1%	89.1%	0.164
2-3	21.8%±1.9%	27.6%	18.9%	91.3%	0.162
<b>1-3</b>	<b>23.6%±1.3%</b>	<b>29.2%</b>	<b>22.1%</b>	<b>89.6%</b>	<b>0.169</b>

Tabela 5.9: Resultados obtidos para a tarefa de previsão de *emojis* do modelo SVM

que os resultados obtidos usando bigramas e trigramas não é muito diferente. Este ponto reforça a teoria que o teor emocional para a previsão de *emojis* se encontra também em expressões e não apenas em termos singulares. Mas tal como o modelo Naive Bayes, muitos destes resultados são obtidos devido ao conjunto de dados não se encontrar balanceado. Neste seguimento, efetuamos uma análise mais aprofundada ao melhor modelo SVM obtido. Os resultados obtidos por este modelo para cada emoção estão na tabela 5.10.

Emoção	Precision	Recall	Specificity	F1
Felicidade	71%	98%	12%	0.823
Raiva	73%	3%	100%	0.057
Nojo	92%	0%	100%	0.066
Medo	0%	0%	100%	0.000
Surpresa	0%	0%	100%	0.000
Tristeza	66%	16%	98%	0.265
<b>Total</b>	<b>70%</b>	<b>71%</b>	<b>40%</b>	<b>0.62</b>

Tabela 5.10: Resultados obtidos para a classificação de emoção do melhor modelo SVM

Analisando os resultados, verificamos que o problema de balanceamento de dados persiste no modelo SVM. Tendo em conta este problema, também verificamos a incapacidade de este modelo efetuar a classificação nas classes de medo e surpresa, onde o valor de F1 é igual a 0. Para além desta incapacidade, verificamos que o modelo classifica quase todos os dados como pertencentes à classe de felicidade e uma pequena minoria à classe de tristeza, sendo esta inferior que o modelo Naive Bayes. Analisando os resultados e comparando-os com o Naive Bayes, verificamos que o modelo SVM efetua de maneira inferior a classificação de emoções e previsão de *emojis*. Vamos analisar a matriz de confusão resultante deste processo, visível na figura 5.4.

A matriz de confusão revela que o cenário é ainda pior que o Naive Bayes, pois os valores ainda estão mais concentrados na coluna de felicidade. Para além deste problema, também verificamos que muitos valores na diagonal estão a 0. Isto indica que as classes de nojo, medo e surpresa são praticamente ignoradas pelo modelo quando aplicados a novos dados. Tal como no modelo Naive Bayes, isto é um cenário que não é nada aceitável e é necessário fornecer soluções para combater este problema. Para o problema de previsão de *emojis* já verificamos que os resultados obtidos foram piores. No entanto é importante verificar aqueles que obtiveram melhores resultados. Ao observar os *emojis* com melhores resultados, podemos verificar se a capacidade de previsão diminui ou não consoante a sua popularidade, bem como a relação entre emoção e *emojis*. Tal como para o modelo anterior, apresentam-se na tabela 5.11, os resultados obtidos da previsão do modelo SVM para os melhores *emojis*.

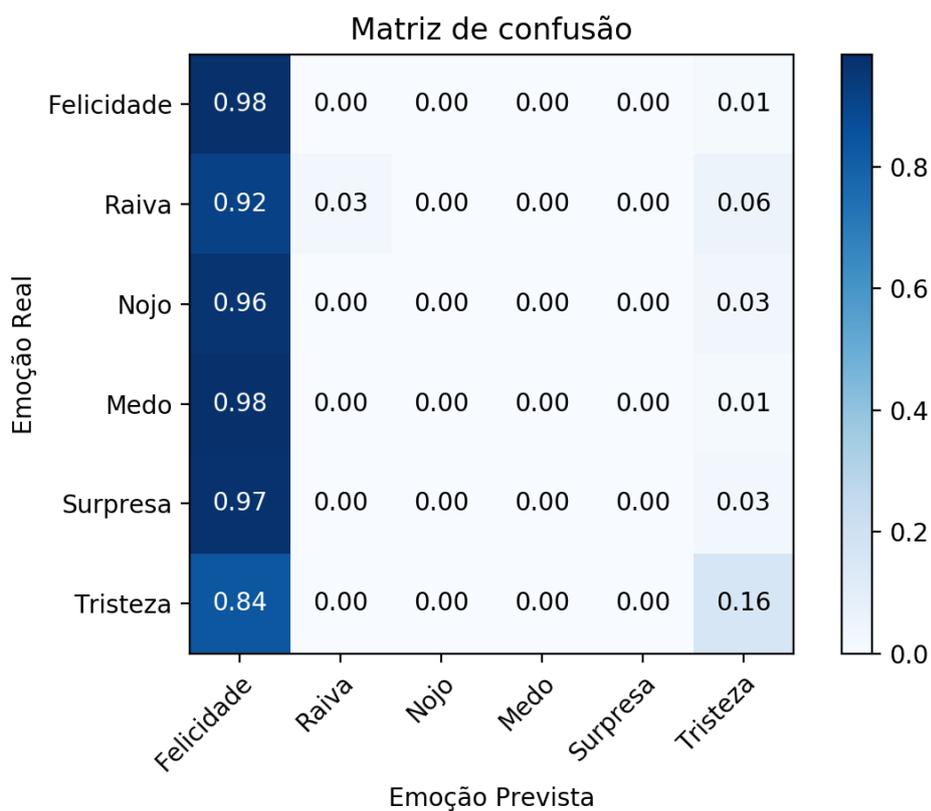


Figura 5.4: Matriz de confusão obtida para a classificação de emoção do modelo SVM

Emoji	Quantidade	Emoção Associada	Precision	Recall	Specificity	F1
😏	1%	Felicidade	36%	44%	89%	0.403
😠	2%	Raiva	34%	56%	85%	0.428
😞	1%	Raiva	37%	55%	83%	0.441
😊	3%	Felicidade	34%	38%	98%	0.366
❤️	12%	Felicidade	21%	20%	98%	0.214
😮	2%	Surpresa	26%	11%	99%	0.169
😇	1%	Felicidade	21%	19%	99%	0.202

Tabela 5.11: Melhores resultados obtidos da previsão de *emojis* do modelo SVM

Verificamos que estes resultados são ligeiramente superiores e incluem a adição de alguns *emojis* que antes não estavam presentes. O caso mais peculiar é o caso do *emoji* de coração (❤️), que obtém um resultado significativamente alto quando comparado com o modelo anterior. Para além deste ponto, o facto de ser muito utilizado no conjunto de dados torna este acontecimento mais notável. Analisando todos os *emojis*, verificamos que o modelo SVM consegue reconhecer significativamente alguns deles, mas por contrapartida há imensos valores de F1 a 0. Outro ponto curioso é que os *emojis* mais utilizados apresentam valores consistentemente mais altos que para o modelo Naive Bayes. Os resultados obtidos para esse conjunto de *emojis* encontram se presentes na tabela 5.12.

Emoji	Quantidade	Emoção Associada	Precision	Recall	Specificity	F1
😊	17%	Felicidade	28%	13%	99%	0.181
😡	1%	Raiva	20%	11%	100%	0.142
😞	4%	Tristeza	18%	10%	100%	0.137
😲	2%	Surpresa	22%	11%	100%	0.152
😱	1%	Medo	19%	8%	100%	0.110
🤢	1%	Nojo	25%	16%	96%	0.201

Tabela 5.12: Resultados obtidos da previsão dos *emojis* mais usados do modelo SVM

Analisando este conjunto específico, vemos que o modelo SVM é o oposto do Naive Bayes no tratamento destes *emojis*. Apesar de resultados de F1 significativamente baixos, o modelo SVM consegue efetuar alguma previsão acertada. O caso mais positivo é o do *emoji* associado à emoção de felicidade, que é o mais comum do conjunto de dados (correspondendo a 15%) e mesmo assim obtém o segundo valor mais elevado do conjunto em questão. Apesar dos resultados gerais serem inferiores quando comparados com o Naive Bayes, verificamos que o modelo SVM é ligeiramente superior no tratamento dos casos mais recorrentes. Tendo em conta que os *emojis* mais utilizados são aplicados em imensos contextos, o modelo SVM consegue, em certa medida, efetuar a previsão acertada destes, o que é um ponto a favor deste modelo.

Tal como no modelo Naive Bayes, procedeu-se ao balanceamento dos dados utilizando o método de *undersampling*. De maneira a fazer as comparações finais, verificamos o desempenho do modelo após ser aplicado o balanceamento dos dados, ou seja, num conjunto onde todas as classes têm o mesmo número de elementos. Estes dados são então aplicados a um novo processo de treino e conseqüentemente avaliado. Os resultados desta experiência para a tarefa da classificação de emoção encontram-se na tabela 5.13.

Emoção	Precision	Recall	Specificity	F1
Felicidade	88%	46%	89%	0.603
Raiva	25%	41%	91%	0.318
Nojo	21%	35%	84%	0.274
Medo	4%	49%	93%	0.088
Surpresa	3%	38%	85%	0.063
Tristeza	47%	40%	90%	0.431
<b>Total</b>	<b>67%</b>	<b>43%</b>	<b>89%</b>	<b>0.505</b>

Tabela 5.13: Resultados obtidos com dados balanceados para a classificação de emoção do modelo SVM

Os resultados obtidos para a tarefa de classificação com os dados balanceados foram

algo surpreendentes, tendo em conta a sucessão de valores obtidos anteriormente. Em grande parte, isto deve-se ao facto do modelo apresentar um desempenho inferior de aproximadamente 10% quando comparado com o modelo Naive Bayes. Neste caso obtemos um modelo com valores muito semelhantes, sendo que a dificuldade em classificar as classes de medo e surpresa se mantêm. As restantes classes apresentam valores muito idênticos. Para uma análise mais detalhada é apresentado a matriz de confusão obtida, na figura 5.5.

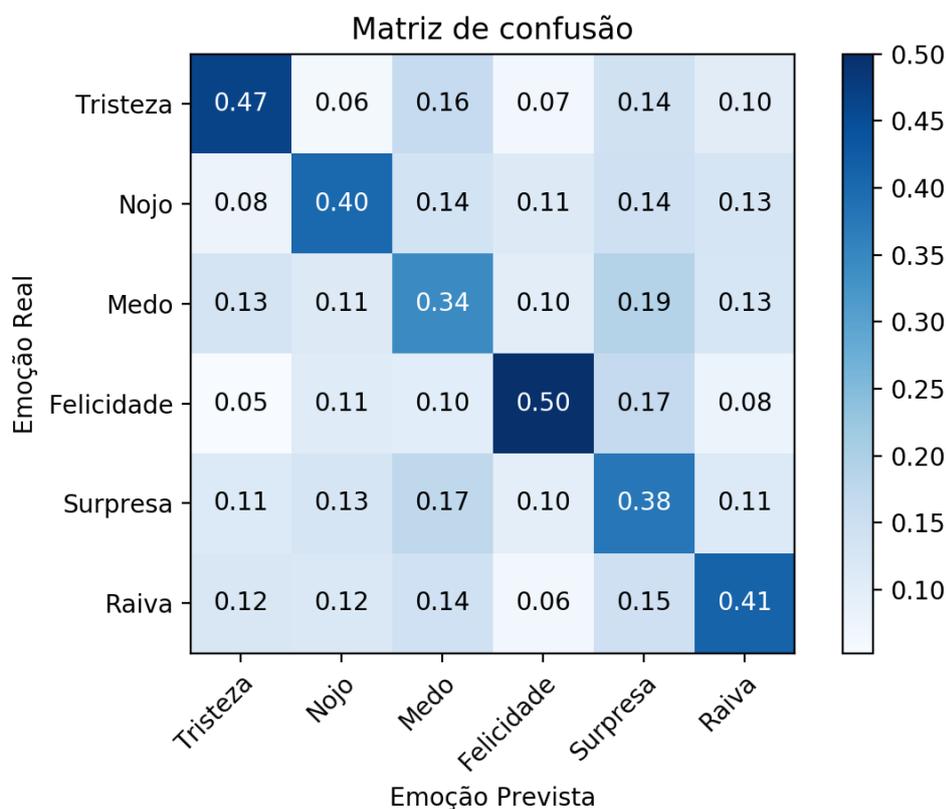


Figura 5.5: Matriz de confusão para a classificação de emoção com dados balanceados do modelo SVM

A matriz de confusão demonstra que afinal os resultados da tabela 5.13 são um pouco enganadores. Verificamos que o modelo SVM não é assim tão equivalente ao modelo Naive Bayes em termos de resultados. A principal razão deve-se aos elementos fora da diagonal serem ligeiramente superiores aos do modelo Naive Bayes. Assim, verifica-se que apesar dos resultados por classe serem semelhantes, o modelo tem mais dificuldade a efetuar a classificação do que usando Naive Bayes. Existem diversos motivos pelo qual o modelo Naive Bayes, que inicialmente pensamos que obteria piores resultados, ser na realidade superior ao SVM. Essa discussão é aprofundada no capítulo 6. Efetuamos de seguida a análise de resultados para o último modelo desenvolvido, baseado em redes neuronais com arquitetura LSTM.

## 5.4 LSTM

A rede neuronal LSTM foi o modelo desenvolvido em que se depositaram mais esperanças relativamente ao desempenho. Isto porque este tipo de redes tem apresentado melhores resultados em grande parte das tarefas de classificação e na bibliografia analisada (Barbieri

et al., 2018a; Felbo et al., 2017; Tato, Nkambou e Frasson, 2018). Para além deste ponto, o modelo LSTM efetua o processo de classificação de forma diferente, simulando até certa parte o cérebro humano. Apesar de parecer um ponto insignificante, os resultados obtidos irão clarificar se o teor emocional está relacionado com um conjunto específico de termos ou expressões utilizadas com regularidade no conjunto de dados.

A principal tarefa em que se espera que o modelo LSTM seja superior é na previsão de *emojis*. Isto deve-se à própria arquitetura do modelo, onde esperamos encontrar padrões na utilização de certos *emojis* e assim efetuar a previsão correta com mais facilidade. Por outro lado, a utilização de *emojis* diferentes em textos semelhantes, como verificamos, torna toda esta tarefa muito mais difícil. Ao contrário dos modelos anteriores, a rede LSTM não necessita de efetuar as experiências verificando o melhor conjunto de n-gramas a utilizar. Isto porque o modelo irá efetuar a classificação tendo em conta o texto todo, baseado num *tokenizer* específico da biblioteca Keras. Para além disso, é necessário efetuar um *padding* aos dados, e assim garantir que, independentemente do formato dos dados de entrada, estes tenham todos a mesma dimensão. De forma a efetuar as primeiras análises do modelo, foram realizadas as tarefas de classificação da emoção e previsão de *emoji*. Os resultados obtidos para a classificação de emoção encontram-se na tabela 5.14 e os de previsão de emoji na tabela 5.15.

Emoção	Precision	Recall	Specificity	F1
Felicidade	81%	91%	55%	0.861
Raiva	48%	22%	98%	0.304
Medo	35%	7%	99%	0.128
Nojo	47%	12%	96%	0.206
Surpresa	27%	5%	98%	0.082
Tristeza	57%	58%	90%	0.587
<b>Total</b>	<b>72%</b>	<b>75%</b>	<b>67%</b>	<b>0.726</b>

Tabela 5.14: Resultados obtidos para a classificação de emoção do modelo LSTM

	10 Cross Validation	Treino e Teste			
	Accuracy	Precision	Recall	Specificity	F1
<b>Total</b>	24.3%±1.8%	26%	29%	93%	0.247

Tabela 5.15: Resultados obtidos para a tarefa de previsão de *emojis* do modelo LSTM

Analisando os resultados obtidos verificamos que as esperanças depositadas no modelo LSTM se confirmam. Verificamos que o modelo LSTM efetua a classificação e previsão de forma mais eficiente do que os modelos Naive Bayes e SVM. No entanto, este continua a sofrer de alguns problemas semelhantes. Apesar do problema de balanceamento de dados persistir, o modelo LSTM não tem tanta tendência em classificar os dados como pertencentes à classe de felicidade. As classes de medo e de surpresa continuam a ser aquelas mais difíceis de classificar mas verifica-se um aumento significativo nos valores de F1 quando comparado com os restantes. As classes de tristeza, raiva e nojo também têm um aumento significativo nos seus valores, demonstrando mais uma vez que o modelo LSTM parece ser superior aos restantes. Na previsão de *emojis* apenas verificamos que esta obtém resultados ligeiramente superiores do que os restante modelos, mas nada de muito significativo. Para podermos efetuar mais comparações entres os modelos vamos recorrer à matriz de confusão de forma a verificar o desempenho entre as classes. A matriz de confusão encontra-se na figura 5.6.

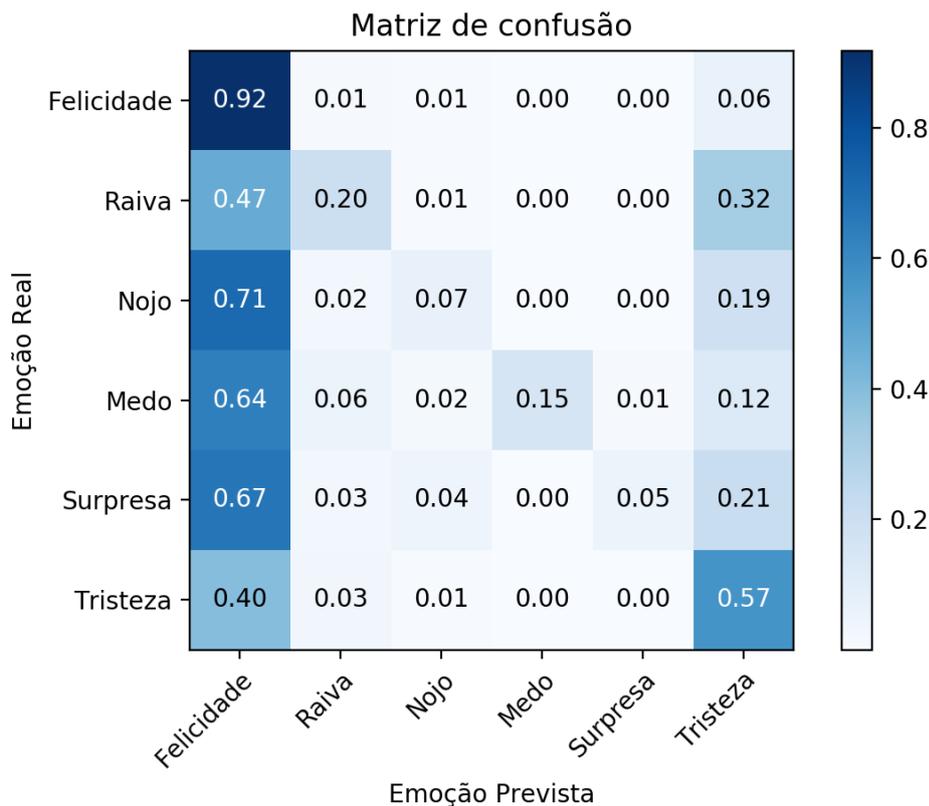


Figura 5.6: Matriz de confusão para a classificação de emoção do modelo LSTM

Da matriz de confusão verificamos que o modelo se assemelha mais ao pretendido do que os modelos. Isto porque os valores da diagonal estão mais preenchidos que os restantes. Por outro lado, a coluna correspondente à felicidade continua com valores muito elevados, o que mostra uma tendência em classificar os dados como pertencentes a esta classe.

Analisando os resultados obtidos para a previsão de *emojis*, verificamos que o modelo LSTM é ligeiramente superior aos restantes. Mais do que os resultados gerais, é crucial verificar como é que este modelo está a efetuar a previsão ao nível de cada *emoji*. Na tabela 5.16 estão apresentados os resultados obtidos para os melhores *emojis* da tarefa de previsão.

Emoji	Quantidade	Emoção Associada	Precision	Recall	Specificity	F1
😞	0.7%	Tristeza	21%	33%	94%	0.256
😄	2%	Felicidade	33%	49%	87%	0.408
😡	1%	Raiva	24%	13%	99%	0.178
😁	3%	Felicidade	34%	38%	98%	0.366
❤️	12%	Felicidade	20%	21%	97%	0.212
😮	2%	Surpresa	32%	60%	82%	0.419
😓	3%	Tristeza	27%	20%	99%	0.235

Tabela 5.16: Melhores resultados obtidos da previsão de *emojis* do modelo LSTM

Ao verificar os *emojis* pertencentes à tabela 5.16 verificamos que estes, não divergem muito dos melhores *emojis* dos modelos anteriores. Verificando estas ocorrências podemos

supor que estes são os *emojis* mais fáceis de prever do conjunto de dados, onde existe uma variedade de hipóteses para tal acontecer. No contexto dos resultados, verificamos que estes são equivalentes aos modelos anteriores. No entanto as expectativas iniciais era que o modelo LSTM deveria ter a capacidade de produzir resultados mais acertados, do que os restantes modelos. Vamos verificar os resultados obtidos para os *emojis* mais comuns no conjunto de dados. Esses resultados estão disponíveis na tabela 5.17.

Emoji	Quantidade	Emoção Associada	Precision	Recall	Specificity	F1
😊	17%	Felicidade	0%	0%	100%	0.000
😡	1%	Raiva	19%	4%	99%	0.072
😞	4%	Tristeza	0%	0%	100%	0.000
😮	2%	Surpresa	98%	2%	99%	0.055
😱	1%	Medo	15%	19%	97%	0.176
💩	1%	Nojo	18%	18%	98%	0.189

Tabela 5.17: Resultados obtidos da previsão dos *emojis* mais usados do modelo LSTM

No caso dos *emojis* mais comuns, verificamos que os resultados obtidos são bastante mistos. O que se retira principalmente é que o modelo só consegue efetuar corretamente a previsão nos *emojis* menos utilizados. No caso dos *emojis* com a emoção associada de felicidade e tristeza, aqueles com mais utilização dentro deste conjunto, não se consegue efetuar a previsão. Destes pontos é demonstrado o porquê de o modelo LSTM obter resultados significativamente mais altos que os restantes modelos. Os *emojis* com menos utilização são aqueles que conseguem ser previstos, muito devido ao facto de não serem utilizados com tanta ambiguidade. Isto demonstra que o modelo LSTM efetua a previsão para muitos dos 62 *emojis* consideravelmente bem, e aqueles com maior utilização, obtêm piores resultados.

Já se verificou que o modelo aplicado ao conjunto de dados na sua totalidade, apresenta resultados mais adequados do que os anteriores. Apesar das métricas obtidas não serem muito distantes do modelo Naive Bayes, verifica-se que o processo de classificação é menos tendencioso usando LSTM. Esta observação fornece a base para algumas conclusões que são depois descritas no capítulo 6. Mas para poder dizer que o modelo LSTM é melhor a efetuar a classificação do que os restantes modelos, é necessário verificar o seu desempenho quando treinado com o conjunto de dados balanceados. Desta forma, prosseguiu-se para o balanceamento de dados usando novamente o processo de *undersampling*. Após efetuar este processo, aplicou-se o conjunto de dados resultante ao treino do modelo. Os resultados obtidos para a tarefa de classificação de emoção, estão presentes na tabela 5.18.

Emoção	Precision	Recall	Specificity	F1
Felicidade	48%	53%	89%	0.509
Raiva	39%	39%	87%	0.355
Nojo	41%	44%	88%	0.423
Medo	26%	18%	89%	0.211
Tristeza	28%	34%	83%	0.318
Surpresa	29%	28%	87%	0.282
<b>Total</b>	<b>35%</b>	<b>35%</b>	<b>87%</b>	<b>0.358</b>

Tabela 5.18: Resultados obtidos com dados balanceados para a classificação de emoção do modelo LSTM

Analisando os resultados obtidos, verificamos que estes parecem ser ligeiramente diferentes aos obtidos pelos outros modelos. O primeiro aspeto que diverge dos modelos anteriores é que os resultados para as classes de medo e surpresa são significativamente superiores. No entanto, os valores para as restantes classes diminuem, quando comparados com estes mesmos modelos. De forma a podermos entender as diferenças entre os modelos, recorreremos à matriz de confusão para observar como o modelo LSTM está a efetuar a classificação. A matriz encontra-se disposta na imagem 5.7.

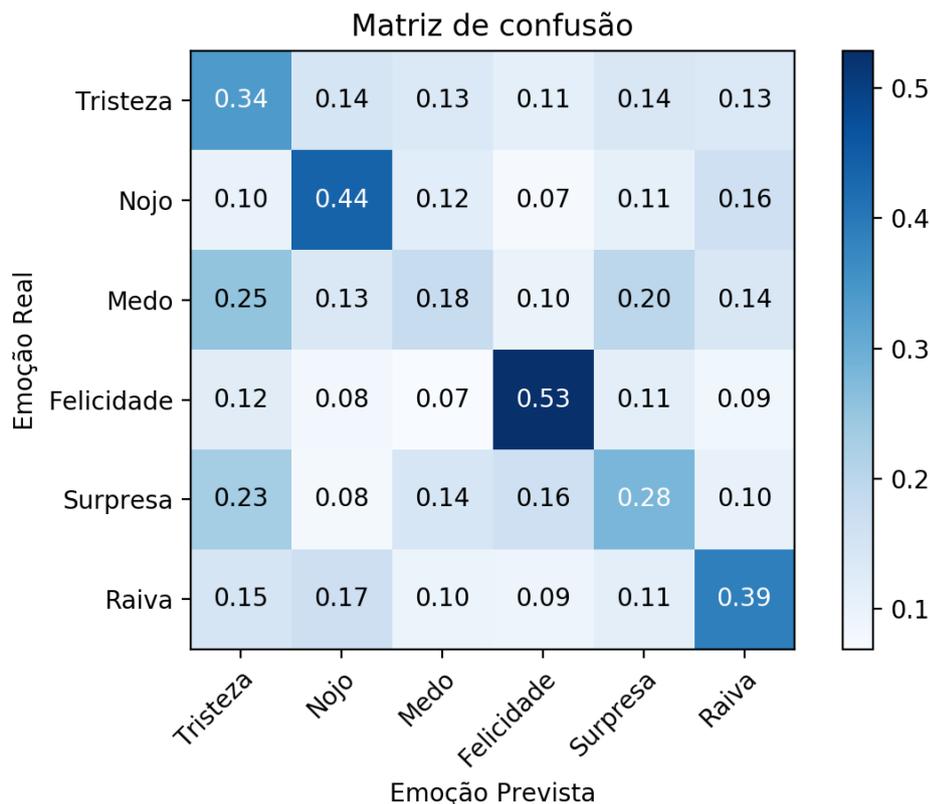


Figura 5.7: Matriz de confusão para a classificação de emoção com dados balanceados do modelo LSTM

Verificamos que o modelo LSTM tem resultados inferiores aos modelos anteriores, usando dados balanceados. O primeiro ponto que mostra esta inferioridade, tem a ver com o facto de os valores fora da diagonal serem superiores que os restantes modelos. O facto de o modelo LSTM obter melhores resultados usando o conjunto de dados completo, e ser inferior com os dados balanceados abre imensa discussão sobre a melhor arquitetura a aplicar ao problema. Apesar desta inferioridade, o modelo demonstra melhor capacidade na classificação dos casos raros, aqui representados como a classe de medo e surpresa. No entanto, não é capaz de atingir valores equivalentes para as restantes classes. Também verificamos que a classe de felicidade continua a ser aquela, com melhores resultados. A classe de nojo, ultrapassa a classe de raiva a ser classificada corretamente. O modelo LSTM apresenta assim diferenças significativas dos modelos Naive Bayes e SVM. Estas diferenças são discutidas no capítulo 6 bem como as conclusões retiradas de todos os modelos para a classificação de emoções.

## 5.5 Exploração das relações entre emojis

O modelo *word2vec* não apresenta resultados com base nas métricas apresentadas, pois não efetua nenhum processo de classificação. Como abordado na secção 2.3, o modelo permite calcular as distâncias e/ou similaridades, traduzidas por valores numéricos para cada par de termos do conjunto de dados. Desta forma, podemos localizar cada termo num espaço dimensional e conseqüentemente verificar, por exemplo, se a sua disposição sugere um agrupamento, que pode até estar relacionado com as emoções transmitidas. O foco desta experiência foi então a visualização da localização espacial dos vários *emojis* em estudo. Desta forma, poderia-se também verificar se estes *emojis*, são agrupados conforme as divisões propostas por Wood e Ruder (2016).

Como abordado na secção 2.3, o resultado do modelo *word2vec* é um vetor de 300 dimensões para cada termo. A visualização destes termos no espaço só é possível até três dimensões, logo é necessário uma transformação do vetor de 300 dimensões para duas ou três. Existem diversos métodos para efetuar esta redução, sendo que a mais usada é o método de Principal Component Analysis (PCA). O PCA é muito utilizado no âmbito de *feature reduction* de forma a eliminar características que estejam relacionadas e que não acrescentam informação adicional. Devido à necessidade que temos em visualizar a localização espacial e não propriamente em reduzir a dimensão dos termos optou-se por outra técnica, o T distributed Stochastic Neighbor Embedding (t-SNE). O t-SNE é um algoritmo usado para a redução de dimensionalidade em dados com muitas dimensões, de forma a que seja possível a visualização num espaço de duas ou três dimensões. Especificamente, o t-SNE vai modelar cada objeto de alta dimensão num ponto bi ou tridimensional, de tal forma que termos similares, são modelados por pontos próximos. Conseqüentemente, termos diferentes são modelados por pontos distantes no espaço definido.

A aplicação do método de t-SNE permite a representação dos termos do modelo *word2vec*, num espaço de duas ou três dimensões. Entre eles, é possível visualizar todos os *emojis* em análise, tal como na figura 5.8.

Analisando a disposição dos *emojis*, verifica-se que não existem grupos bem definidos de emoções. No entanto é possível observar resultados interessantes na disposição dos *emojis*. O primeiro é que os *emojis* pertencentes à classe de tristeza e de raiva encontram-se muito próximos. Esta proximidade pode dever-se em grande parte ao sentimento negativo com que são utilizados, mas é um ponto difícil de concluir, pois existem alguns *emojis* da classe de felicidade também próximos. Numa análise geral, verificamos que existem alguns *emojis* com valores fora do comum. Alguns destes casos são as variantes de *emojis* com cara de gato (e.g. 🐱) e os *emojis* pertencentes à classe de nojo. Estes últimos, não são possíveis de serem visualizados devido a problemas de representação dos *emojis* com a biblioteca do *Python*, *matplotlib*<sup>2</sup>.

Apesar de os *emojis* não criarem grupos específicos com base nas emoções definidas por Wood e Ruder (2016), é possível verificar que existe alguma similaridade entre si. As únicas exceções são os casos de alguns *emojis* pertencentes à classe de felicidade, algo que pode ser justificado com o elevado número de *emojis* pertencentes a esta classe. Um caso que ocorre no conjunto de dados e que foi verificado pelo resultado deste modelo, é que existem ocorrências de textos muito idênticos, mas com utilização de *emojis* diferentes. Isto faz com que os *emojis* em questão tenham uma representação próxima, dificultando, ao mesmo tempo, a previsão de *emojis*. Exemplos de textos semelhantes mas com utilização

---

<sup>2</sup><https://matplotlib.org>

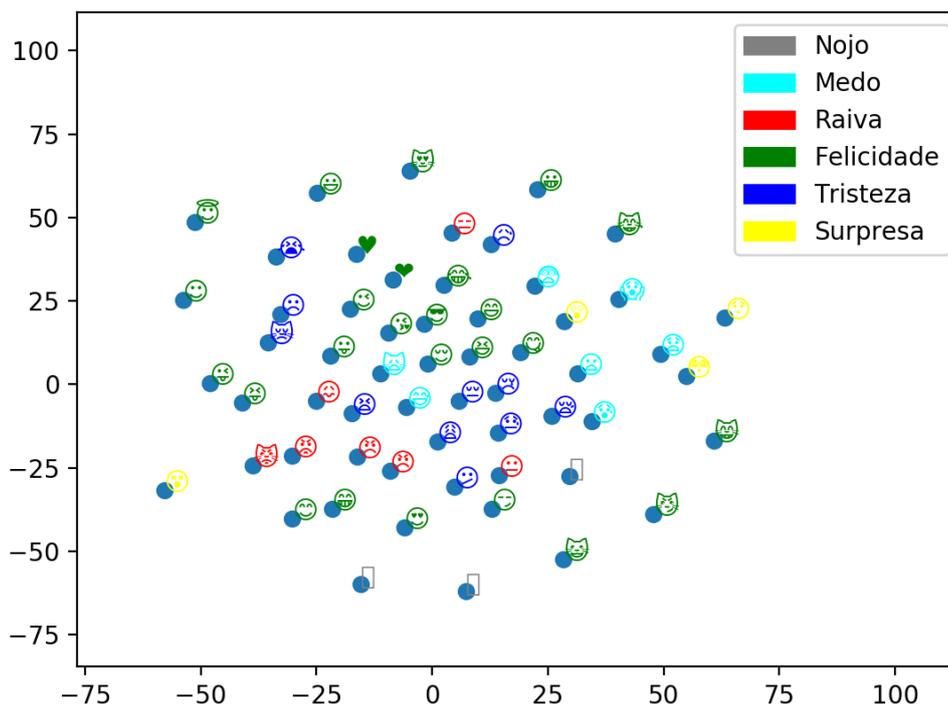


Figura 5.8: Similaridade entre os *emojis* obtidos pelo modelo *word2vec*

diferente de *emojis* encontram-se na figura 5.9.

Como eu odeio calor !!! 🤢🤔  
 Mano odeio esse calor 😡🔥

@USERNAME grande amor da minha vida 😍  
 @USERNAME Amor a primeira vista! 😍

Figura 5.9: Exemplos de *tweets* semelhantes com *emojis* diferentes

Pelos textos anteriores e outros semelhantes, verificamos que a seleção do *emoji* em situações idênticas varia de pessoa para pessoa sendo, em alguns quase, arbitrária. No caso do texto à direita da figura 5.9, a classificação do *emoji* será muito difícil de efetuar, por estarmos perante duas variações do mesmo emoji — a única diferença é que um utiliza a cara de gato em vez da cara humana. Esta opção estará em grande parte dependente da vontade do utilizador e assim torna mais difícil a generalização da classificação/previsão. No caso do texto da esquerda, denota-se a utilização de *emojis* da mesma classe mas todos com representação gráfica distinta entre si. Este tipo de textos diz bastante sobre a escolha dos *emojis* e a sua utilização nas redes sociais, que serão abordadas mais à frente nas conclusões obtidas.

A representação dos resultados do modelo *word2vec* para três dimensões foi também explorada, no entanto, resulta num agrupamento muito compacto e difícil de visualizar. O método de t-SNE permite a visualização dos vetores obtidos de cada termo em duas ou três dimensões, de uma dimensão inicial de 300, onde se perde sempre alguma informação. Desta forma, a localização inicial de cada termo será diferente da apresentada na figura 5.8.

Devido às limitações da mera observação dos resultados do t-SNE, decidiu-se

complementar esta exploração com uma abordagem alternativa, que permitisse entender melhor a relação entre *emojis*. Tirando partido do facto dos *emojis* se encontrarem representados por um vetor com 300 dimensões, realizou-se uma experiência em que se procura identificar agrupamentos de *emojis* de forma automática, recorrendo para isso ao algoritmo de *clustering k-means* (Hartigan, 1975). O método *k-means* atribui a novos dados um grupo definido tendo em conta a distância do centro do grupo mais próximo, sendo que esta proximidade foi calculada com base na distância Euclidiana. O valor de *k* indica o número de grupos existentes para agrupar os novos dados. Devido à figura 5.8 não apresentar nenhum grupo claro, aplicou-se o *k-means* aos *emojis* alvo, que procuramos agrupar em seis grupos, o número das emoções básicas de Ekman, também adotadas por Wood e Ruder (2016). Os seis grupos obtidos estão encontram-se na tabela 5.19.

Emojis	Cluster
	1
	2
	3
	4
	5
	6

Tabela 5.19: Clusters obtidos efetuando o método *k-means*

Desde logo verificamos que os *clusters* obtidos são diferentes dos grupos propostos por Wood e Ruder (2016). No entanto, têm forte relação emocional, sugerida também pela sua representação visual. O *cluster* 1 é aquele que apresenta mais resultados mistos, pois existe alguma diversidade em termos das classes originalmente atribuídas. Pela análise verificamos que todos os *emojis* têm tendência a transmitir um sentimento negativo. Encontramos *emojis* pertencentes à classe de tristeza, raiva e surpresa onde no entanto todos transmitem ideia de tristeza através da sua representação visual. Os *clusters* 2, 3 e 6 denotam todos o caráter de sentimento positivo sendo que pertencem quase todos à classe de felicidade. Muito interessante é o resultado obtido no terceiro *cluster*, onde todos os elementos usam representações com corações e estão, por isso, relacionados com o conceito de ‘amor’, também ele muito ligado a sentimentos positivos. O *cluster* 4 é constituído maioritariamente por *emojis* relacionados com a emoção de surpresa, para além de *emojis* de gatos, menos populares. Por fim o *cluster* 5 é aquele que apresenta mais discrepâncias. Podemos verificar que é constituído por elementos denotados como negativos, pertencendo mais à emoção de raiva. Também verificamos que alguns destes *emojis* encontram-se um pouco desenquadrados, sendo que são muito pouco utilizados ao longo do conjunto de dados. Esta pouca popularidade pode ter causado a alguns *emojis* serem agrupados com aqueles onde inicialmente eram indicativos claros de expressar a emoção de raiva.

Esta experiência permite-nos verificar que os *emojis* efetuam um agrupamento significativo por emoção. Apesar de não serem os grupos inicialmente estipulados conseguimos verificar que estes grupos de emoção são adequados para a atribuição da etiqueta no processo de extração de dados. De lembrar que os grupos criados por Wood

e Ruder (2016) tinham como base a representação e descrição dos *emojis* e não o contexto cultural dos utilizadores.

## 5.6 Desempenho dos modelos em títulos de notícias

Tendo em conta que um dos objetivos da abordagem proposta é um aliviar do esforço de recolha e anotação de dados para a classificação de emoções, decidimos averiguar qual seria o desempenho destes modelos para a classificação de emoções noutros tipos de texto. Assim, recorreremos ao conjunto de dados disponibilizado por Dosciatti, Ferreira e Paraiso (2015) para verificar o desempenho dos modelos treinado no nosso conjunto de dados. Como referido na secção 3.1, este conjunto de dados é constituído por 1556 pequenos textos, em português do Brasil, no qual dizem respeito ao título e subtítulo de notícias em jornais online. Tanto quanto sabemos, este conjunto de dados será o único com texto português manualmente anotado com as seis emoções de Ekman. Para além das seis emoções básicas, ainda existe uma categoria adicional designada por “neutro” que diz respeito aos textos que não têm conteúdo emocional, de acordo com os anotadores. Assim, utilizou-se este conjunto de dados onde foi retirado todos os textos que estavam atribuídos esta categoria. O conjunto de dados utilizado ficou então com a seguinte distribuição, visível na figura 5.10

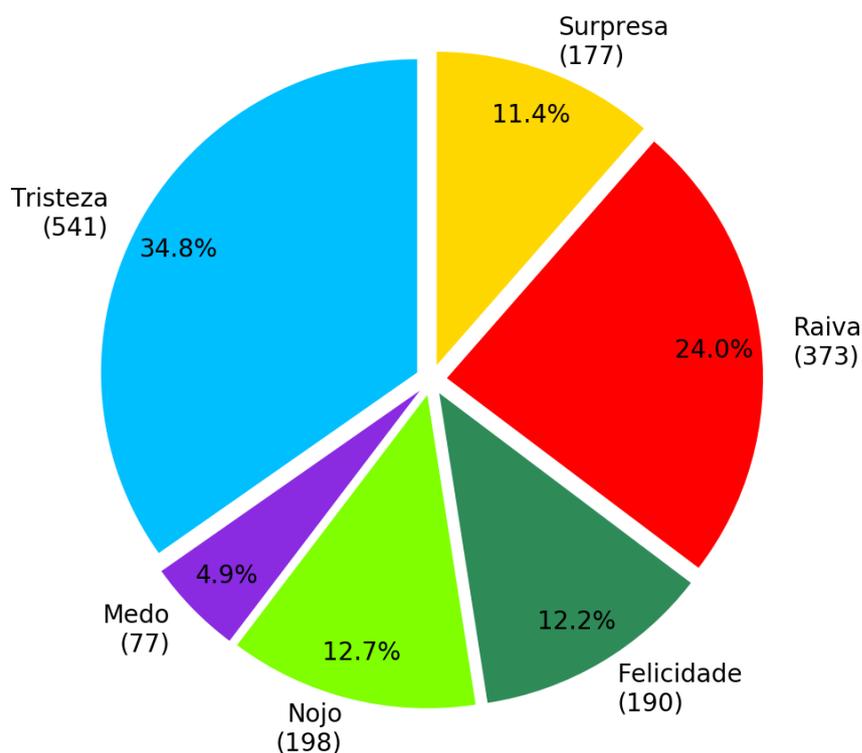


Figura 5.10: Distribuição do novo conjunto de dados a aplicar por emoção

Ao analisar a distribuição por emoção deste conjunto de dados verificamos que tem uma distribuição diferente do que o desenvolvido para o treino e teste dos modelos. Verificamos que existe um maior equilíbrio entre as seis emoções, mas com a predominância das classes de raiva e de tristeza em vez da felicidade. O próximo passo consistiu em utilizar

os três modelos treinados com o conjunto de dados balanceado que obtiveram melhores medidas F1 nas experiências anteriores para classificar a emoção neste conjunto de dados de títulos de notícia. Para além destes modelos também se aplicou a abordagem *baseline* anteriormente aplicada para este conjunto de dados. Os resultados obtidos encontram-se nas tabelas 5.20, 5.21, 5.22 e 5.23.

Emoção	Precision	Recall	Specificity	F1
Felicidade	21%	9%	95%	0.182
Raiva	8%	19%	89%	0.177
Nojo	13%	15%	87%	0.191
Medo	14%	26%	77%	0.199
Tristeza	45%	25%	84%	0.324
Surpresa	25%	29%	73%	0.276
<b>Total</b>	<b>28%</b>	<b>23%</b>	<b>82%</b>	<b>0.255</b>

Tabela 5.20: Resultados obtidos do modelo Naive Bayes para o novo conjunto de dados

Emoção	Precision	Recall	Specificity	F1
Felicidade	15%	11%	92%	0.124
Raiva	9%	29%	84%	0.181
Nojo	10%	21%	75%	0.173
Medo	14%	17%	85%	0.169
Tristeza	44%	20%	89%	0.287
Surpresa	28%	22%	84%	0.267
<b>Total</b>	<b>28%</b>	<b>21%</b>	<b>86%</b>	<b>0.201</b>

Tabela 5.21: Resultados obtidos do modelo SVM para o novo conjunto de dados

Emoção	Precision	Recall	Specificity	F1
Felicidade	13%	12%	89%	0.161
Raiva	4%	10%	88%	0.097
Nojo	14%	21%	83%	0.185
Medo	16%	24%	81%	0.223
Tristeza	38%	6%	95%	0.140
Surpresa	22%	32%	64%	0.291
<b>Total</b>	<b>24%</b>	<b>17%</b>	<b>83%</b>	<b>0.182</b>

Tabela 5.22: Resultados obtidos do modelo LSTM para o novo conjunto de dados

Emoção	Accuracy	Precision	Recall	F1
Felicidade	3.1%	2.7%	1.5%	0.025
Raiva	2.5%	1.5%	1.5%	0.016
Nojo	2.7%	2.0%	1.9%	0.019
Medo	1.0%	0.8%	0.7%	0.007
Tristeza	5.2%	4.3%	3.8%	0.039
Surpresa	0.8%	0.1%	0.1%	0.001
<b>Total</b>	<b>2.5%</b>	<b>1.9%</b>	<b>1.6%</b>	<b>0.022</b>

Tabela 5.23: Resultados obtidos da abordagem *baseline* para o novo conjunto de dados

Ao analisar os resultados obtidos verificamos que estes têm valores de F1 mais baixos que os obtidos para a classificação da emoção de tweets. Isto já seria de esperar, dado que

os dados são significativamente diferentes. Ao comparar os nossos resultados com aqueles obtidos em Dosciatti, Ferreira e Paraiso (2015) verificamos que também são inferiores, visto que os autores obtêm um valor de  $F1 \approx 0.51$  usando um modelo SVM treinado e testado com o conjunto de dados em questão. Outro ponto que se verifica é que os modelos Naive Bayes e SVM continuam com valores semelhantes enquanto o modelo LSTM fica ligeiramente atrás. No entanto, apesar destes valores serem mais baixos que os resultados obtidos no nosso conjunto de dados surgem resultados interessantes. O primeiro diz respeito aos valores obtidos para a classe de felicidade, sendo que no conjunto de dados obtido esta é a emoção predominante e neste caso surge como uma das classes menos representada. Os valores obtidos para a felicidade são das seis emoções, aquela com pior desempenho, o que demonstra que efetuar o balanceamento ajudou a melhorar os modelos desenvolvidos. Para além do mau desempenho para a classe de felicidade, também se verifica, por contraste, que as emoções de surpresa e nojo obtêm resultados superiores. Usando a abordagem *baseline* esta obtêm resultados ligeiramente superiores do que aqueles usando o nosso conjunto de dados, mas no entanto apresenta valores muito baixos.

Existem várias razões para os modelos não terem obtido resultados semelhantes aos anteriores ( $F1 \approx 0.5$ ), sendo que algumas são explicadas no trabalho de Dosciatti, Ferreira e Paraiso (2015). Como referido anteriormente, uma das conclusões a que os autores chegaram foi que os textos jornalísticos não têm grande variabilidade de emoções, o que pode criar discordâncias da classe a atribuir entre os anotadores humanos. Logo, surge a hipótese de que as classes atribuídas aos textos podem estar incorretas e assim os modelos poderão estar a classificar a emoção corretamente mas através da discordância entre anotadores, foi atribuída outra emoção. Devido à utilização dos *emojis* como forma de atribuir a emoção ao texto, no nosso conjunto de dados este problema é minimizado.

Outro aspeto que leva os modelos a obter resultados inferiores deve-se à forma de como os textos estão escritos. Como indicado pelos autores os textos jornalísticos são objetivos, com pouco conteúdo emotivo. Por outro lado, os textos presentes no nosso conjunto de dados são totalmente distintos, com predominância para as emoções de felicidade e de tristeza o que cria confusão entre os modelos. Com esta experiência conseguimos perceber que quanto mais objetiva for a escrita menos emocional será, logo, os modelos desenvolvidos terão dificuldade em atribuir a emoção a este tipo de textos. Por outro lado com este tipo de textos é verificado usando a abordagem *baseline* que as palavras existentes neste conjunto de dados ocorrem mais frequentemente no léxico NRC devido a não conter erros ortográficos. No entanto mesmo conseguindo atribuir uma emoção ao texto verificamos que esta abordagem continua a não ter um bom desempenho. No capítulo 6 a discussão dos resultados obtidos será aprofundada.

Esta página foi intencionalmente deixada em branco.

## Capítulo 6

# Conclusão

Este capítulo relata as principais conclusões obtidas durante o desenvolvimento deste trabalho e a sua discussão. De forma a estruturar melhor todo o conhecimento adquirido, bem como as conclusões, este capítulo foi dividido em seções dedicadas a cada ponto importante do trabalho. Estas seções abordam a extração de dados para a construção do conjunto de dados, as diferenças e semelhanças dos três modelos de aprendizagem computacional desenvolvidos e ainda uma discussão geral sobre a classificação de emoções em texto e previsão de *emojis*. Por fim, o capítulo dedica uma seção ao trabalho futuro que pode ser realizado para uma melhor resolução dos problemas em questão.

### 6.1 Extração de dados

Das diversas conclusões que podem ser retiradas, a mais significativa é que o conjunto de dados é bastante importante para a tarefa em questão. Ou seja, temos de encarar o processo de construção do conjunto de dados com muito rigor e esta fase não pode ser encarada como uma simples extração. Outro ponto importante é a dificuldade em categorizar os dados com a sua emoção respetiva. Este aspeto constitui um grande entrave ao processo de construção de dados, pois grande parte deste tipo de trabalho recorre a anotadores humanos, para atribuir a cada texto a sua respetiva classe. Devido a esta dificuldade, optou-se por um método de categorização dos dados com base em novas características textuais muito presentes na *social media*, mais propriamente os *emojis*. Este método provou-se ser eficaz, pois permitiu obter cerca de milhão e meio de textos categorizados num curto espaço de tempo, e sem recorrer diretamente a recursos humanos. Para além disso, ao evitar utilizar anotadores humanos para efetuar esta categorização evitam-se também ambiguidades que estes introduzem durante este processo, o que torna o processo muito mais rápido. Por outro lado surgem outros problemas devido à forma como estas novas características são utilizadas, por exemplo, devido à existência de várias variantes de um *emoji*.

Neste contexto, foram tomadas decisões iniciais sobre a categorização de dados que influenciaram os resultados obtidos. A primeira foi atribuir a emoção de cada texto com base nos *emojis* definidos no trabalho de Wood e Ruder (2016). Apesar de se ter verificado que a divisão emocional não está muito distante da que é apresentada neste trabalho, com base no *clustering* e similaridade de emojis calculados sobre o conjunto de dados, a elevada quantidade de *emojis* pertencentes à classe de felicidade desequilibrou o conjunto de dados. A existência de pequenas variantes entre os vários *emojis* definidos só aumentou as divisões efetuadas e assim tornou a tarefa de previsão de *emoji* mais complexa. Assim uma redução

dos *emojis* pertencentes a cada grupo, usando apenas aqueles mais utilizados e com maior relação entre a representação visual e a emoção em questão será algo a pensar em trabalhos futuros.

Para além deste aspeto, nas tarefas efetuadas até agora, dá para retirar pequenas conclusões que são importantes referir. Uma conclusão retirada do conjunto de dados, é relativa ao teor dos conteúdos partilhados na *social media*. A predominância de dados pertencentes à classe de felicidade é um indicativo que os utilizadores têm uma maior tendência a partilhar conteúdo positivo. A utilização de textos provenientes das redes sociais terá maior teor emocional mas, em contrapartida, este texto seguirá menos convenções como a utilização de maiúsculas, para além de erros e termos de calão, o que pode dificultar a classificação da emoção.

## 6.2 Processamento dos dados

O processamento de dados é uma tarefa muito importante no âmbito da aprendizagem computacional com textos. O processo efetuado começou por realizar uma limpeza inicial aos dados obtidos. Esta limpeza consistiu na remoção de conteúdo do texto que não acrescenta nenhum tipo de informação, e textos contraditórios no âmbito do problema em questão. Como abordado, todas estas decisões foram suportadas por bibliografia existente (Felbo et al., 2017; Hussien et al., 2016), que suportam a ideia de que o processo foi bem efetuado.

Efetuando testes de  $\chi^2$ , verificamos a existência do elevado relacionamento entre um conjunto de termos e as emoções definidas. Isto sugere que a ocorrência destes termos seja um ponto forte para classificar o texto como pertencente à emoção com que este está relacionado. É certo que o valor de  $\chi^2$  não nos vai permitir efetuar qualquer tipo de classificação, mas demonstra mais uma vez que parte significativa do teor emocional de um texto/frase está presente num determinado conjunto de termos. Para além desta relação, vemos que estes termos são principalmente utilizados em português do Brasil, pois como já foi abordado, é a variante do idioma português mais presente no conjunto de dados. Ou seja, a análise de resultados terá de sempre de ser feita com esta salvaguarda.

O trabalho também incluiu a extração de características adicionais dos textos. Apesar de se ter extraído um número considerável de características, verificamos através do processo de *gridsearch*, que estas não trazem benefícios ao desempenho dos modelos desenvolvidos. Isto deve-se ao facto de textos semelhantes, apresentarem valores diferentes para estas características e por isso serem dispostos muito distantes no espaço de procura. É certo que a utilização de palavras maiúsculas, bem como o número de pontos de exclamação, introduzem a noção de que é possível realçar mais o conteúdo emotivo da frase. Apesar de para o ser humano ser relativamente fácil perceber este ponto, é difícil passar esta ideia com o conjunto de dados obtido, para os modelos desenvolvidos, levando assim a mais confusão.

Os valores de valência, excitação e dominância calculados dos dois léxicos também levam a muita confusão. A principal causa é o facto de estes cobrirem apenas uma porção de palavras de um vocabulário bastante extenso. Outra causa deve-se à maneira como os textos estão escritos, sendo que grande parte vem com erros ortográficos, com a existência de termos em calão e palavras ou expressões típicas do português do Brasil. Isto leva a que grande quantidade de palavras não esteja presente nestes léxicos e a que muitos dos textos tenham um valor de 0 para estas características. Ao atribuir este valor, está a acontecer o mesmo que no caso dos pontos de exclamação e a quantidade de palavras

em maiúsculas, onde textos semelhantes podem ser levados para localizações no espaço de procura totalmente opostas. Isto introduz ao modelo maior confusão em vez de ser um benefício, daí a necessidade de retirar também estas características.

Por fim, a criação dos diversos modelos, tem imensos parâmetros para otimizar. Este sempre foi um dos maiores problemas da aprendizagem computacional, onde um processo de tentativa-erro é dos mais adequados e utilizados. Os principais parâmetros foram obtidos através do processo de *gridsearch*, de forma a obter sempre a melhor combinação possível. É certo que, para efetuar este processo é sempre necessário escolher um conjunto de valores de um universo infinito de possibilidades. Desta forma, tentou-se escolher os conjuntos mais adequados para a tarefa em questão e é um ponto da experiência em que não há muito mais a efetuar. Por fim, e como já referido, a construção do modelo LSTM é bastante flexível por permitir definir, em teoria, um número infinito de camadas. Tal como na otimização de parâmetros, foi escolhida a arquitetura que se verificou ser a mais adequada para a tarefa em questão. Esta arquitetura define o funcionamento do próprio modelo e variações da arquitetura escolhida serão um ponto de análise num trabalho futuro.

### 6.3 Modelos desenvolvidos

Dos modelos desenvolvidos, diversas conclusões relevantes podem ser retiradas e discutidas. O modelo *word2vec* era algo que não estava planeado inicialmente, mas foi desenvolvido de forma a explorar melhor a relação entre *emojis* e as emoções a que estes estão fortemente ligadas. Era também ambicionado verificar se a atribuição das emoções a *emojis* feitas por Wood e Ruder (2016) estavam corretas, visto este ser um ponto base de todo este trabalho. É certo que a visualização dos resultados através da técnica t-SNE não nos permite chegar a conclusões significativas, mas verificamos uma tendência no qual os *emojis* se agrupem conforme o sentimento. A utilização de diferentes *emojis* em textos quase iguais não ajuda muito ao modelo *word2vec*. Apesar de não obtermos claramente os seis grupos distintos de *emojis*, com base nas emoções estipuladas, a figura 5.8 é o resultado da visualização em duas dimensões. Visto que o resultado obtido pelo modelo é de 300 dimensões, mesmo usando o método t-SNE, a perda de informação é algo que devemos ter em conta. Por outro lado, ao agrupar os emojis com recurso ao *k-means*, verificamos que os *clusters* obtidos estão minimamente relacionados com as emoções que estes transmitem. Assim validamos também o processo de criação automática de dados, que terá, contudo, de ter em conta os contextos culturais dos autores dos textos.

Uma abordagem inicial com recurso ao léxico NRC mostrou-se ineficiente, com resultados de F1 muito perto de 0%. Como referido, estes valores devem-se a dois pontos em específico. O primeiro deve-se ao tipo de escrita informal presente no conjunto de dados, o que leva a que grande parte dos termos presentes nos textos não esteja presente no léxico. Devido a esta situação existe um número significativo de textos que não têm uma emoção atribuída ou que a emoção classificada tenha como fundamento apenas um termo presente no texto. O outro ponto diz respeito à forma como a análise ao texto é efetuada, sendo que esta abordagem faz uma análise de palavras individuais sem que a ordem desta tenha algum tipo de importância.

Analisando os resultados obtidos das diversas experiências, nenhum modelo se mostrou claramente superior. No entanto, pelos resultados obtidos, também verificamos que o modelo SVM não será o mais adequado à tarefa de classificação de emoção. Os dados obtidos e o seu consequente processamento foi algo que influenciou os resultados obtidos.

O problema dos dados terem uma classe predominante, a classe de felicidade, levou a que muitos dos textos aplicados posteriormente ao modelo fossem classificados como tal, quando não o são. Neste seguimento, decidimos repetir as experiências após um balanceamento dos dados. Ao efetuar o balanceamento, estamos a fazer com que a probabilidade de ocorrência de textos seja completamente ignorada e descartada, algo que é muito importante. Isto deve-se principalmente ao facto de os textos partilhados nas redes sociais serem conteúdos expressando felicidade ou tristeza, deixando as restantes emoções um pouco à parte. Para além disto, a quantidade de *emojis* associados a estas duas emoções é superior aos *emojis* das restantes classes.

Este problema é bastante comum no âmbito da aprendizagem computacional. Como é recorrente, também existe muita discussão sobre o tratamento a aplicar a estes dados. Ao aplicar o balanceamento de dados estamos a equilibrar os modelos desenvolvidos, mas também nos distanciamos da realidade do problema. O modelo LSTM (Barbieri et al., 2018a; Felbo et al., 2017; Tato, Nkambou e Frasson, 2018) foi pensado inicialmente como aquele que iria obter melhores resultados, muito devido aos resultados que tem obtido para tarefas de classificação, bem como por atribuir e se lembrar da significância de termos. Ao analisar os resultados para a tarefa de classificação com o conjunto de dados sem balanceamento, verificamos que esta expectativa inicial se confirmou, mas ao efetuar o balanceamento dos dados foi de facto o pior dos três modelos. Esta diferença de resultados deve-se ao modelo LSTM lidar melhor com grandes quantidades de exemplos, o que não acontece quando efetuamos o balanceamento através de *under-sampling*, como foi o caso, pois o número total de exemplos por classe passa a ser aproximadamente de 4 mil.

Na previsão de *emojis*, todos os modelos obtêm um fraco desempenho. Como dito anteriormente, a tarefa em questão é de elevada dificuldade devido ao número de classes para classificar. Os modelos desenvolvidos apresentam resultados semelhantes, mas há vários pontos interessantes a explorar. O principal é quanto mais utilizado e popular é o *emoji*, mais difícil se torna de o classificar. Isto deve-se ao facto de estes *emojis* serem utilizados em diversos contextos distintos, e em alguns casos de forma arbitrária, o que introduz mais um grau de dificuldade à tarefa em questão. Para os *emojis* com uma utilização média, os modelos já conseguem fazer alguma coisa, mas ainda com resultados muito baixos. Como visto no modelo *word2vec*, a proximidade de vários *emojis* também é um aspeto que dificulta a capacidade de previsão de *emojis* por parte dos modelos, levando assim a um baixo desempenho.

O balanceamento de dados permite que os modelos efetuem a classificação de forma a não darem tanta predominância para a classe de felicidade. Como dito anteriormente, neste cenário, o modelo LSTM é aquele que apresenta piores resultados. Por outro lado, tanto os modelos Naives Bayes como SVM obtêm resultados de F1 muito próximos e na casa dos 50%. Apesar de estar muito longe das métricas aceitáveis para modelos computacionais, temos de reforçar que o reconhecimento de emoções é uma tarefa altamente complexa e com alguma subjetividade, para além destes valores terem sido calculados para 4000 pequenos textos de cada classe, dum universo inicial de mais de um milhão.

Como referido, o balanceamento de dados influenciou todos os resultados. Um dos trabalhos a implementar no futuro, descrito na secção 6.4, passaria por extrair mais dados pertencentes às classes minoritárias de forma a tentar aumentar o seu número de exemplos. Mas com esta extração vem conseqüentemente a difusão da realidade do conteúdo partilhado e assim a perda de informação importante.

A aplicação destes modelos já com um treino de dados balanceados demonstrou alguma capacidade, ainda que pequena, de generalização. A utilização do conjunto de dados do trabalho de Dosciatti, Ferreira e Paraiso (2015) aos melhores modelos desenvolvidos obteve

resultados de F1 próximos de 20%. Um dos pontos principais para a obtenção destes resultados deve-se à escrita dos textos, que como os autores concluíram não tem grande conteúdo emocional. Para além de não serem tão expressivos como os textos obtidos do *Twitter*, o conjunto de dados de Dosciatti, Ferreira e Paraiso (2015) também poderá apresentar uma má classificação da emoção do texto por parte dos anotadores utilizados. Como estes textos são objetivos, devido a serem retirados de notícias, e de conterem pouco conteúdo emocional, os anotadores utilizados podem ter atribuído uma emoção que não é a adequada. Como estes textos não contêm características como os *emojis*, não é possível atribuir a emoção respetiva usando o processo efetuado durante a construção do conjunto de dados utilizado para este trabalho.

## 6.4 Discussão geral

De certo que os *emojis* influenciam em grande parte o conteúdo emocional dos textos em que estes ocorrem. Ao longo deste trabalho tentou-se efetuar a ponte sobre emoção e o próprio *emoji*, algo que se revelou difícil. Verificou-se que atribuir uma das seis emoções básicas de Ekman (1999) através dos *emojis* permite uma anotação rápida sem ambiguidades, pelo menos ao nível do objetivo a alcançar, já que cada *emoji* corresponde a uma emoção diferente. Com esta forma também surgem os problemas iniciais, como por exemplo, efetuar a atribuição de *emoji* a emoção. Uma das experiências realizadas procurou verificar se a divisão efetuada por Wood e Ruder (2016) seria a mais adequada no nosso contexto, mas observamos que a utilização e conseqüente agrupamento dos *emojis*, de acordo com o seu contexto, é diferente da inicialmente estipulada.

O conjunto de dados obtidos para efetuar o treino e validação dos modelos influenciou todas as experiências efetuadas. Um primeiro ponto será dizer que o conjunto de dados foi construído sem ter em conta a ocorrência de certos problemas. O conjunto de dados é o resultado das partilhas pelos utilizadores da rede social *Twitter*, em tempo real. Desta forma, se analisarmos os dados obtidos verificamos que os utilizadores partilham, em grande parte, conteúdo supostamente feliz ou triste, sendo este último menos frequente. Basta verificar os conteúdos dos utilizadores que seguimos ou que partilhamos nas redes sociais para confirmar esta observação. As pessoas partilham muito menos conteúdo nojento ou surpreendente, mas efetuam reações sobre algumas notícias, sendo que é aqui que se encontram mais frequentemente os conteúdos associados às restantes emoções. Apesar de generalista, esta é a forma de como as pessoas utilizam as redes sociais e isso é refletido no conjunto de dados obtido.

A previsão de *emojis* é uma tarefa muito complexa, muito devido ao teor de aleatoriedade subjacente. Atualmente a lista de *emojis* correspondentes a expressões faciais é superior a mais de 100 elementos, o que dificulta a tarefa de previsão. Se englobarmos a lista de todos os *emojis* este número ainda é superior visto existir categorias de *emojis* que vão desde bandeiras a animais. Para além desta centena, alguns destes *emojis* são variações entre si (por exemplo, os *emojis* regulares e as suas variantes de gato), o que torna a sua escolha muito mais difícil. É aqui que é preciso analisar o perfil do utilizador, pois a escolha do *emoji* está muito ligada a diversos fatores. Estes fatores vão desde o contexto da conversa, preferência do utilizador e em último caso, à aleatoriedade. No caso do contexto, tem de se ter em conta o contexto atual da conversa. Por exemplo, considerando o apoio pelo seu clube futebolístico, o utilizador utilizará um *emoji* de coração mas de cor afeta ao seu clube, em vez do *emoji* de coração regular e também mais popular. No caso da preferência do utilizador, deve-se ao facto de alguns dos autores dos textos darem preferência por um conjunto de *emojis* sobre outros. Por

fim temos o caso de utilização em que os utilizadores simplesmente selecionam o *emoji* de forma aleatória. Devido à seleção do *emoji*, em alguns casos enquadrar-se num destes três cenários, torna a tarefa de previsão muito mais complicada.

O tipo de escrita em que aplicamos os modelos computacionais também é importante para a tarefa em questão. Este ponto é reforçado pelos resultados obtidos usando a abordagem *baseline* para o nosso conjunto de dados, com resultados próximos de 0%. Estes resultados devem-se em grande parte aos termos usados nos textos do conjunto de dados não se encontrarem no léxico NRC. Desta forma é muito difícil atribuir uma emoção com esta abordagem, e quando é possível atribuir, é com base em alguns termos sem relação entre si. Ao classificar o conjunto de dados de Dosciatti, Ferreira e Paraiso (2015), com uma escrita mais objetiva o desempenho desta abordagem *baseline* é ligeiramente superior. Este aumento deve-se à ocorrência dos termos existentes nas notícias também estarem presentes no léxico NRC. Isto abre novamente a discussão sobre a localização do teor emocional dentro de um texto, sendo que os resultados obtidos demonstram que este se encontra em alguns termos singulares, mas principalmente em expressões que são usadas regularmente. Estes resultados introduzem a ideia que para além de serem expressões comuns, é muito provável serem expressões de calão muito utilizadas nas redes sociais.

Por fim, não podemos negar a influência que os *emojis* têm na emoção presente no texto. Ao utilizar o *emoji*, bem como procurar aquele conjunto restrito de termos altamente relacionados com as emoções, será o ponto de partida para desenvolver modelos que permitam efetuar a classificação de emoção presente num texto de forma correta.

## 6.5 Trabalho Futuro

O trabalho desenvolvido poderá sempre ser melhorado em diversos aspetos num futuro próximo. Como concluído, devido ao não balanceamento dos dados os resultados obtidos ficaram um pouco além do esperado, e como tal deverá combater-se inicialmente este aspeto. Desta forma, deverá extrair-se novos *tweets* sendo que estes deverão pertencer a todas as classes menos à classe de felicidade. A par deste ponto também deverá ser reduzido o número de *emojis* em estudo de forma a reduzir este conjunto. Uma forma de efetuar esta redução será eliminar variantes de *emojis* (e.g. 😍👉), e assim procurar combater os problemas de aleatoriedade que estes trazem. Ao efetuar este balanceamento, será desejável atingir uma divisão de classes próximo de 60% para a felicidade e 40% para as restantes classes, preservando assim um pouco a real distribuição dos dados. O conjunto de dados resultante do processo também deverá ser disponibilizado, para utilização futura noutros trabalhos semelhantes. Há, contudo, de o fazer de acordo com as políticas do *Twitter*.

Outro ponto será o melhoramento dos modelos em vigor. Nomeadamente o modelo LSTM, pois é aquele que apresenta mais hipóteses de escolha e maior flexibilidade na determinação da sua arquitetura. Devido à infinidade de combinações possíveis, a arquitetura desenvolvida para o modelo foi baseada na literatura existente. Como tal, uma análise exploratória de outras combinações com camadas, funções de ativação e outros parâmetros, podem levar a resultados ligeiramente superiores aos obtidos.

À imagem do que foi feito para o inglês e castelhano, num futuro próximo, seria interessante recriar a tarefa de *emoji prediction* para o português, num formato de avaliação conjunta (em inglês, *shared task*), aberta a outros participantes. Começaríamos pela disponibilização inicial de uma parte de um novo conjunto de dados, com os *emojis* a prever, de forma a dar a conhecer o formato dos dados e permitir aos participantes

validar e treinar os seus modelos. Numa fase mais avançada seria disponibilizado outra parte do conjunto, agora sem os *emojis*, sobre a qual os participantes teriam como objetivo usar os seus modelos para prever os *emojis* originalmente usados em cada texto e, num período de tempo limitado (por exemplo, uma semana), enviar os seus resultados. Os resultados seriam depois avaliados pelos organizadores, com base nas métricas comuns, discutidos num *workshop* e, idealmente, publicados, juntamente com uma descrição das várias soluções propostas para resolver o problema. No entanto, devido às leis de proteção de dados em vigor, o conjunto de dados teria de ser disponibilizado de forma a que sejam os próprios participantes a obter os textos da rede social e não serem os organizadores a fornecerem-nos diretamente. Como tal, será necessário o desenvolvimento de *scripts* adicionais, bem como a remoção de alguns campos que dizem respeito aos autores dos textos.

Por fim, há que explorar os dados obtidos de forma a efetuar comparações mais significativas. Nomeadamente a análise da influência dos *emojis*, para as duas variantes da língua portuguesa. Devido ao baixo número de textos de português europeu, esta experiência efetuada com o conjunto de dados atual, não obteria grande significado. Daí surge a necessidade, tal como no primeiro ponto, de aumentar este tipo de dados no conjunto final, para que as experiências levem a melhores resultados. Embora a maioria das publicações no Twitter não estejam geo-localizadas, poderíamos recorrer a outras heurísticas para identificar a origem de um conjunto significativo de publicações (e.g., origem do autor).

Outra experiência futura será efetuar uma nova análise e classificação de dados, mas agora tendo em conta o grau de expressividade da emoção. Um ponto não aprofundado é que existem vários tipos de emoção. A emoção retratada neste trabalho diz respeito à emoção momentânea do utilizador no texto. Um trabalho para o futuro será continuar esta classificação de emoções, mas tendo em conta os restantes tipos de emoção que existem. Para além desta experiência, podem-se explorar outros modelos de emoções categóricas para efetuar a classificação, como por exemplo as oito emoções de Plutchik (2001).

Todo este trabalho, engloba uma série de aspetos que podem complementar significativamente todas as experiências realizadas até agora. Como tal, espera-se que estas sejam concluídas em breve e os novos resultados disponibilizados na forma de, pelo menos, um novo artigo científico.

Esta página foi intencionalmente deixada em branco.

# Referências

- Agarwal, Apoorv et al. (2011). “Sentiment analysis of twitter data”. Em: *Proceedings of the workshop on languages in social media*. Association for Computational Linguistics, pp. 30–38.
- Alm, Cecilia Ovesdotter, Dan Roth e Richard Sproat (2005). “Emotions from text: machine learning for text-based emotion prediction”. Em: *Proceedings of the conference on human language technology and empirical methods in natural language processing*. Association for Computational Linguistics, pp. 579–586.
- Amati, Gianni e Cornelis Joost Van Rijsbergen (2002). “Probabilistic models of information retrieval based on measuring the divergence from randomness”. Em: *ACM Transactions on Information Systems (TOIS)* 20.4, pp. 357–389.
- Araújo, Matheus et al. (2013). “Métodos para análise de sentimentos no twitter”. Em: *Proceedings of the 19th Brazilian symposium on Multimedia and the Web (WebMedia'13)*.
- Arifin, Agus Zainal et al. (2014). “Emotion detection of tweets in indonesian language using non-negative matrix factorization”. Em: *International Journal of Intelligent Systems and Applications* 6.9, p. 54.
- Barbier, Geoffrey e Huan Liu (2011). “Data mining in social media”. Em: *Social network data analytics*. Springer, pp. 327–352.
- Barbieri, Francesco, Miguel Ballesteros e Horacio Saggion (2017). “Are Emojis Predictable?” Em: *arXiv preprint arXiv:1702.07285*.
- Barbieri, Francesco et al. (2018a). “Interpretable Emoji Prediction via Label-Wise Attention LSTMs”. Em: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 4766–4771.
- Barbieri, Francesco et al. (2018b). “SemEval 2018 Task 2: Multilingual Emoji Prediction”. Em: *Proceedings of The 12th International Workshop on Semantic Evaluation*, pp. 24–33.
- Binali, Haji, Chen Wu e Vidyasagar Potdar (2010). “Computational approaches for emotion detection in text”. Em: *Proceedings of the IEEE international conference on digital ecosystems and technologies (DEST 2010)*. IEEE, pp. 172–177.
- Bradley, Margaret M e Peter J Lang (1999). *Affective norms for English words (ANEW): Instruction manual and affective ratings*. Rel. téc. Citeseer.
- Buechel, Sven e Udo Hahn (2016). “Emotion Analysis as a Regression Problem-Dimensional Models and Their Implications on Emotion Representation and Metrical Evaluation.” Em: *ECAI*, pp. 1114–1122.
- (2018a). “Emotion Representation Mapping for Automatic Lexicon Construction (Mostly) Performs on Human Level”. Em: *COLING 2018 — Proceedings of the 27th International Conference on Computational Linguistics*. Vol. 1, technical papers. Santa Fe, New-Mexico, USA, August 20–26, 2018.
- (2018b). “Representation Mapping: A Novel Approach to Generate High-Quality Multi-Lingual Emotion Lexicons”. Em: *LREC 2018 — Proceedings of the 11th*

- International Conference on Language Resources and Evaluation*. Miyazaki, Japan, May 7–12, 2018, pp. 184–191.
- Cambria, Erik (2016). “Affective computing and sentiment analysis”. Em: *IEEE Intelligent Systems* 31.2, pp. 102–107.
- Cambria, Erik et al. (2010). “Senticnet: A publicly available semantic resource for opinion mining.” Em: *AAAI fall symposium: commonsense knowledge*. Vol. 10. 0.
- Caridakis, George, Kostas Karpouzis e Stefanos Kollias (2008). “User and context adaptive neural networks for emotion recognition”. Em: *Neurocomputing* 71.13-15, pp. 2553–2562.
- Carvalho, Paula e Mário J Silva (2015). “SentiLex-PT: Principais características e potencialidades”. Em: *Oslo Studies in Language* 7.1.
- Carvalho, Paula et al. (2009). “Clues for detecting irony in user-generated contents: oh...!! it’s so easy;-”. Em: *Proceedings of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*. ACM, pp. 53–56.
- Chawla, Nitesh V et al. (2002). “SMOTE: synthetic minority over-sampling technique”. Em: *Journal of artificial intelligence research* 16, pp. 321–357.
- Costa, Sara Filipa Oliveira (2012). “Adaptação e teste de uma base lexical de palavras emocionais para o português europeu:(EMOTAIX. PT)”. Em:
- Danisman, Taner e Adil Alpkocak (2008). “Feeler: Emotion classification of text using vector space model”. Em: *AISB 2008 Convention Communication, Interaction and Social Intelligence*. Vol. 1, p. 53.
- De Silva, Liyanage C, Tsutomu Miyasato e Ryohei Nakatsu (1997). “Facial emotion recognition using multi-modal information”. Em: *Information, Communications and Signal Processing, 1997. ICICS., Proceedings of 1997 International Conference on*. Vol. 1. IEEE, pp. 397–401.
- Ding, Xiaowen, Bing Liu e Philip S Yu (2008). “A holistic lexicon-based approach to opinion mining”. Em: *Proceedings of the 2008 international conference on web search and data mining*. ACM, pp. 231–240.
- Dosciatti, Mariza Miola, Lohann Paterno Coutinho Ferreira e Emerson Cabrera Paraiso (2015). “Anotando um Corpus de Notícias para a Análise de Sentimento: um Relato de Experiência”. Em: *Anais do X Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana*. SBC, pp. 121–130.
- Dosciatti, Mariza Miola, LPC Ferreira e EC Paraiso (2013). “Identificando emoções em textos em português do brasil usando máquina de vetores de suporte em solução multiclasse”. Em: *ENIAC-Encontro Nacional de Inteligência Artificial e Computacional. Fortaleza, Brasil*.
- Driscoll, Kevin e Shawn Walker (2014). “Big data, big questions— working within a black box: Transparency in the collection and production of big twitter data”. Em: *International Journal of Communication* 8, p. 20.
- Duarte, Luís, Luís Macedo e Hugo Gonçalo Oliveira (2019). “Exploring Emojis for Emotion Recognition in Portuguese Text”. Em: *Proceedings of 19th EPIA Conference on Artificial Intelligence, 3–6 September, 2019, Vila Real, Portugal*. LNCS, In press.
- Ekman, Paul (1971). “Universals and cultural differences in facial expressions of emotion.” Em: *Nebraska symposium on motivation*. University of Nebraska Press.
- (1993). “Facial expression and emotion.” Em: *American psychologist* 48.4, p. 384.
- (1999). “Basic emotions”. Em: *Handbook of cognition and emotion*, pp. 45–60.
- Ekman, Paul e Wallace V Friesen (1971). “Constants across cultures in the face and emotion.” Em: *Journal of personality and social psychology* 17.2, p. 124.
- Esuli, Andrea e Fabrizio Sebastiani (2007). “SentiWordNet: a high-coverage lexical resource for opinion mining”. Em: *Evaluation* 17, pp. 1–26.

- Felbo, Bjarke et al. (2017). “Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm”. Em: *arXiv preprint arXiv:1708.00524*.
- Fernandes, J. et al. (2019). “A Human-in-the-Loop Cyber-Physical Approach for Students Performance Assessment”. Em: *Proceedings of the Fourth International Workshop on Social Sensing*. SocialSense’19. Montreal, QC, Canada: ACM, pp. 36–42. ISBN: 978-1-4503-6706-6. DOI: 10.1145/3313294.3313387. URL: <http://doi.acm.org/10.1145/3313294.3313387>.
- Freitas, Cláudia (2013). “Sobre a construção de um léxico da afetividade para o processamento computacional do português”. Em: *Revista Brasileira de Linguística Aplicada* 13.4.
- Frijda, Nico H (2004). “Emotions and action”. Em: *Feelings and emotions: The Amsterdam symposium*, pp. 158–173.
- Gomes, Ricardo (2005). “A tradição do ator entre oriente e ocidente”. Em: *Sala Preta* 5, pp. 39–46.
- Gunning, David (2017). “Explainable artificial intelligence (xai)”. Em: *Defense Advanced Research Projects Agency (DARPA), nd Web* 2.
- Gustafsson, Viktor (2017). “Replacing words with emojis and its effect on reading time”. Em: *USCCS 2017*, p. 73.
- Hartigan, John A (1975). “Clustering algorithms”. Em:
- Hasan, Maryam, Emmanuel Agu e Elke Rundensteiner (2014). “Using hashtags as labels for supervised learning of emotions in twitter messages”. Em: *ACM SIGKDD Workshop on Health Informatics, New York, USA*.
- Hussien, W et al. (2016). “Are emoticons good enough to train emotion classifiers of arabic tweets?” Em: *CSIT. IEEE*, pp. 1–6.
- James, William (1884). “What is an emotion?” Em: *Mind* 9.34, pp. 188–205.
- Java, Akshay et al. (2007). “Why we twitter: understanding microblogging usage and communities”. Em: *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*. ACM, pp. 56–65.
- Kitchenham, Barbara (1998). “A procedure for analyzing unbalanced datasets”. Em: *IEEE transactions on Software Engineering* 24.4, pp. 278–301.
- Kleinginna, Paul R e Anne M Kleinginna (1981). “A categorized list of emotion definitions, with suggestions for a consensual definition”. Em: *Motivation and emotion* 5.4, pp. 345–379.
- Kubat, Miroslav, Stan Matwin et al. (1997). “Addressing the curse of imbalanced training sets: one-sided selection”. Em: *Icml*. Vol. 97. Nashville, USA, pp. 179–186.
- Kunneman, Florian, Christine Liebrecht e Antal van den Bosch (2014). “The (un) predictability of emotional hashtags in twitter”. Em: *Proceedings of the 5th Workshop on Language Analysis for Social Media (LASM)*, pp. 26–34.
- Lazarus, Richard S (1991). “Progress on a cognitive-motivational-relational theory of emotion.” Em: *American psychologist* 46.8, p. 819.
- Lisetti, CL (1998). *Affective computing*.
- Liu, Xu-Ying, Jianxin Wu e Zhi-Hua Zhou (2008). “Exploratory undersampling for class-imbalance learning”. Em: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 39.2, pp. 539–550.
- Ma, Chunling, Helmut Prendinger e Mitsuru Ishizuka (2005). “Emotion estimation and reasoning based on affective textual interaction”. Em: *International Conference on Affective Computing and Intelligent Interaction*. Springer, pp. 622–628.
- Martin, James H e Daniel Jurafsky (2009). *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. Pearson/Prentice Hall Upper Saddle River.

- Martinazzo, Barbara, Mariza Miola Dosciatti e Emerson Cabrera Paraiso (2011). “Identifying emotions in short texts for brazilian portuguese”. Em: *IV International Workshop on Web and Text Intelligence (WTI 2012)*, p. 16.
- Mikolov, Tomas et al. (2013). “Efficient estimation of word representations in vector space”. Em: *arXiv preprint arXiv:1301.3781*.
- Mohammad, Saif M e Svetlana Kiritchenko (2015). “Using hashtags to capture fine emotion categories from tweets”. Em: *Computational Intelligence* 31.2, pp. 301–326.
- Mohammad, Saif M e Peter D Turney (2013). “Nrc emotion lexicon”. Em: *National Research Council, Canada*.
- Montefinese, Maria et al. (2014). “The adaptation of the affective norms for English words (ANEW) for Italian”. Em: *Behavior research methods* 46.3, pp. 887–903.
- Mozetič, Igor, Miha Grčar e Jasmina Smailović (2016). “Multilingual Twitter sentiment classification: The role of human annotators”. Em: *PloS one* 11.5, e0155036.
- Nakov, Preslav et al. (jun. de 2013). “SemEval-2013 Task 2: Sentiment Analysis in Twitter”. Em: *Second Joint Conference on Lexical and Computational Semantics (\*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*. Atlanta, Georgia, USA: Association for Computational Linguistics, pp. 312–320. URL: <https://www.aclweb.org/anthology/S13-2052>.
- Nascimento, Paula et al. (2012). “Análise de sentimento de tweets com foco em notícias”. Em: *Brazilian Workshop on Social Network Analysis and Mining*.
- Paiva, Vera Lúcia Menezes de Oliveira (2016). “THE LANGUAGE OF EMOJIS”. Em: *Trabalhos em Linguística Aplicada* 55.2, pp. 379–401.
- Pak, Alexander e Patrick Paroubek (2010). “Twitter as a corpus for sentiment analysis and opinion mining.” Em: *LREc*. Vol. 10. 2010, pp. 1320–1326.
- Pennebaker, James W, Martha E Francis e Roger J Booth (2001). “Linguistic inquiry and word count: LIWC 2001”. Em: *Mahway: Lawrence Erlbaum Associates* 71.2001, p. 2001.
- Perrin, Andrew (2015). “Social media usage”. Em: *Pew research center*, pp. 52–68.
- Plutchik, Robert (2001). “The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice”. Em: *American scientist* 89.4, pp. 344–350.
- Redondo, Jaime et al. (2007). “The Spanish adaptation of ANEW (affective norms for English words)”. Em: *Behavior research methods* 39.3, pp. 600–605.
- Reisenzein, Rainer e Hannelore Weber (2009). “Personality and emotion”. Em: *The Cambridge handbook of personality psychology*, pp. 54–71.
- Rodrigues, David et al. (2018). “Lisbon Emoji and Emoticon Database (LEED): Norms for emoji and emoticons in seven evaluative dimensions”. Em: *Behavior research methods* 50.1, pp. 392–405.
- Rosenthal, Sara, Noura Farra e Preslav Nakov (ago. de 2017). “SemEval-2017 Task 4: Sentiment Analysis in Twitter”. Em: *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Vancouver, Canada: Association for Computational Linguistics, pp. 502–518. DOI: 10.18653/v1/S17-2088. URL: <https://www.aclweb.org/anthology/S17-2088>.
- Russell, James A (1979). “Affective space is bipolar.” Em: *Journal of personality and social psychology* 37.3, p. 345.
- (1991). “Culture and the categorization of emotions.” Em: *Psychological bulletin* 110.3, p. 426.
- Russell, James A, Maria Lewicka e Toomas Niit (1989). “A cross-cultural study of a circumplex model of affect.” Em: *Journal of personality and social psychology* 57.5, p. 848.

- Schmidtke, David S et al. (2014). “ANGST: Affective norms for German sentiment terms, derived from the affective norms for English words”. Em: *Behavior research methods* 46.4, pp. 1108–1118.
- Schwartz, H Andrew et al. (2013). “Personality, gender, and age in the language of social media: The open-vocabulary approach”. Em: *PloS one* 8.9, e73791.
- Shivhare, Shiv Naresh e Saritha Khethawat (2012). “Emotion detection from text”. Em: *arXiv preprint arXiv:1205.4944*.
- Silva, Zuzanna Bułat (2012). “Saudade—a key Portuguese emotion”. Em: *Emotion Review* 4.2, pp. 203–211.
- Smith, Timothy W et al. (2004). “Hostility, anger, aggressiveness, and coronary heart disease: An interpersonal perspective on personality, emotion, and health”. Em: *Journal of personality* 72.6, pp. 1217–1270.
- Soares, Ana Paula et al. (2012). “The adaptation of the Affective Norms for English words (ANEW) for European Portuguese”. Em: *Behavior research methods* 44.1, pp. 256–269.
- Souza, Marlo e Renata Vieira (2012). “Sentiment analysis on twitter data for portuguese language”. Em: *International Conference on Computational Processing of the Portuguese Language*. Springer, pp. 241–247.
- Sreeja, PS e GS Mahalakshmi (2016). “Comparison of probabilistic corpus based method and vector space model for emotion recognition from poems”. Em: *Asian Journal of Information Technology* 15.5, pp. 908–915.
- (jan. de 2017). “Emotion Models: A Review”. Em: *International Journal of Control Theory and Applications* 10, pp. 651–657.
- Sreeja, PS e GS Mahalakshmi (2015). “Applying vector space model for poetic emotion recognition”. Em: *Advances in Natural and Applied Sciences* 9.6 SE, pp. 486–491.
- Suttles, Jared e Nancy Ide (2013). “Distant supervision for emotion classification with discrete binary values”. Em: *International Conference on Intelligent Text Processing and Computational Linguistics*. Springer, pp. 121–136.
- Tato, Ange, Roger Nkambou e Claude Frasson (2018). “Predicting Emotions From Multimodal Users’ Data”. Em: *Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization*. ACM, pp. 369–370.
- Tausczik, Yla R e James W Pennebaker (2010). “The psychological meaning of words: LIWC and computerized text analysis methods”. Em: *Journal of language and social psychology* 29.1, pp. 24–54.
- Thayer, Robert E (1990). *The biopsychology of mood and arousal*. Oxford University Press.
- Van Hee, Cynthia, Els Lefever e Véronique Hoste (2018). “Semeval-2018 task 3: Irony detection in english tweets”. Em: *Proceedings of The 12th International Workshop on Semantic Evaluation*, pp. 39–50.
- Vanin, Aline A et al. (2013). “Some clues on irony detection in tweets”. Em: *Proceedings of the 22nd International Conference on World Wide Web*. ACM, pp. 635–636.
- Wang, Wenbo et al. (2012). “Harnessing twitter”big data”for automatic emotion identification”. Em: *Privacy, Security, Risk and Trust (PASSAT), 2012 International Conference on and 2012 International Confernece on Social Computing (SocialCom)*. IEEE, pp. 587–592.
- Wang, Yequan, Minlie Huang, Li Zhao et al. (2016). “Attention-based LSTM for aspect-level sentiment classification”. Em: *Proceedings of the 2016 conference on empirical methods in natural language processing*, pp. 606–615.
- Wood, I e Sebastian Ruder (2016). “Emoji as emotion tags for tweets”. Em: *Proceedings of the Emotion and Sentiment Analysis Workshop LREC2016, Portorož, Slovenia*, pp. 76–79.
- Wu, David YH (1982). “Psychotherapy and emotion in traditional Chinese medicine”. Em: *Cultural conceptions of mental health and therapy*. Springer, pp. 285–301.