



**IPL**

**escola superior de tecnologia e gestão**  
instituto politécnico de leiria

Mestrado em Ciência de Dados

IMPLEMENTAÇÃO DE FERRAMENTA DE  
AVALIAÇÃO DE SATISFAÇÃO DE MUNICÍPIOS  
ATRAVÉS DAS REDES SOCIAIS

LUÍS MIGUEL DA SILVA BARREIRO

Leiria, Setembro de 2023





**IPL**

**escola superior de tecnologia e gestão**  
instituto politécnico de leiria

Mestrado em Ciência de Dados

IMPLEMENTAÇÃO DE FERRAMENTA DE  
AVALIAÇÃO DE SATISFAÇÃO DE MUNÍCIPES  
ATRAVÉS DAS REDES SOCIAIS

LUÍS MIGUEL DA SILVA BARREIRO

Trabalho de projeto realizado sob orientação do Professor Nuno Filipe Pereira Salvador ([nuno.salvador@ipleiria.pt](mailto:nuno.salvador@ipleiria.pt)) e Professora Doutora Maria Beatriz Guerra da Piedade ([beatriz.piedade@ipleiria.pt](mailto:beatriz.piedade@ipleiria.pt)).

Leiria, Setembro de 2023



## AGRADECIMENTOS

---

A conclusão de um projeto desta dimensão não é algo que se consiga fazer totalmente sozinho. O apoio emocional, logístico e motivacional é de extrema importância. Assim, os primeiros agradecimentos são para a minha família, e em especial, para a minha companheira Marisa. Com muito amor, apoiaram-me a 100% para que conseguisse levar este projeto a bom termo.

Os próximos agradecimentos vão para o professor Nuno Salvador, pela proposta e idealização do funcionamento da ferramenta desenvolvida, assim como o acompanhamento que deu ao longo de todo o ciclo de desenvolvimento desta e do presente documento. Agradeço-lhe a ele e também ao Município de Pombal a disponibilização das plataformas tecnológicas dos serviços municipais.

De seguida, agradeço à professora Doutora Maria Beatriz Piedade, que com a sua experiência na orientação de projetos, guiou-me no desenvolvimento deste trabalho e, em especial, deste relatório. Os seus comentários foram sempre muito construtivos nas revisões deste documento.

Por último, mas não menos importante, quero também agradecer a todos os meus professores e colegas, tanto do mestrado, como da licenciatura e do trabalho no Observador. Foram eles que me ajudaram a consolidar os conhecimentos, que tornaram possível realizar grandes projetos, que tocam em diferentes áreas das ciências dos dados e dos computadores, e sempre com humor e boa disposição.



## RESUMO

---

Este relatório descreve o processo de desenvolvimento de uma ferramenta para análise de sentimentos em publicações de páginas públicas do Facebook.

Para isso, utiliza a *Application programming interface* (API) fornecida pelo Facebook para recolha e carregamento de dados. Após o pré processamento de dados, onde é efetuada a limpeza dos mesmos, utiliza um modelo de aprendizagem supervisionado, desenvolvido e avaliado de acordo com os objetivos do projeto. Este modelo é depois integrado na ferramenta desenvolvida. Esta ferramenta funciona como uma aplicação *web*, onde é possível a qualquer utilizador, com uma conta no Facebook e privilégios de administração de alguma página, visualizar graficamente e por ordem cronológica o volume de interações que a sua página tem. Para cada publicação, são ainda disponibilizados detalhes como número de partilhas, interações (emojis), comentários, e para cada comentário, a sua classificação, positiva ou negativa.

Espera-se que com esta ferramenta, os responsáveis pela gestão de páginas do Facebook possam criar melhores conteúdos que permitam às organizações crescer de forma saudável e sustentável. Especificamente no caso do Município de Pombal, espera-se que a sua utilização permita à câmara estar mais perto dos seus munícipes e compreender as diferentes opiniões e sentimentos em relação à administração do município, assim como apoiar a tomada de decisões.

**Palavras chave**— *Natural language processing, Sentiment analysis, Machine learning, Data mining, Text mining, Artificial intelligence*



## ABSTRACT

---

This report describes the development process of a tool for sentiment analysis on public Facebook page posts.

To achieve this, it utilizes the Facebook `\acrfull{api}` for data collection and retrieval. After data preprocessing, including data cleaning, it employs a supervised learning model that was developed and evaluated according to the project's objectives. This model is then integrated into the developed tool. This tool functions as a web application, where any user with a Facebook account and administrative privileges for a page can visually and chronologically view the volume of interactions their page receives. For each post, details such as the number of shares, interactions (emojis), comments, and, for each comment, its sentiment classification as positive or negative, are also provided.

It is expected that with this tool, Facebook page administrators will be able to create better content that allows organizations to grow in a healthy and sustainable manner. Specifically, in the case of the Municipality of Pombal, it is expected that its use will enable the municipal government to be closer to its residents and understand different opinions and sentiments regarding municipal administration, as well as support decision-making.

**Keywords**— *Natural language processing, Sentiment analysis, Machine learning, Data mining, Text mining, Artificial intelligence*



# ÍNDICE

---

Agradecimentos	i
Resumo	iii
Abstract	v
Índice	vii
Lista de Figuras	ix
Lista de Tabelas	xi
Lista de Abreviaturas	xiii
1 Introdução	1
1.1 Objetivos do trabalho . . . . .	1
1.2 Estrutura do documento . . . . .	2
2 Trabalho Relacionado	5
2.1 Breve história da <b>AI</b> . . . . .	5
2.2 <i>Text mining</i> . . . . .	6
2.3 <i>Natural language processing</i> (NLP) . . . . .	7
2.4 <i>Sentiment analysis</i> . . . . .	8
3 Desenvolvimento	11
3.1 Metodologias aplicadas . . . . .	11
3.2 Aplicação da Metodologia <b>CRISP-DM</b> para criação do modelo de análise de sentimentos . . . . .	11
3.2.1 Compreensão do negócio . . . . .	12
3.2.2 Compreensão dos dados . . . . .	13
3.2.3 Preparação dos dados . . . . .	14
3.2.4 Modelação . . . . .	15
3.2.5 Avaliação . . . . .	18
3.2.6 Implementação . . . . .	21
3.2.7 Monitorização . . . . .	22
3.3 Metodologia ágil Scrum, desenvolvimento da aplicação <i>web</i> . . . . .	22
3.3.1 Escolha de tecnologias . . . . .	23
3.3.2 Definição do <i>backlog</i> . . . . .	24
3.3.3 Planeamento dos <i>Sprints</i> : . . . . .	27
3.3.4 Desenvolvimento e Entrega: . . . . .	28

## ÍNDICE

4	Conclusões	31
4.1	Trabalho futuro . . . . .	32
	Bibliografia	33
 <i>Apêndices</i>		
	Declaração	47

## LISTA DE FIGURAS

---

Figura 1	Evolução do número de artigos de <b>AI</b> publicados por ano Kaya, 2023 . . . . .	6
Figura 2	Evolução do crescimento do número de artigos de <b>AI</b> por ano, segmentados por área Kaya, 2023 . . . . .	6
Figura 3	Diagrama LB «Sentiment analysis in Facebook and its application to e-learning» . . . . .	9
Figura 4	Metodologia CRISP-DM . . . . .	12
Figura 5	Avaliação do modelo de aprendizagem supervisionada, curva <b>ROC</b> . . . . .	19
Figura 6	Matriz de confusão do modelo de aprendizagem supervisionada	19
Figura 7	Avaliação do modelo baseado no Léxico, curva <b>ROC</b> . . . . .	20
Figura 8	Matriz de confusão do modelo baseado no léxico . . . . .	20
Figura 9	<b>CLI</b> para análise de sentimentos . . . . .	21
Figura 10	Relatório com médias da análise de sentimentos e <i>engagement</i>	22
Figura 11	Arquitetura dos serviços . . . . .	24
Figura 12	Gráfico temporal com os <i>posts</i> e métrica <i>engagement</i> . . . . .	43
Figura 13	Relatório com médias da análise de sentimentos e <i>engagement</i>	44
Figura 14	Página de configuração de emails para alertas . . . . .	44
Figura 15	Componente de detalhes de publicação . . . . .	45



## LISTA DE TABELAS

---

Tabela 1	Avaliação do modelo de aprendizagem supervisionada, <i>Precision</i> , <i>Recall</i> e <i>F1 score</i> . . . . .	18
Tabela 2	Avaliação do modelo de aprendizagem supervisionada, <b>ROC AUC score</b> . . . . .	18
Tabela 3	Avaliação do modelo de aprendizagem supervisionada, <i>accuracy</i>	18
Tabela 4	Avaliação do modelo baseado no Léxico, <i>Precision</i> , <i>Recall</i> e <i>F1 score</i> . . . . .	19
Tabela 5	Avaliação do modelo baseado no Léxico, <b>ROC AUC score</b> . . . . .	19
Tabela 6	Avaliação do modelo baseado no Léxico, <i>accuracy</i> . . . . .	20
Tabela 7	Planeamento dos <i>sprints</i> . . . . .	28

LISTA DE TABELAS

## LISTA DE ABREVIATURAS

---

AI	<i>Artificial intelligence.</i>
API	<i>Application programming interface.</i>
AUC	<i>Area Under the Curve.</i>
BD	<i>Big Data.</i>
BERT	<i>Bidirectional Encoder Representations from Transformers.</i>
BN	<i>Bayesian networks.</i>
BOW	<i>Bag of words.</i>
CLI	<i>Command line interface.</i>
CRISP-DM	<i>Cross Industry Standard Process for Data Mining.</i>
DL	<i>Deep learning.</i>
DM	<i>Data mining.</i>
GOPS	Google Play Store.
HTTP	<i>Hypertext Transfer Protocol.</i>
IJIM	<i>International Journal of Innovation Management.</i>
JSON	<i>JavaScript Object Notation.</i>
LB	<i>Lexicon-based.</i>
LeIA	Léxico para Inferência Adaptada.
LSTM	<i>Long short-term memory.</i>
ML	<i>Machine learning.</i>

Lista de Abreviaturas

NER	<i>Named entity recognition.</i>
NLP	<i>Natural language processing.</i>
OAuth2	<i>Open Authorization 2.</i>
POS	<i>Part of speech.</i>
POST	<i>Part of speech tagging.</i>
PR	<i>Pattern recognition.</i>
RAM	<i>Random access memory.</i>
RBS	<i>Rule-based system.</i>
REST	<i>Representational State Transfer.</i>
RNN	<i>Recurrent neural network.</i>
ROC	<i>Receiver Operating Characteristic.</i>
RS	Redes sociais.
SA	<i>Sentiment analysis.</i>
SN	<i>Semantic networks.</i>
SQL	<i>Structured Query Language.</i>
TLS	<i>Transport Layer Security.</i>
TM	<i>Text mining.</i>
UI	<i>User interface.</i>
UX	<i>User experience.</i>
VADER	<i>Valence Aware Dictionary and Sentiment Reasoner.</i>

## INTRODUÇÃO

---

Com a popularidade das redes sociais, cada vez mais utilizadas pelos cidadãos para expressarem as suas opiniões sobre os serviços oferecidos pelas empresas, sejam públicas ou privadas, o valor dos dados contidos nas redes sociais torna-se cada vez maior. A possibilidade de extrair os dados e transformá-los em informação relevante torna-se um objetivo para muitas empresas, organizações, partidos políticos, instituições, entre outros. Estas entidades procuram ferramentas que permitam estar mais próximas das suas audiências, servindo-as melhor e com mais eficiência.

As perceções da audiência de uma entidade é sempre um assunto muito importante. Conhecer e analisar os pensamentos e sentimentos, a nosso respeito, daqueles com quem nos cruzamos mais frequentemente, é algo importante e que procuramos inconscientemente. A busca por aprovação é algo natural ao ser humano. Observamos comportamentos, percebemos através daquilo que é dito, mas também naquilo que não o é, os pensamentos e sentimentos dos outros em relação ao que dizemos, fazemos e acreditamos.

Nas sociedades, o conhecimento e poder político são muito importantes e uma parte crucial desse poder, pelo menos nas democracias, é conhecer a população. Atualmente, vivemos uma grande parte do dia-a-dia no mundo digital, em especial nas redes sociais. É assim natural que os representantes políticos queiram ferramentas capazes de extrair e transformar os dados das suas páginas nas redes sociais, com o objetivo de transformá-los em conhecimento valioso. Este conhecimento pode permitir a melhoria da relação e proximidade com a população, assim como perceber se as iniciativas autárquicas correspondem às expetativas dos cidadãos.

### 1.1 OBJETIVOS DO TRABALHO

Este relatório apresenta o trabalho de conceção, desenvolvimento e implementação de uma ferramenta de análise de sentimentos através das redes sociais, mais especificamente o Facebook. O projeto foi proposto pela Câmara Municipal de Pombal como resposta à necessidade crescente de compreender a opinião e satisfação dos munícipes acerca dos serviços, iniciativas, eventos e políticas públicas realizadas pela autarquia.

O projeto tem como objetivo desenvolver uma ferramenta, sob a forma de uma aplicação *web*, que permita extrair os dados do Facebook das páginas públicas geridas pelo utilizador desta ferramenta. Os dados podem então ser transformados em informação relevante promotora de *insights* que permitam melhorar a interação com a audiência, assim como suportar a tomada de decisões.

Para atingir este objetivo, foram estudadas, analisadas e aplicadas diversas técnicas de *Artificial intelligence* (AI), tais como, *Sentiment analysis* (SA), *Machine learning* (ML), *Data mining* (DM), *Text mining* (TM) e *Natural language processing* (NLP), em especial no contexto de redes sociais. A implementação da ferramenta também envolveu trabalhar com outras tecnologias, desde a utilização da *Application programming interface* (API) da *Google Play Store* (GOPS) e do Facebook, virtualização de sistemas com *containers* Docker, gestão de *web proxys*, desenvolvimento *web* e bases de dados.

Este relatório descreve todo o processo de desenvolvimento da ferramenta, incluindo as metodologias aplicadas, arquitetura, tecnologias escolhidas e a apresentação dos resultados obtidos. Além disso, são apresentadas as conclusões subjacentes e sugestões para trabalho futuro. A implementação desta ferramenta no Município de Pombal pode ser uma mais valia para a autarquia, na procura contínua de uma melhor comunicação com os seus cidadãos, compreender os seus sentimentos em relação aos serviços municipais e obter *insights* para ações futuras.

## 1.2 ESTRUTURA DO DOCUMENTO

Este documento encontra-se dividido em 4 capítulos. No capítulo 1, é efetuada uma introdução à origem do projeto e aos requisitos para o seu desenvolvimento, assim como uma descrição da estrutura do presente documento.

No capítulo 2, intitulado "Trabalho relacionado", é apresentada uma breve revisão do trabalho associado ao tema da *Sentiment analysis*. O capítulo inicia com uma introdução ao tema da *Artificial intelligence*. São depois apresentados alguns conceitos fundamentais do *Text mining* e *Natural language processing*. Termina com uma descrição do conjunto de problemas que a *Sentiment analysis* tenta resolver, e quais são os métodos, técnicas e algoritmos mais utilizados na área à data deste projeto.

O capítulo 3, intitulado "Desenvolvimento", encontra-se dividido em três secções e descreve o processo de desenvolvimento da ferramenta. Na primeira secção, são apresentadas as metodologias aplicadas ao desenvolvimento do projeto. Na segunda e terceira secções, são descritos os desenvolvimentos do modelo de *Machine learning* e da aplicação *web*, seguindo as recomendações das metodologias adotadas, a

*Cross Industry Standard Process for Data Mining* (CRISP-DM) e a metodologia ágil Scrum.

O 4 e último capítulo abrange a conclusão, a apresentação e avaliação dos resultados obtidos, a identificação dos desafios e problemas encontrados, bem como a exploração de opções para trabalhos futuros.



## TRABALHO RELACIONADO

---

Muito do trabalho relacionado com a *Sentiment analysis* envolve áreas como o *Text mining*, *Machine learning* e *Natural language processing*. De seguida são revistos alguns dos conceitos fundamentais destas três áreas pertencentes à família da *Artificial intelligence*

### 2.1 BREVE HISTÓRIA DA AI

A AI existe há mais de seis décadas, tendo no entanto despertado um grande interesse nos últimos anos. O aumento da potência de computação e as tecnologias de *Big Data* (BD) vieram impulsionar o desenvolvimento das tecnologias associadas à AI de uma forma exponencial.

Este desenvolvimento da AI tem originado muitos comentários controversos. Por um lado, a CEO da IBM, Ginni Rometty, defende que as tecnologias de AI são “tecnologias para aumentar a inteligência humana” e que podemos esperar uma parceria entre humanos e máquinas que nos tornará melhores. Por outro lado, Stephen Hawking alertou que o desenvolvimento de uma AI completa poderia ameaçar a existência da raça humana, e Bill Gates também expressou preocupações sobre a ameaça que a AI representa. Estas diferentes opiniões de especialistas e líderes chamam a atenção para a necessidade de investigar como os seres humanos podem conviver com a AI e como minimizar os possíveis impactos negativos desta tecnologia.

Duan et al., 2019 relata que umas das primeiras técnicas usadas nos sistemas de AI foi a *Rule-based system* (RBS). O primeiro artigo, Lu e Mooney, 1989, publicado no *International Journal of Innovation Management* (IJIM) sobre AI, foi sobre RBS. No mesmo jornal, a primeira menção de um sistema de AI em produção refere um RBS, onde Bowonder e Miyake, 1992 discutem a gestão da informação na Nippon Steel Corporation do Japão. Até o ano 2000, quase todos os sistemas de AI utilizavam RBS, uma abordagem ainda muito utilizada atualmente. No entanto, “técnicas” como *Semantic networks* (SN), *Bayesian networks* (BN), TM, DL, NLP, entre outras, foram desenvolvidas, obtendo melhores resultados consoante o âmbito dos projetos.

Kaya, 2023 dá alguns *insights* sobre as tendências na área de **AI**, de destacar as figuras 1 e 2, onde está representado o crescimento expressivo do número de artigos escritos por ano, tendo duplicado nos últimos dez anos, em especial os relacionados com *Pattern recognition* (PR) e ML.

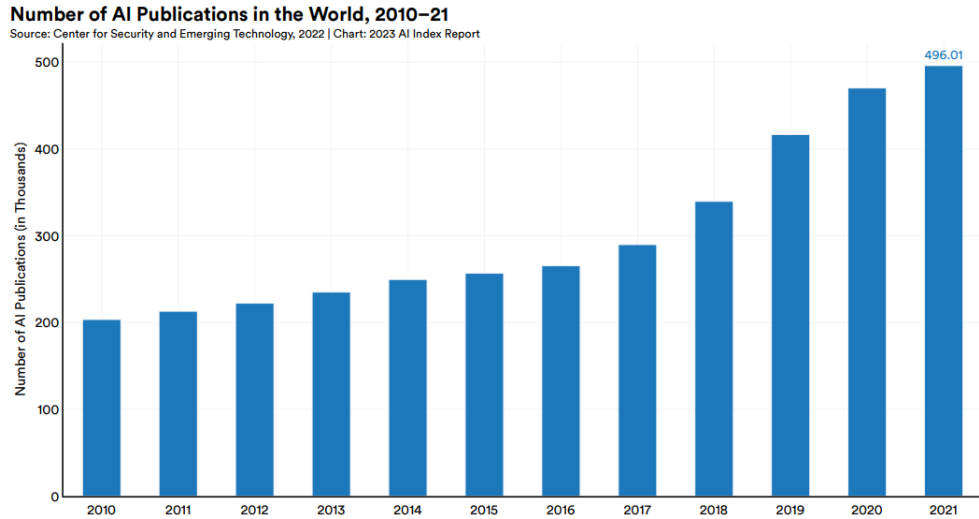


Figura 1: Evolução do número de artigos de **AI** publicados por ano Kaya, 2023

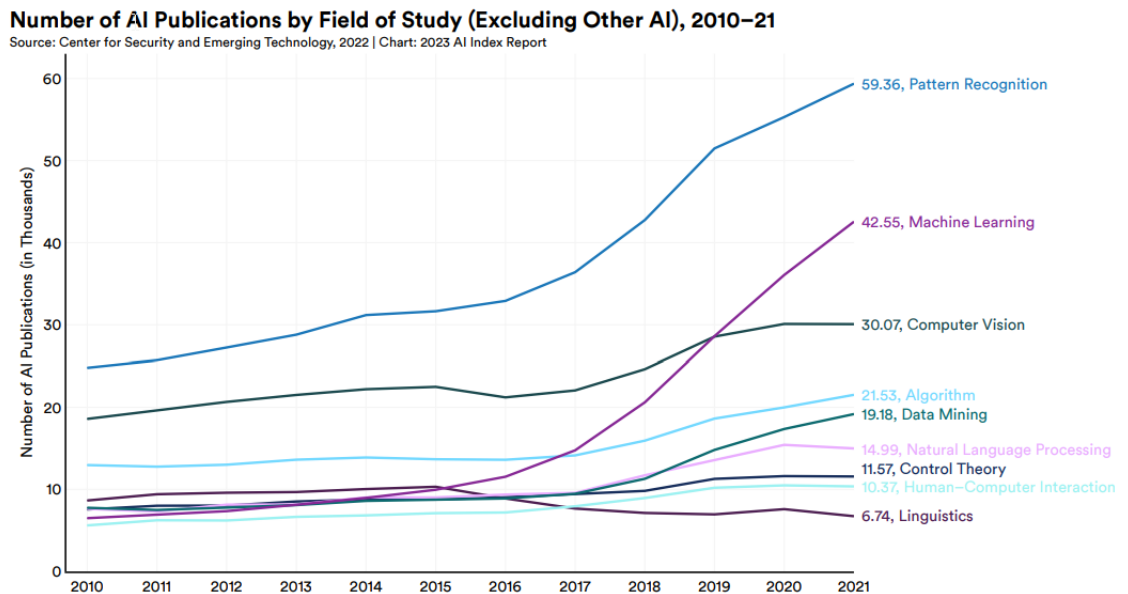


Figura 2: Evolução do crescimento do número de artigos de **AI** por ano, segmentados por área Kaya, 2023

## 2.2 *TEXT MINING*

Segundo Hearst, 2003, **TM** é um processo para descobrir novas e previamente desconhecidas informações através da extração e análise de dados textuais de várias fontes. Envolve a extração automática de informações de recursos escritos

e a ligação destes pedaços extraídos para formar novos conhecimentos, factos ou hipóteses, que podem ser posteriormente explorados. Ao contrário das pesquisas na *web* tradicionais, que procuram informações conhecidas, o **TM** procura descobrir *insights* ou conhecimentos novos que ainda não foram documentados.

### 2.3 NATURAL LANGUAGE PROCESSING (NLP)

**NLP** é um campo da **AI** que se concentra na linguagem humana. O objetivo principal do **NLP** é compreender, interpretar e gerar texto ou fala da mesma maneira que os seres humanos. Isso envolve uma série de tarefas, como reconhecimento de fala, tradução automática, **SA**, geração de texto, sumarização automática, entre outras. O **NLP** utiliza uma combinação de técnicas linguísticas, estatísticas e de **ML** para realizar essas tarefas. Isso inclui o processamento de texto para identificar palavras-chave, análise gramatical, análise de contexto e até mesmo o reconhecimento de padrões em grandes volumes de texto. O artigo de Nadkarni et al., 2011 define como algumas das principais tarefas do **NLP**:

1. *Sentence boundary detection*: identificação de abreviações e títulos, como, por exemplo 'm.g.' 'Dr.';
2. *Tokenization*: identificação de *tokens* individuais (palavras, pontuação) dentro de uma frase. Um analisador de léxico desempenha um papel fundamental para esta tarefa e a anterior. Em determinados contextos, como textos biomédicos, os *tokens* frequentemente contêm caracteres normalmente usados como limites de *token*, como hífen e barras diagonais ('10 mg/dia,' 'N-acetilcisteína');
3. *Part of speech (POS)*: processo de atribuir rótulos gramaticais a cada palavra num texto para indicar a sua classe gramatical ou função na frase;
4. *Stemming e lemmatizing*: segundo Jivani, 2011 o *stemming* e *lemmatizing*, são métodos utilizados no **NLP** para reduzir as variantes das palavras à sua raiz, mas diferem na abordagem. O *stemming* aplica um conjunto de regras para obter a raiz da palavra sem considerar a classe gramatical ou o contexto. Já o *lemmatizing*, por outro lado, envolve compreender o contexto e a classe gramatical, visando encontrar o lema linguístico correto, que é a forma canónica de um lexema. Embora o *stemming* seja mais rápido e adequado para algumas aplicações, o *lemmatizing* é mais preciso, mas computacionalmente complexo;
5. *Shallow parsing (chunking)*: identificação de frases a partir de *tokens* etiquetados como partes do discurso. Por exemplo, uma frase nominal pode ser composta por uma sequência de adjetivos seguida de um substantivo;

6. *Problem-specific segmentation*: segmentação de texto em grupos significativos, como secções, por exemplo, Queixa Principal, Histórico Médico Anterior, etc;
7. *Spelling*: Identificação e correção de erros ortográficos e gramaticais;
8. *Named entity recognition (NER)*: identificação de palavras ou frases específicas (entidades) e a sua categorização, por exemplo, como pessoas, locais, doenças, genes ou medicamentos.

## 2.4 SENTIMENT ANALYSIS

Segundo Liu, 2010, a SA em texto, uma área específica do TM, pode ser definida como o estudo computacional de opiniões, sentimentos e emoções em textos.

A SA tem um conjunto específico de problemas a resolver. Um exemplo disso é a limitação na compreensão da ironia e sarcasmo, tal como identificado em Farias e Rosso, 2017. Já Ortigosa et al., 2014 identifica como principal problema a classificação de polaridade. Segundo Pang e Lee, 2008 os métodos mais utilizados para a SA são os algoritmos de ML e LB. Os métodos de ML envolvem o treino de modelos com dados classificados, para aprender a prever automaticamente o sentimento de novos textos. Já a abordagem LB envolve a criação de um dicionário de palavras com as suas polaridades (positiva, negativa ou neutra) e a utilização desse léxico para determinar o sentimento de um texto.

Pang e Lee, 2008 afirma que os algoritmos de ML têm melhores resultados em domínios específicos. Porém, quando é necessário classificar sentimentos de texto de diferentes domínios, a abordagem LB é superior.

Em trabalhos mais recentes, F. S. Britto, 2019 refere a realização de um projeto de SA em comentários de apps na Apple Store em português. Através de algoritmos de ML, foi verificado um melhor desempenho com modelos de *Recurrent neural network* (RNN) e *Long short-term memory* (LSTM).

No trabalho «Sentiment analysis in Facebook and its application to e-learning», Ortigosa et al., 2014, foi utilizada uma abordagem híbrida com dois modelos. O primeiro modelo utiliza a abordagem LB, utilizando um dicionário de palavras com polaridade emocional. Além disso, deteta outros elementos como interjeições, emojis, erros de ortografia, *Part of speech tagging* (POST) ou negação. Podemos ver na figura 3 o diagrama deste modelo.

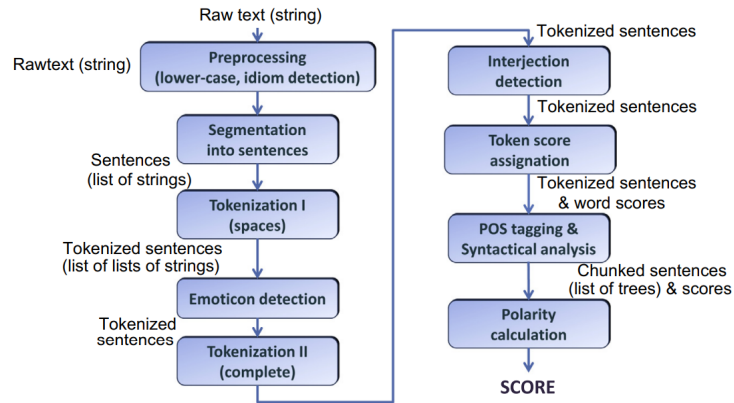


Figura 3: Diagrama LB «Sentiment analysis in Facebook and its application to e-learning»

Depois, foram utilizados os dados obtidos pelo classificador **LB** para treinar o segundo modelo baseado em **ML**, dando origem a uma abordagem híbrida de classificação.



## DESENVOLVIMENTO

---

### 3.1 METODOLOGIAS APLICADAS

Nesta secção, são apresentadas as metodologias utilizadas na implementação da ferramenta de avaliação de satisfação de munícipes através das redes sociais. O projeto envolve duas áreas principais: a criação do modelo de análise de sentimentos e a implementação da aplicação *web* que utiliza o modelo desenvolvido.

Para a primeira área foi escolhida a metodologia *Cross Industry Standard Process for Data Mining* (CRISP-DM). Esta metodologia é amplamente reconhecida e utilizada em projetos de *Data mining*, fornecendo um processo estruturado para guiar o desenvolvimento do modelo de análise de sentimentos. A metodologia CRISP-DM envolve várias etapas, desde a compreensão do negócio e dos dados até a implementação e monitorização do modelo.

Para a segunda área, o desenvolvimento da aplicação *web*, foi utilizada a metodologia ágil Scrum. Esta metodologia é adequada para projetos de desenvolvimento de software, especialmente aqueles que envolvem a implementação de novas funcionalidades ou melhorias contínuas em aplicações. A metodologia Scrum permite uma abordagem iterativa e incremental, permitindo que o desenvolvimento seja adaptado às mudanças e necessidades dos utilizadores e do negócio.

De seguida descreve-se como foram implementadas as metodologias CRISP-DM e Scrum em cada área do projeto. Além disso, são destacadas as principais vantagens de cada metodologia e como elas contribuem para o sucesso da implementação da ferramenta. Ambas as metodologias podem em determinados pontos sobrepor-se, e a conclusão de uma etapa pode implicar o desenvolvimento da outra. Torna-se assim necessário, na prática, uma integração das duas metodologias.

### 3.2 APLICAÇÃO DA METODOLOGIA CRISP-DM PARA CRIAÇÃO DO MODELO DE ANÁLISE DE SENTIMENTOS

No contexto do projeto, é necessária a criação de um modelo de análise de sentimentos. Para guiar este processo, foi escolhida a metodologia CRISP-DM. Esta envolve seis fases que podem ser observadas na Figura 4 e que são exploradas de seguida.

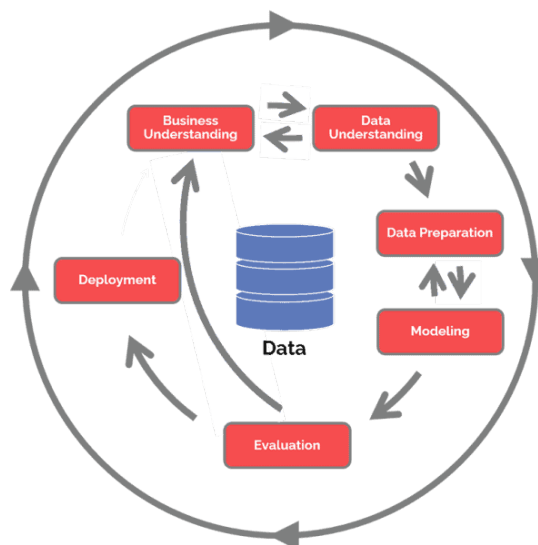


Figura 4: Metodologia CRISP-DM

### 3.2.1 *Compreensão do negócio*

No contexto do [CRISP-DM](#), a fase de compreensão do negócio é fundamental para estabelecer uma base sólida para o projeto de avaliação de sentimentos no Município de Pombal. Esta fase visa fornecer uma visão abrangente do problema, dos objetivos e das necessidades do negócio, a fim de orientar todas as etapas subsequentes da análise de dados.

#### 3.2.1.1 *Objetivos do Negócio*

O principal objetivo do projeto é avaliar os sentimentos e opiniões dos cidadãos do Município de Pombal em relação às políticas e ações adotadas pela administração municipal. Procura-se compreender a percepção geral da população em relação à administração, identificando áreas de satisfação e insatisfação.

#### 3.2.1.2 *Objetivos de [Data mining](#) e do modelo de [Sentiment analysis](#)*

Como referido anteriormente, pretende-se construir um modelo que deverá conseguir classificar o sentimento dos cidadãos aos comentários das publicações realizadas pelo Município. O requisito para o modelo é conseguir uma taxa de acerto superior a 80%.

### 3.2.1.3 *Contexto Municipal*

Pombal é um município Português situado no Distrito de Leiria e possui uma população diversificada com diferentes necessidades e expectativas em relação aos serviços municipais e políticas públicas.

### 3.2.1.4 *Motivações para o Projeto*

A auscultação com o representante do Município de Pombal revelou o interesse em estabelecer um canal de comunicação eficaz com seus munícipes, visando melhorar a prestação de serviços e a tomada de decisões informadas. Entender as percepções e sentimentos da comunidade é crucial para aprimorar a liderança local e promover o envolvimento cívico.

### 3.2.1.5 *Fontes de Dados*

As principais fontes de dados incluem comentários e interações dos munícipes nas publicações da página de Facebook do município, onde eles frequentemente expressam as suas opiniões e *feedbacks*.

### 3.2.1.6 *Restrições e Limitações*

O projeto deve lidar com a limitação de depender principalmente de dados disponíveis publicamente nas redes sociais, mais especificamente o Facebook, o que pode não representar completamente todas as perspectivas da população. As opiniões expressas nas redes sociais podem ser polarizadas e influenciadas por eventos específicos, tornando importante a contextualização dos dados.

### 3.2.1.7 *Benefícios Esperados*

O projeto visa fornecer informações importantes para a administração do Município de Pombal, promovendo melhores processos de decisão, informados com base nas percepções da comunidade. Espera-se que a análise de sentimentos contribua para melhorar a prestação de serviços, fortalecer a confiança da população e promover uma administração mais eficaz.

## 3.2.2 *Compreensão dos dados*

Nesta fase, são identificados os dados e efetuada a sua recolha. Neste caso, os dados a recolher estão disponíveis na página do Facebook do município. A recolha dos

dados começa pelas publicações e, em cada publicação, os seus comentários e reações. Para a recolha dos dados, foi necessário seguir os seguintes procedimentos:

- Criar uma aplicação no Facebook que permita a autenticação de utilizadores com a sua conta desta rede social. Neste caso, foi criada a aplicação "Análise sentimentos políticos".
- Neste contexto específico, além do acesso ao email para autenticação, queremos que o utilizador forneça permissão para acesso às publicações e comentários das páginas das quais é administrador. Assim que o utilizador fornece essa permissão, é possível, através da *Application programming interface* (API), recolher todas as publicações, reações e comentários a estas páginas.
- Foi necessário iniciar o desenvolvimento da ferramenta, nomeadamente as funcionalidades de autenticação e solicitação de permissões para a transferência dos dados desejados.

É importante destacar que a criação de modelos de aprendizagem supervisionada requer um conjunto de dados previamente classificado. Dado que a classificação manual de comentários não era viável devido ao considerável tempo necessário e à falta de recursos disponíveis para essa tarefa, adotou-se uma metodologia que passou por criar um conjunto de dados utilizando comentários de aplicações Android já classificados. Com base no número de estrelas atribuídas, avaliações com uma ou duas estrelas foram consideradas como negativas, enquanto avaliações com três ou mais estrelas foram consideradas como positivas. Reconhece-se que, devido a esta abordagem, o modelo pode apresentar um viés em relação ao seu domínio específico. No entanto, essa escolha foi aceite, uma vez que cumpre com objetivos propostos.

### 3.2.3 *Preparação dos dados*

A preparação dos dados incluiu a remoção de emails e números de telefone. Esta é justificada principalmente por questões de privacidade, conformidade com regulamentação e segurança. Além disso, a presença de emails e números de telefone pode influenciar negativamente o desempenho do modelo, uma vez que esses elementos não contribuem para o conteúdo textual relevante. Removê-los permite que a análise se concentre no conteúdo significativo.

Como vimos anteriormente, o trabalho com dados para domínios de *Natural language processing*, muitas vezes implica a utilização de modelos pré treinados para realizar tarefas como *Named entity recognition*, *Part of speech*, *lemmatization*, entre outros. Esta abordagem é extremamente útil em determinados contextos. Porém, neste projeto, esta não foi realizada pelos seguintes motivos:

1. Simplicidade e Eficiência: A implementação de **NER**, **POS** e *lemmatization* pode ser complexa, depender de anotações manuais, recursos linguísticos e exigir recursos computacionais significativos. Optou-se por uma abordagem mais simples, reduzindo a complexidade da *pipeline* do modelo, tornando-o mais interpretável, sem prejuízo nas métricas de avaliação e ajustado às necessidades e recursos disponíveis.
2. Foco na Classificação de Sentimentos: As técnicas mencionadas são valiosas em tarefas de análise linguística mais abrangentes, como, por exemplo, detecção de entidades. Não acrescentam, no entanto, utilidade no domínio da classificação de sentimentos com o modelo utilizado de **ML**.

### 3.2.4 Modelação

Nesta fase, é desenvolvido o modelo utilizando algoritmos de aprendizagem de acordo com os objetivos do projeto. Foram realizados alguns testes com diferentes modelos de classificação de texto, com o objetivo de perceber qual destes se comporta melhor de acordo com os dados recolhidos. De acordo com o que foi referido na secção 2.4, existem três caminhos a explorar:

#### 3.2.4.1 Modelos baseados no Léxico

A primeira hipótese, referida na secção 2.4, foi a criação de modelos baseados no léxico. Para exploração deste tipo de abordagem foi utilizada a biblioteca **Léxico para Inferência Adaptada (LeIA)**, que é uma adaptação para português do *Valence Aware Dictionary and Sentiment Reasoner (VADER)*, com um grande ponto positivo: a facilidade de utilização. Como referido na documentação da biblioteca, esta tenta resolver alguns dos problemas identificados na análise de sentimentos em texto, como:

1. Uso de contrações como negações (por exemplo, “wasn’t very good”);
2. Uso convencional de pontuação para sinalizar intensidade de sentimento aumentada (por exemplo, "Bom!!!");
3. Uso convencional de formato de palavra para sinalizar ênfase (por exemplo, usar TODAS AS LETRAS MAIÚSCULAS para palavras/frases);
4. Uso de modificadores de grau para alterar a intensidade do sentimento (por exemplo, intensificadores de intensidade como "muito" e suavizadores de intensidade como “meio”);
5. Compreensão de muitas palavras da gíria carregadas de sentimento (por exemplo, "sux");

6. Compreensão de muitas palavras da gíria carregadas de sentimento como modificadores (“uber” ou “friggin” ou “kinda”);
7. Compreensão de muitos *emojis* carregados de sentimento, como :) e :D;
8. Tradução de emojis codificados em utf-8;
9. Compreensão de siglas e acrónimos carregados de sentimento (por exemplo, “lol”).

Para ver o código utilizado na análise de sentimentos através de modelos baseado no léxico, consultar o apêndice 2.

#### 3.2.4.2 Modelos de aprendizagem supervisionada

A segunda abordagem explorada, também referida na secção 2.4, foi a aprendizagem supervisionada. Tendo em conta que não existia nenhum *dataset* previamente classificado, foi necessário criar um. Para o efeito, foi desenvolvido um *script* em Python que faz pedidos à *Application programming interface* da [Google Play Store \(GOPS\)](#) para receber os comentários dos utilizadores às aplicações Android. Como garantia que a maior parte dos comentários fossem em português, analisaram-se algumas das empresas portuguesas com mais instalações na [GOPS](#): empresas como Continente, RTP, Novo Banco, Público, Observador entre outras. Cada comentário está classificado de uma a cinco estrelas. Definiu-se uma ou duas estrelas como um comentário negativo, e uma classificação de 3 ou mais estrelas como positivo.

Os comentários foram escritos em dois ficheiros, um com os classificados como positivos e outro com os classificados como negativos. Na lista 1, é possível ver um exemplo com alguns comentários obtidos.

Listagem 1: Exemplos de comentários obtidos com o *script*

---

```

1 Tem o essencial. E o interface esta ao nivel do millennium online. E o
  ↳ complemento k faltava.
2 Dou 4 pk podia ser mais completa. Para quem so quer o basico esta mt bem e nao
  ↳ ha melhor.
3 Muito bom aplicação. Só faltava uma coisa. Ou eu não estou conseguindo
  ↳ descobrir isso. É o modo de terminar a secção
4 Bom dia Gostava de saber se e possiverl adicionar outra conta nesta APP Obj

```

---

É possível consultar o código relacionado com a construção do dataset de comentários no apêndice 3.

Foi conseguido um dataset com cerca de 24000 comentários, 12342 negativos e 11583 positivos. Para a criação do modelo foi utilizada a biblioteca Spacy, que é uma ferramenta dedicada à *Natural language processing* e foram seguidos os seguintes passos:

1. Dividir os *datasets* em treino e teste, sendo que o *dataset* de teste tem 5% das amostras.
2. Através da biblioteca Spacy, guardar os *datasets* em documentos binários para poderem ser processados mais tarde, consultar apêndice 5.
3. Criar um ficheiro de configuração que defina quais os *datasets*, *pipelines* e parâmetros a utilizar. Neste caso, a *pipeline* de classificação de texto, e os parâmetros necessários ao treino do modelo, como o otimizador, número de épocas de treino, número máximo de passos entre outros. Para o efeito, a Spacy disponibiliza um ficheiro pré preenchido no site adequado à tarefa.
4. Utilizar a aplicação de linha de comandos fornecida pela Spacy para o treino e criação do modelo de classificação.

É importante salientar que o requisito deste projeto é colocar em produção um modelo de classificação. Como vimos, para o efeito foi selecionada a ferramenta Spacy que fornece os melhores parâmetros para esta tarefa num ficheiro de configuração. Não sendo o objetivo do trabalho a exploração de diversos algoritmos, otimizadores e diferentes parâmetros, o modelo foi construído com os parâmetros recomendados por omissão. Estes podem ser encontrados no apêndice 4. Porém, refere-se que foi utilizado um modelo para classificação do tipo *Bag of words* e o otimizador AdamV1.

#### 3.2.4.3 Modelos Pré treinados (*Bidirectional Encoder Representations from Transformers (BERT)*)

Foram ainda efetuadas algumas experiências através da biblioteca *transformers* com o **BERT**, um modelo de *Natural language processing* desenvolvido pela Google em 2018. Algumas das suas características são:

- Contexto Bidirecional: o **BERT** foi projetado para compreender o contexto das palavras numa frase, considerando tanto o contexto à esquerda quanto à direita. Ao contrário de modelos anteriores que analisam as palavras de maneira unidirecional, o **BERT** utiliza *transformers* bidirecionais, permitindo-lhe capturar o contexto completo de uma palavra;
- Pré-treino e Ajuste: o **BERT** é pré-treinado num grande corpus de dados, o que significa que aprende padrões e relações linguísticas a partir destes. Após o pré-treino, o **BERT** pode ser ajustado para tarefas específicas de **NLP**, como classificação de texto;
- *Transformers*: o **BERT** é baseado na arquitetura *transformer*, que é uma rede neuronal. Os *transformers* usam mecanismos de auto-atenção para ponderar a importância de diferentes partes de uma sequência ao processá-la.

As experiências efetuadas com o [BERT](#) foram todas em Inglês e a classificação de texto é feita de forma extremamente simples, não tendo sido necessário nenhum ajuste para serem conseguidos bons resultados. No entanto, a classificação de texto na língua Portuguesa com a biblioteca BERTimbau não foi conseguida, devido a erros na compilação do código.

### 3.2.5 Avaliação

Nesta fase, foi avaliada a eficácia dos modelos de análise de sentimentos, tanto do classificador baseado no léxico como do modelo de aprendizagem supervisionada. Foram comparadas as métricas de avaliação dos modelos, como *precision*, *recall*, *f1 score* e *Area Under the Curve (AUC) of the Receiver Operating Characteristic (ROC)*. Por fim, foram avaliados os resultados em relação aos objetivos do projeto e requisitos do município. Como é possível verificar nas tabelas e imagens em baixo, o modelo de aprendizagem supervisionada conseguiu resultados muito superiores aos resultados do classificador baseado no léxico, com uma *accuracy* de 95% contra 54%. A superioridade revela-se principalmente na classificação de comentários positivos, onde o modelo baseado no léxico provou-se ineficaz com uma precisão pouco acima dos 5%, ao passo que o modelo de aprendizagem supervisionada conseguiu uma precisão na ordem dos 95%, tanto nos comentários positivos como negativos. Tendo em conta os resultados obtidos, foi o modelo de aprendizagem supervisionada o escolhido para integrar a aplicação *web*.

Modelo de aprendizagem supervisionada			
	Precision	Recall	F1-Score
Positivo	96.82	94.80	95.80
Negativo	94.96	96.97	95.96

Tabela 1: Avaliação do modelo de aprendizagem supervisionada, *Precision*, *Recall* e *F1 score*

Modelo de aprendizagem supervisionada	
ROC AUC score	
0.99	

Tabela 2: Avaliação do modelo de aprendizagem supervisionada, [ROC AUC score](#)

Modelo de aprendizagem supervisionada	
<i>Accuracy</i>	
0.95	

Tabela 3: Avaliação do modelo de aprendizagem supervisionada, *accuracy*

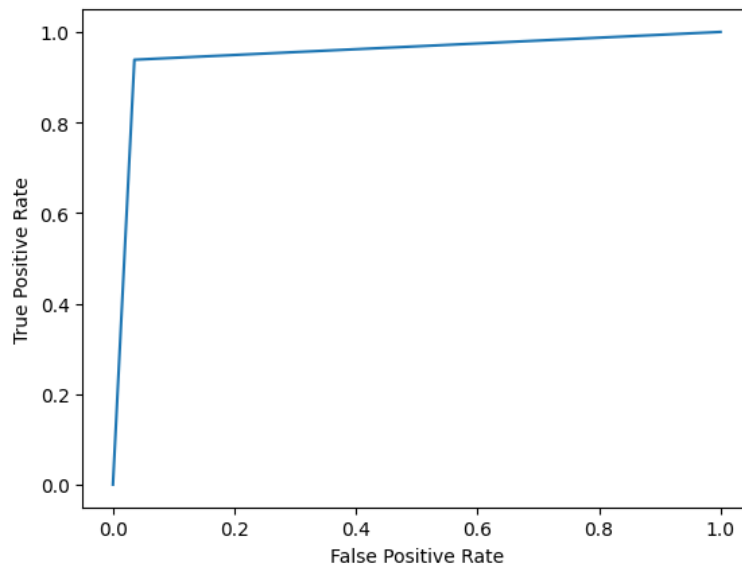


Figura 5: Avaliação do modelo de aprendizagem supervisionada, curva **ROC**

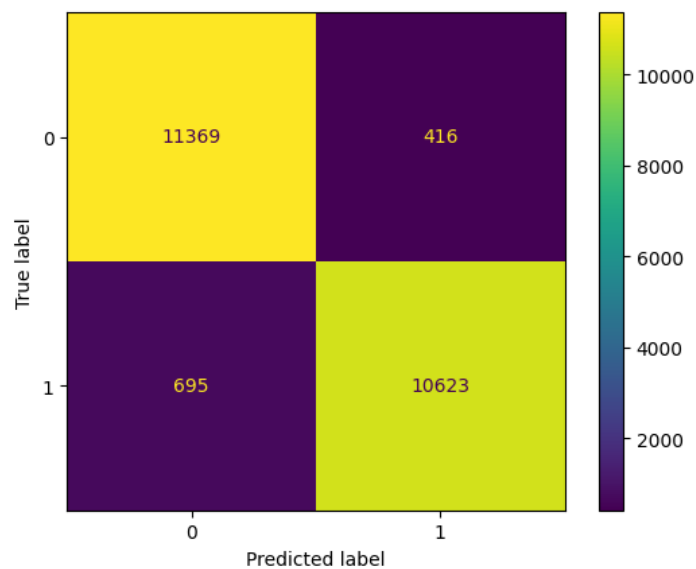


Figura 6: Matriz de confusão do modelo de aprendizagem supervisionada

Modelo baseado no Léxico			
	Precision	Recall	F1-Score
Positivo	5.11	99.86	9.37
Negativo	97.36	4.92	67.66

Tabela 4: Avaliação do modelo baseado no Léxico, *Precision*, *Recall* e *F1 score*

Modelo baseado no Léxico	
ROC AUC score	
0.52	

Tabela 5: Avaliação do modelo baseado no Léxico, **ROC AUC score**

Modelo baseado no Léxico
<i>Accuracy</i>
0.54

Tabela 6: Avaliação do modelo baseado no Léxico, *accuracy*

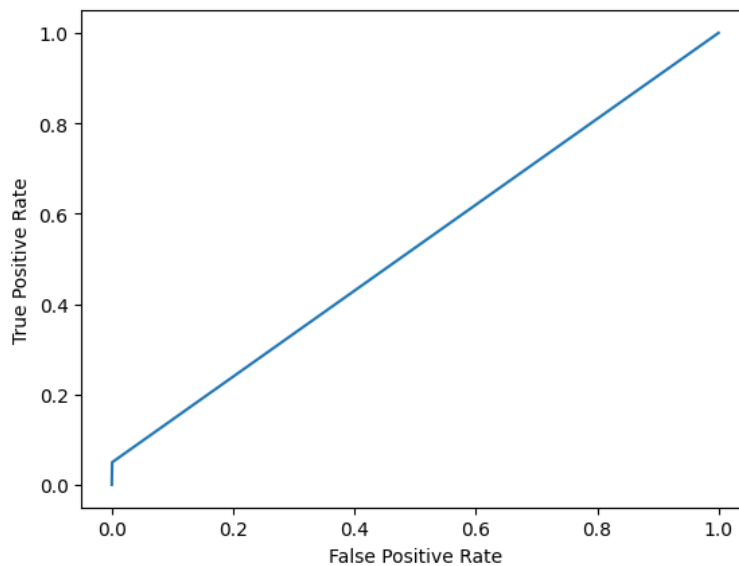


Figura 7: Avaliação do modelo baseado no Léxico, curva ROC

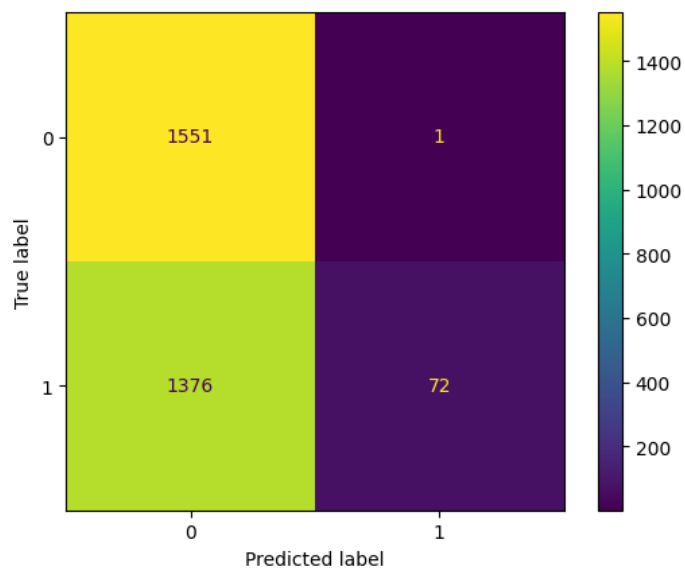


Figura 8: Matriz de confusão do modelo baseado no léxico

### 3.2.6 Implementação

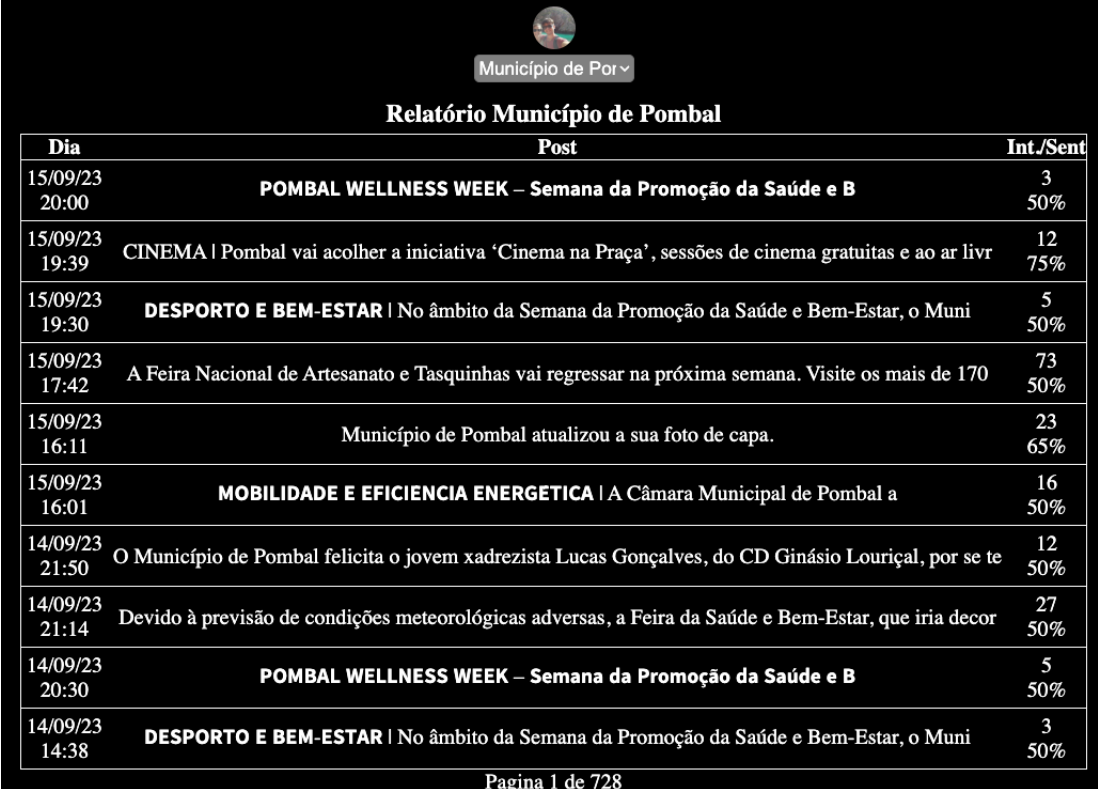
Para a implementação do modelo, foi inicialmente desenvolvida uma ferramenta de linha de comandos onde é possível inserir texto para o modelo classificar o sentimento, ver figura 9. Consultar o apêndice 7.

```
#####
Escreva um texto para classificação ou pressione 1 e ENTER para sair
Acho que o presidente foi infeliz
As tags que o modelo prevê são:
Acho que o presidente foi infeliz
tag          precision
Negative     0.5894649624824524
Positive     0.4010196328163147
#####
Escreva um texto para classificação ou pressione 1 e ENTER para sair
Parabens à camara
As tags que o modelo prevê são:
Parabens à camara
tag          precision
Positive     0.7157456278800964
Negative     0.19464541971683502
#####
Escreva um texto para classificação ou pressione 1 e ENTER para sair
```

Figura 9: CLI para análise de sentimentos

Após o desenvolvimento da aplicação *web*, o modelo foi integrado a esta como um micro serviço, utilizando a *framework* FastAPI. Esta *framework* permite registrar *endpoints* *Representational State Transfer* (REST) de forma simples e utilizar o protocolo *Hypertext Transfer Protocol* (HTTP) para receber os comentários a serem classificados e enviar as respostas. No apêndice 6, é possível observar o código que mostra em detalhe a forma como foi implementado o serviço para utilizar o modelo.

A figura 10 mostra um dos ecrãs da aplicação. Neste ecrã, é possível visualizar uma página com publicações apresentadas numa tabela de 3 colunas. Na primeira coluna, consta a data de criação do *post* e na segunda, o texto. Na última coluna, estão representadas duas métricas. A primeira representa o nível de envolvimento dos utilizadores e é calculada pelo número de comentários, emojis e partilhas: quanto maior o envolvimento, maior será o valor. A segunda métrica, em percentagem, é a média dos sentimentos identificados nos comentários à publicação, podendo concluir que uma média superior a 50% indica uma publicação com uma reação positiva.



Dia	Post	Int./Sent
15/09/23 20:00	<b>POMBAL WELLNESS WEEK – Semana da Promoção da Saúde e B</b>	3 50%
15/09/23 19:39	CINEMA   Pombal vai acolher a iniciativa ‘Cinema na Praça’, sessões de cinema gratuitas e ao ar livre	12 75%
15/09/23 19:30	<b>DESPORTO E BEM-ESTAR</b>   No âmbito da Semana da Promoção da Saúde e Bem-Estar, o Muni	5 50%
15/09/23 17:42	A Feira Nacional de Artesanato e Tasquinhas vai regressar na próxima semana. Visite os mais de 170	73 50%
15/09/23 16:11	Município de Pombal atualizou a sua foto de capa.	23 65%
15/09/23 16:01	<b>MOBILIDADE E EFICIENCIA ENERGETICA</b>   A Câmara Municipal de Pombal a	16 50%
14/09/23 21:50	O Município de Pombal felicita o jovem xadrezista Lucas Gonçalves, do CD Ginásio Louriçal, por se te	12 50%
14/09/23 21:14	Devido à previsão de condições meteorológicas adversas, a Feira da Saúde e Bem-Estar, que iria decor	27 50%
14/09/23 20:30	<b>POMBAL WELLNESS WEEK – Semana da Promoção da Saúde e B</b>	5 50%
14/09/23 14:38	<b>DESPORTO E BEM-ESTAR</b>   No âmbito da Semana da Promoção da Saúde e Bem-Estar, o Muni	3 50%

Página 1 de 728

Figura 10: Relatório com médias da análise de sentimentos e *engagement*

### 3.2.7 Monitorização

A monitorização da eficácia de um modelo de classificação passa por conseguir detetar os erros (neste caso, a incorreta classificação dos comentários), assim como dar *feedback* ao modelo dos acertos (comentários bem classificados). Uma maneira de conseguir este *feedback* é ir criando o próprio *dataset*, em vez de utilizar comentários das aplicações da **GOPS**, ou seja, à medida que os utilizadores vão encontrando comentários às publicações, assinalarem se a classificação está ou não correta. Assim que exista um número considerável de comentários manualmente classificados é possível treinar novos modelos com melhor desempenho.

## 3.3 METODOLOGIA ÁGIL SCRUM, DESENVOLVIMENTO DA APLICAÇÃO WEB

A metodologia Scrum enfatiza a flexibilidade e a adaptação constante, permitindo que as equipas respondam às mudanças de forma eficaz e priorizem o trabalho com base no valor para o município. No contexto deste projeto, escolheu-se aplicar esta metodologia para o desenvolvimento da aplicação *web*, que vai utilizar o modelo da análise de sentimentos criado anteriormente. O processo segue as seguintes etapas:

### 3.3.1 Escolha de tecnologias

A etapa de seleção das tecnologias representa uma etapa importante no desenvolvimento de qualquer nova aplicação. É um momento em que se delinea o cenário tecnológico que sustentará o projeto e moldará a sua implementação. Nesta fase, é imperativo considerar não apenas os objetivos específicos da aplicação, mas também a experiência e familiaridade dos programadores.

A arquitetura de uma aplicação *web* moderna é frequentemente construída com um conjunto de micro serviços, cada um a desempenhar um papel diferente nas funcionalidades do sistema. O sucesso desta arquitetura reside na escolha das tecnologias subjacentes, que devem ser capazes de ter em conta não apenas a performance, escalabilidade e manutenção dos diversos componentes, como também o planeamento de tarefas, gestão do código fonte e capacidade de correr a aplicação de forma simples em ambientes de testes e produção. Por fim, a escolha de linguagens de programação com bom suporte de comunidade e que permitam um desenvolvimento rápido e seguro é também algo importante. Tendo em conta estes pressupostos e requisitos, foram identificadas as seguintes necessidades:

- Manter um registo do *backlog* e das tarefas
- Usar o Github para gestão do código fonte, versões e adição de novas funcionalidades de forma simples, rápida e colaborativa.
- Usar o Docker para separar a aplicação da infraestrutura, e conseguir de forma rápida e fácil instalar e servir a aplicação em qualquer sistema, seja para ambiente de testes ou produção.
- Usar Typescript e a *framework* NextJs para o desenvolvimento da aplicação, garantindo boas práticas de programação, suporte da comunidade, desenvolvimento rápido e uma *codebase* de fácil acesso a novos programadores.
- Usar a linguagem Golang, mais especificamente a *framework* PocketBase, para acesso à base de dados e tarefas que envolvam concorrência em *backend*, garantindo um sistema rápido e eficiente no uso de recursos. Esta *framework* utiliza uma base de dados SQLite para armazenamento de todos os dados.

Na figura 11, encontra-se esquematizada a arquitetura dos serviços integrantes da aplicação.

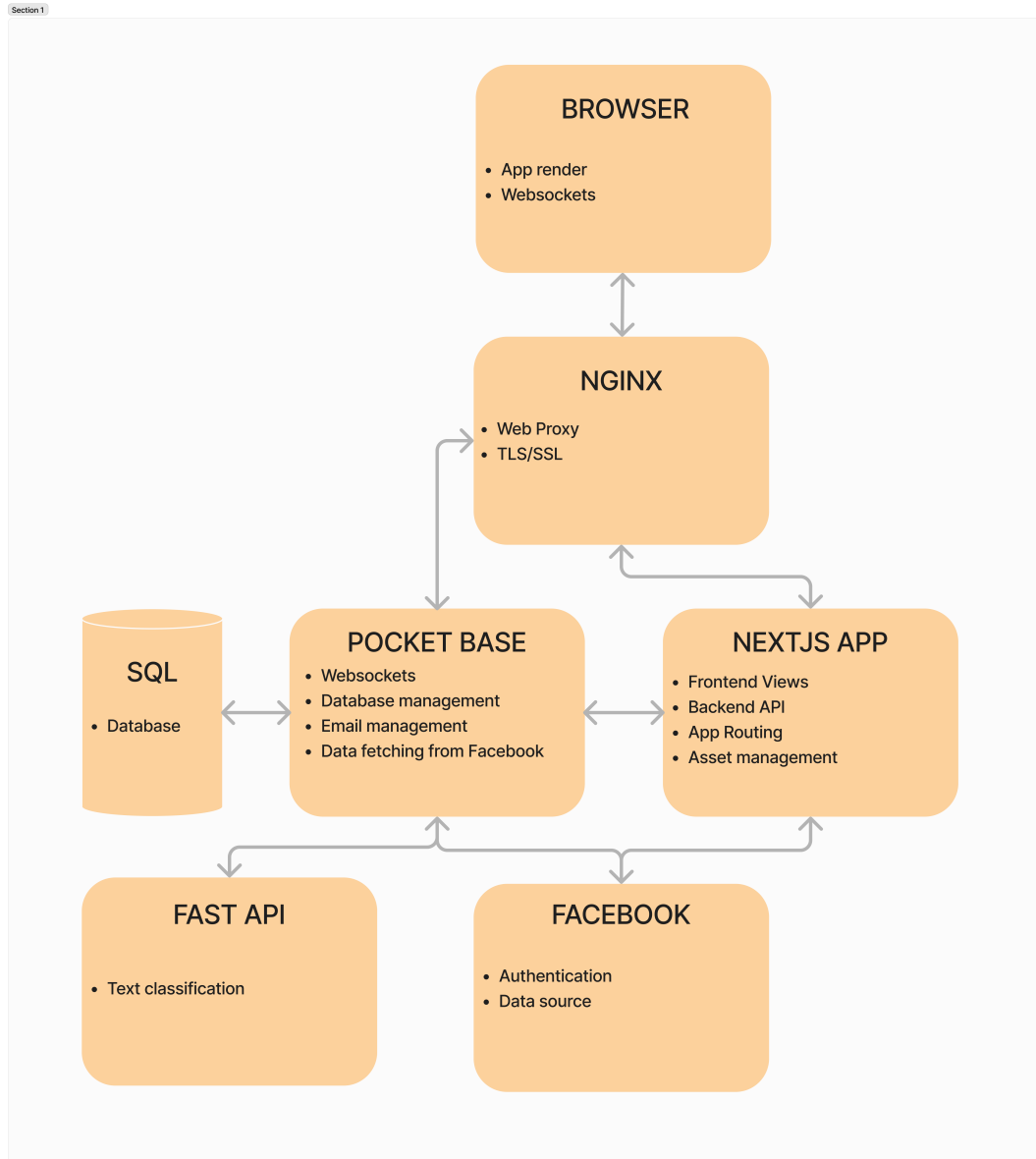


Figura 11: Arquitetura dos serviços

### 3.3.2 Definição do backlog

Nesta etapa, são identificadas todas as funcionalidades que a aplicação *web* deverá ter, assim como todos os passos necessários para a sua implementação, tendo em consideração as necessidades dos utilizadores e os objetivos do projeto. Essas funcionalidades são organizadas no *backlog* do produto, que é utilizado para guiar o desenvolvimento da aplicação.

#### 3.3.2.1 Configuração Inicial:

Criação de um repositório Git:

- Uma pasta com todos os ficheiros e configurações relacionados com os micro serviços Docker.
  - FastAPI para classificação de comentários.
  - Nginx para *proxy* dos vários serviços, com certificados *Transport Layer Security* (TLS) e ficheiros de configuração independentes para desenvolvimento e produção.
  - Base de dados unida à parte do *backend* responsável pela comunicação com o Facebook, através da *framework* PocketBase. É necessário manter um Dockerfile para desenvolvimento com acesso a *debuggers* para facilitar o processo de desenvolvimento, e outro Dockerfile para produção com o *backend* compilado para código máquina, garantido a máxima performance.
  - Utilizar *scripts* em *bash* para a inicialização correta dos *containers*.
  - *Frontend* e parte do *backend* em NextJs, com um ficheiro Dockerfile para desenvolvimento com acesso a *debugger*, e outro Dockerfile para produção com *bundled assets*, minimizando o tamanho dos ficheiros servidos.
  - Ficheiros Docker Compose independentes, para desenvolvimento e produção, garantindo a gestão de todos os micro serviços de forma simples.
  - Ficheiro de variáveis de ambiente para fácil configuração dos serviços.
- Criar pastas para o código fonte de cada serviço que vão ser mapeadas para dentro dos *containers* durante o desenvolvimento da aplicação, com as respetivas variáveis de ambiente necessárias para os serviços (por exemplo, chaves de API, credenciais do Facebook, etc.).

#### 3.3.2.2 Desenvolvimento do Backend (PocketBase):

- Implementar autenticação via Facebook.
- Fazer *download* de todas as publicações do utilizador, assim como os comentários, partilhas e reações.
- Garantir os *downloads* de forma concorrente fornecendo um serviço rápido e de utilização satisfatória.
- Fazer pedidos ao serviço de classificação para classificar cada comentário como positivo ou negativo.
- Guardar em base de dados todas as páginas dos utilizadores, assim como publicações, reações e comentários associados a estas.

- Desenvolver o sistema com *websockets*, para comunicação em tempo real com o utilizador, garantindo o *feedback* das operações efetuadas.
- Correr *cron jobs* diariamente para sincronização de todas as páginas em sistema, e envio de emails para os utilizadores configurados na ferramenta.

### 3.3.2.3 *Desenvolvimento da aplicação web em NextJs:*

- Implementar o fluxo de autenticação via Facebook, incluindo a obtenção do *token* de acesso de utilizador, de página e *token* de longa duração.
- Página de *dashboard*:
  - Seletor de página.
  - Gráfico com todas as publicações e respetiva pontuação, ordenadas cronologicamente e paginadas.
  - Seletor de publicação para abrir a caixa de detalhes.
  - *Modal window* com os detalhes da publicação, resumo das reações, comentários e possibilidade de expandir os comentários para acesso individual.
  - Botão de sincronização de página onde são atualizadas as publicações, reações e comentários.
- Página de relatório:
  - Seletor de página.
  - Tabela onde podem ser visualizadas as publicações, pontuação de engajamento e uma média dos sentimentos associados aos comentários e emojis.
- Página de configurações:
  - Seletor de página.
  - Lista de emails para receberem notificações semanais da página selecionada.
  - Botão para adicionar nova entrada na lista.
- Implementar funcionalidade de *logout*.

### 3.3.2.4 *Desenvolvimento do serviço de classificação de publicações:*

- Criar *endpoints* no serviço com a *framework* FastAPI para receber e classificar comentários de publicações.

#### 3.3.2.5 *Configuração do NGINX como proxy:*

- Criar e utilizar certificados [TLS](#)
- Configurar o servidor NGINX para atuar como *proxy* reverso.
- Implementar as regras de redirecionamento de tráfego para os serviços apropriados (*frontend* e *backend*).

#### 3.3.2.6 *Preparação para Produção:*

- Configurar o Docker Compose para produção, incluindo otimizações de imagem, variáveis de ambiente seguras, entre outros.
- Testar a aplicação em ambiente de produção para garantir que todos os serviços funcionem corretamente.

#### 3.3.2.7 *Testes e Otimizações Finais:*

- Realizar testes a todos os níveis para verificar a funcionalidade completa da aplicação.
- Identificar possíveis problemas de desempenho e otimizar o código, consultas à base de dados e tempos de resposta.

#### 3.3.3 *Planeamento dos Sprints:*

Nesta etapa são selecionadas as funcionalidades do *backlog* do produto que são implementadas em cada *sprint*. Os *sprints* são ciclos de desenvolvimento de curto prazo (geralmente de 2 a 4 semanas) nos quais o objetivo é implementar um conjunto específico de funcionalidades. A [tabela 7](#) mostra o planeamento dos *sprints* pela ordem necessária.

<i>Sprint</i>	Tempo previsto
Criação do repositório em Git e estrutura de pastas	1 a 3 dias
Desenho das páginas de autenticação, <i>dashboard</i> , relatórios e configuração	1 semana
Configuração do servidor NGINX para atuar como proxy reverso com <a href="#">TLS</a>	2 a 3 dias
Iniciar um novo projeto em Nextjs	2 a 3 dias
Iniciar um novo projeto em PocketBase	2 a 5 dias
Iniciar um novo projeto em FastAPI	1 dia
Desenvolvimento dos componentes <i>Header</i> em Nextjs	1 dia
Implementar a autenticação via Facebook	2 a 4 dias
Desenvolvimento dos componentes necessários ao <i>Dashboard</i> , com gráficos e sincronização	1 a 2 semanas, ver apêndice <a href="#">12</a>
Sincronização das publicações	3 semanas
Sincronização das reações	2 semanas
Sincronização dos comentários	2 semanas
Desenvolvimento dos componentes para visualização dos detalhes de uma publicação, comentários, reações e análise	2 semanas, ver apêndice <a href="#">15</a>
Desenvolvimento da página de relatórios onde se podem ver as publicações em formato tabular e o resumo das suas métricas	1 a 2 semanas, ver apêndice <a href="#">13</a>
Desenvolvimento da página de configurações onde podem ser adicionados os emails que devem receber notificações	1 semana, ver apêndice <a href="#">14</a>
Desenvolvimento do mecanismo de sincronização automática de publicações	1 a 3 semanas

Tabela 7: Planeamento dos *sprints*

### 3.3.4 *Desenvolvimento e Entrega:*

Normalmente, durante os *sprints*, são desenvolvidas as tarefas selecionadas, seguindo as práticas de desenvolvimento *ágil*, como colaboração constante, reuniões diárias e entrega rápida de *software*. O trabalho é dividido em tarefas menores e o progresso é monitorizado regularmente. Neste caso, o projeto tem apenas um programador, e não as habituais equipas de média ou grande dimensão. As reuniões com o município

foram quinzenais e não diárias. No entanto o objetivo das reuniões manteve-se, foram mostrados os progressos, recebido *feedback* do *sprint* e ajustados os objetivos até à próxima reunião.



## CONCLUSÕES

---

O projeto foi dividido em duas fases distintas, cada uma seguindo uma abordagem metodológica diferente.

Na primeira fase, optou-se pela metodologia [CRISP-DM](#) para criar um modelo de classificação dos comentários às publicações no Facebook. Este modelo desempenhou um papel fundamental na identificação dos sentimentos presentes nos comentários às publicações, proporcionando uma base sólida para a análise subsequente. Nesta fase, foram encontradas algumas dificuldades como referido na secção [3.2.4.3](#), onde a utilização do [BERT](#) em Português não foi conseguida.

Na segunda fase, foi utilizada a metodologia ágil Scrum para desenvolver a aplicação em NextJs, com um *backend* em Pocketbase. Esta abordagem iterativa permitiu criar a ferramenta de forma planeada e organizada, assegurando a sua funcionalidade e usabilidade. Aqui, as maiores dificuldades foram a aprendizagem de novas *frameworks* (PocketBase e NextJs).

A intenção inicial era servir apenas o município de Pombal, mas à medida que o projeto progrediu, foi desenvolvida uma plataforma que poderá estar disponível para qualquer pessoa interessada na análise dos sentimentos que as suas publicações despertam. É necessário o acesso a uma página pública e uma aplicação para programadores no Facebook com os pedidos de permissões referidos anteriormente (acesso a publicações, comentários, partilhas). Para utilizar a aplicação (conforme foi desenvolvida), é necessário recorrer ao Docker como plataforma de virtualização para correr os serviços, e alterar as variáveis de ambiente (como chaves de [API](#) do Facebook, servidor de email, entre outros), consultar o apêndice [8](#).

Realça-se que a ferramenta integra um serviço de sincronização diária de *posts*, mantendo a base de dados atualizada. Inclui também uma interface de configuração de *emails*, segmentados por página. Nesta interface, os utilizadores podem escolher receber notificações sobre novas sincronizações de *posts* (por exemplo, o presidente da câmara pode receber avisos quando existirem novas sincronizações), ver imagem [14](#) e [15](#).

#### 4.1 TRABALHO FUTURO

Uma das partes mais desafiantes do projeto, assim como em muitos projetos de ciência de dados, foi o acesso aos dados. Neste trabalho, apresentou-se o caso particular das dificuldades patentes no trabalho com a [API](#) do Facebook. Os avanços realizados neste projeto possibilitam uma maior facilidade, em trabalhos futuros, no acesso aos dados. Este acesso pode ser agora realizado com recurso à [API](#) que a ferramenta disponibiliza (Pocketbase), ou ao acesso direto à base de dados *Structured Query Language*, e assim retirar mais informação relevante dos dados.

Tendo em conta as limitações do projeto e desenvolvimento de trabalhos futuros, outras potencialidades e sugestões de desenvolvimento que podem vir a ser aplicadas ao projeto são as seguintes:

1. Criar mecanismos de correção de erros;
2. Desenvolver um dataset com comentários que o modelo tenha tido dificuldade em validar, mais próximo da realidade onde irá ser aplicado;
3. Utilizar [API](#) de serviços como OpenAI, ou outros modelos como [BERT](#), entre outros;
4. Melhorar a *User interface* (UI) e *User experience* (UX);
5. Utilizar as [API](#) do Facebook para ir buscar outros dados;
6. Explorar mais arquiteturas para a classificação de sentimentos com a biblioteca Spacy, ou mesmo com outras bibliotecas;
7. Despoletar alertas em publicações que estejam com um nível elevado de envolvimento por parte dos munícipes.

A aplicação oferece aos utilizadores a capacidade de explorar análises de sentimentos em publicações no Facebook. Outra vantagem particular assenta na arquitetura desenvolvida, que permite que, com facilidade, possam ser implementadas funcionalidades semelhantes noutras redes sociais. Pode assim ficar centralizada toda a informação relativa à análise de sentimentos nas redes sociais da instituição.

Em conclusão, apesar das limitações e desafios encontrados ao longo do caminho, o projeto cumpriu a sua missão original. A aplicação resultante não contribui apenas para a análise dos sentimentos, mas também demonstra como abordagens práticas podem gerar soluções úteis num contexto tecnológico em constante evolução.

## BIBLIOGRAFIA

---

- Bowonder, B. e T. Miyake (1992). «Creating and sustaining competitiveness: Information management strategies of Nippon Steel Corporation». Em: *International Journal of Information Management* 12.1, pp. 39–56. ISSN: 0268-4012. DOI: [https://doi.org/10.1016/0268-4012\(92\)90051-Q](https://doi.org/10.1016/0268-4012(92)90051-Q). URL: <https://www.sciencedirect.com/science/article/pii/026840129290051Q>.
- Duan, Yanqing, John S. Edwards e Yogesh K Dwivedi (2019). «Artificial intelligence for decision making in the era of Big Data – evolution, challenges and research agenda». Em: *International Journal of Information Management* 48, pp. 63–71. ISSN: 0268-4012. DOI: <https://doi.org/10.1016/j.ijinfomgt.2019.01.021>. URL: <https://www.sciencedirect.com/science/article/pii/S0268401219300581>.
- F. S. Britto, Luciano D. S. Pacífico (2019). *Sentiment Analysis for Mobile App Reviews in Brazilian Portuguese*. <https://sol.sbc.org.br/index.php/eniac/article/view/9359>.
- Farias, D.I. Hernández e P. Rosso (2017). «Chapter 7 - Irony, Sarcasm, and Sentiment Analysis». Em: *Sentiment Analysis in Social Networks*. Ed. por Federico Alberto Pozzi et al. Boston: Morgan Kaufmann, pp. 113–128. ISBN: 978-0-12-804412-4. DOI: <https://doi.org/10.1016/B978-0-12-804412-4.00007-3>. URL: <https://www.sciencedirect.com/science/article/pii/B9780128044124000073>.
- Hearst, Marti (2003). *What Is Text Mining?* <https://www.jaist.ac.jp/~bao/MOT-Ishikawa/FurtherReadingNo1.pdf>. [Online; accessed 11-September-2023].
- Jivani, Anjali Ganesh (2011). *A Comparative Study of Stemming Algorithms*. [https://kenbenoit.net/assets/courses/tcd2014qta/readings/Jivani\\_ijcta2011020632.pdf](https://kenbenoit.net/assets/courses/tcd2014qta/readings/Jivani_ijcta2011020632.pdf). [Online; accessed 11-September-2023].
- Kaya, Bahadır (2023). *Key Insights from the 2023 AI (Artificial Intelligence) Index Report*. <https://www.linkedin.com/pulse/key-insights-from-2023-ai-artificial-intelligence-index-bahadir-kaya/>. [Online; accessed 29-September-2023].
- Liu, Bing (2010). «Sentiment Analysis and Subjectivity». Em: *Handbook of Natural Language Processing*.
- Lu, Ming-Te e S.P. Mooney (1989). «Assessing expert system applications: A case study». Em: *International Journal of Information Management* 9.4, pp. 267–273. ISSN: 0268-4012. DOI: [https://doi.org/10.1016/0268-4012\(89\)](https://doi.org/10.1016/0268-4012(89))

90050-9. URL: <https://www.sciencedirect.com/science/article/pii/S0268401289900509>.

Nadkarni, Prakash M, Lucila Ohno-Machado e Wendy W Chapman (set. de 2011). «Natural language processing: an introduction». Em: *Journal of the American Medical Informatics Association* 18.5, pp. 544–551. ISSN: 1067-5027. DOI: [10.1136/amiajnl-2011-000464](https://doi.org/10.1136/amiajnl-2011-000464). eprint: <https://academic.oup.com/jamia/article-pdf/18/5/544/5962687/18-5-544.pdf>. URL: <https://doi.org/10.1136/amiajnl-2011-000464>.

Ortigosa, Alvaro, José M. Martín e Rosa M. Carro (2014). «Sentiment analysis in Facebook and its application to e-learning». Em: *Computers in Human Behavior* 31, pp. 527–541. ISSN: 0747-5632. DOI: <https://doi.org/10.1016/j.chb.2013.05.024>. URL: <https://www.sciencedirect.com/science/article/pii/S0747563213001751>.

Pang, Bo e Lillian Lee (2008). «Opinion Mining and Sentiment Analysis». Em: *Foundations and Trends® in Information Retrieval* 2.1–2, pp. 1–135. ISSN: 1554-0669. DOI: [10.1561/1500000011](https://doi.org/10.1561/1500000011). URL: <http://dx.doi.org/10.1561/1500000011>.

## APÊNDICES



## Listagem 2: Teste à biblioteca LeIA, modelo baseado no Léxico

---

```

1 from LeIA import SentimentIntensityAnalyzer
2 import pandas as pd
3 from sklearn.metrics import precision_recall_fscore_support, roc_auc_score
4
5 def classificar_sentimento(frase):
6     sia = SentimentIntensityAnalyzer()
7     polaridade = sia.polarity_scores(frase)
8     if polaridade['pos'] >= 0.5:
9         return 1
10    if polaridade['pos'] < 0.5:
11        return 0
12    else:
13        return 2
14
15
16 negative = pd.read_csv('pt_negative_reviews.csv')
17 positive = pd.read_csv('pt_positive_reviews.csv')
18 negative['sentiment'] = 0
19 positive['sentiment'] = 1
20
21 total = pd.concat([negative, positive], ignore_index=True)
22 total['prev'] = 0
23 total['res'] = False
24 total = total.sample(frac = 1)
25 total.drop(total.index[range(3000, len(total.index))], inplace=True)
26 total = total.reset_index()
27
28 for ind in total.index:
29     total.loc[ind, 'prev'] = classificar_sentimento(total['comentario'][ind])
30     if total.loc[ind, 'prev'] == total.loc[ind, 'sentiment']:
31         total.loc[ind, 'res'] = True
32
33 total.drop(total[total['prev'] == 2].index, inplace=True)
34 total = total.reset_index()
35
36 y_true = total['sentiment'].values.tolist()
37 y_pred = total['prev'].values.tolist()
38
39 flscore = precision_recall_fscore_support(y_true, y_pred)
40 print(flscore)
41 print(roc_auc_score(y_true, y_pred))
42
43
44

```

---

Listagem 3: *Script* para construção de dataset de comentários para aprendizagem supervisionada

---

```

1 from google_play_scraper import app
2 import pandas as pd
3 import numpy as np
4 from google_play_scraper import Sort, reviews_all

```

```

5 import csv
6
7 apps = ['pt.observador.mobile.android', 'pt.publico.android',
8 ↪ 'pt.impresapub.expresso', 'pt.rtp.play',
9 ↪ 'pt.rtp.noticias.android', 'com.impresa.opt',
10 ↪ 'pt.ptinovacao.rma.meomobile', 'com.megasis.android',
11 ↪ 'pt.continente.ContinenteTelemovel', 'pt.observador.mobile.android',
12 ↪ 'pt.publico.android', 'pt.impresapub.expresso', 'pt.rtp.play',
13 ↪ 'pt.rtp.noticias.android', 'com.impresa.opt',
14 ↪ 'pt.ptinovacao.rma.meomobile', 'pt.pingodoce',
15 ↪ 'pt.santander.oneappparticulares',
16 ↪ 'pt.bancobpi.mobile.fiabilizacao', 'pt.novobanco.nbsmarter',
17 ↪ 'wit.android.bcpBankingApp.millennium']
18
19 for app in apps:
20     print(app + "#####")
21     pt_positive_reviews = []
22     pt_negative_reviews = []
23     app_pt_reviews = reviews_all(
24         app,
25         sleep_milliseconds=0, # defaults to 0
26         lang='pt', # defaults to 'en'
27         country='pt', # defaults to 'us'
28         sort=Sort.NEWEST, # defaults to Sort.MOST_RELEVANT
29     )
30     for review in app_pt_reviews:
31         if (review['content'] is None or len(review['content']) < 50):
32             continue
33         if (review['score'] >= 3):
34             pt_positive_reviews.append([review['content']])
35         else:
36             pt_negative_reviews.append([review['content']])
37
38     with open('pt_positive_reviews.csv', 'a') as p:
39         write = csv.writer(p)
40         write.writerows(pt_positive_reviews)
41     with open('pt_negative_reviews.csv', 'a') as n:
42         write = csv.writer(n)
43         write.writerows(pt_negative_reviews)

```

---

Listagem 4: Ficheiro de configuração para criar o modelo de aprendizagem supervisionada

---

```

1 [paths]
2 train = "data/train.spacy"
3 dev = "data/test.spacy"
4 vectors = null
5 init_tok2vec = null
6
7 [system]
8 gpu_allocator = null
9 seed = 0
10
11 [nlp]
12 lang = "en"
13 pipeline = ["textcat_multilabel"]
14 batch_size = 1000

```

```

15 disabled = []
16 before_creation = null
17 after_creation = null
18 after_pipeline_creation = null
19 tokenizer = {"@tokenizers":"spacy.Tokenizer.v1"}
20
21 [components]
22
23 [components.textcat_multilabel]
24 factory = "textcat_multilabel"
25 scorer = {"@scorers":"spacy.textcat_multilabel_scorer.v2"}
26 threshold = 0.5
27
28 [components.textcat_multilabel.model]
29 @architectures = "spacy.TextCatBOW.v2"
30 exclusive_classes = false
31 ngram_size = 1
32 no_output_layer = false
33 nO = null
34
35 [corpora]
36
37 [corpora.dev]
38 @readers = "spacy.Corpus.v1"
39 path = ${paths.dev}
40 max_length = 0
41 gold_preproc = false
42 limit = 0
43 augmenter = null
44
45 [corpora.train]
46 @readers = "spacy.Corpus.v1"
47 path = ${paths.train}
48 max_length = 0
49 gold_preproc = false
50 limit = 0
51 augmenter = null
52
53 [training]
54 dev_corpus = "corpora.dev"
55 train_corpus = "corpora.train"
56 seed = ${system.seed}
57 gpu_allocator = ${system.gpu_allocator}
58 dropout = 0.1
59 accumulate_gradient = 1
60 patience = 1600
61 max_epochs = 0
62 max_steps = 20000
63 eval_frequency = 200
64 frozen_components = []
65 annotating_components = []
66 before_to_disk = null
67 before_update = null
68
69 [training.batcher]
70 @batchers = "spacy.batch_by_words.v1"
71 discard_oversize = false

```

## ANEXOS

```
72 tolerance = 0.2
73 get_length = null
74
75 [training.batcher.size]
76 @schedules = "compounding.v1"
77 start = 100
78 stop = 1000
79 compound = 1.001
80 t = 0.0
81
82 [training.logger]
83 @loggers = "spacy.ConsoleLogger.v1"
84 progress_bar = false
85
86 [training.optimizer]
87 @optimizers = "Adam.v1"
88 beta1 = 0.9
89 beta2 = 0.999
90 L2_is_weight_decay = true
91 L2 = 0.01
92 grad_clip = 1.0
93 use_averages = false
94 eps = 0.00000001
95 learn_rate = 0.001
96
97 [training.score_weights]
98 cats_score = 1.0
99 cats_score_desc = null
100 cats_micro_p = null
101 cats_micro_r = null
102 cats_micro_f = null
103 cats_macro_p = null
104 cats_macro_r = null
105 cats_macro_f = null
106 cats_macro_auc = null
107 cats_f_per_type = null
108
109 [pretraining]
110
111 [initialize]
112 vectors = ${paths.vectors}
113 init_tok2vec = ${paths.init_tok2vec}
114 vocab_data = null
115 lookups = null
116 before_init = null
117 after_init = null
118
119 [initialize.components]
120
121 [initialize.tokenizer]
```

---

### Listagem 5: Criação dos *datasets* de treino e teste

---

```
1 import spacy
2 from spacy.tokens import DocBin
```

```

3 from pandas import read_csv, DataFrame
4 import pandas as pd
5 from sklearn.model_selection import train_test_split
6 from sklearn.utils import shuffle
7
8
9 def getDoc(x,y):
10     db = DocBin()
11     for i, text in x.items():
12         if(type(text) != str):
13             continue
14         doc = nlp.make_doc(text)
15         positive = (y[i] == 1)
16         if(positive):
17             doc.cats = {'Positive': 1, 'Negative': 0 }
18         else:
19             doc.cats = {'Positive': 0, 'Negative': 1 }
20         db.add(doc)
21     return db
22
23 nlp = spacy.blank("pt")
24 negative = read_csv('pt_negative_reviews.csv')
25 positive = read_csv('pt_positive_reviews.csv')
26 negative['sentiment'] = 0
27 positive['sentiment'] = 1
28 total = pd.concat([negative, positive], ignore_index=True)
29
30 X = total['text']
31 y = total['sentiment']
32 X_train, X_test, Y_train, Y_test = train_test_split(X, y, test_size=0.05,
33     ↪ random_state=0)
34
35 dbTrain = getDoc(X_train, Y_train)
36 dbTrain.to_disk('data/train.spacy')
37
38 dbTest = getDoc(X_test, Y_test)
39 dbTrain.to_disk('data/test.spacy')

```

---

### Listagem 6: Micro serviço com *framework* FastAPI

---

```

1 import urllib
2 import spacy
3 from typing import Union
4 from fastapi import FastAPI
5 from fastapi.encoders import jsonable_encoder
6 from fastapi.responses import JSONResponse
7
8 app = FastAPI()
9 nlp = spacy.load("model-best")
10
11 @app.get("/analyse/{comment}")
12 def analyse_comment(comment: str, q: Union[str, None] = None):
13     commentU = urllib.parse.unquote_plus(comment, encoding='utf-8',
14     ↪ errors='replace')

```

```

14     doc = nlp(commentU)
15     json_compatible_item_data = jsonable_encoder({"cats": doc.cats})
16     return JSONResponse(content=json_compatible_item_data)

```

---

### Listagem 7: CLI para classificação de sentimentos através do modelo Spacy

---

```

1 import spacy
2
3 nlp = spacy.load("output/model-best")
4 while True:
5     print("#####]
6     ↪ ##")
7     print("Escreva um texto para classificação ou pressione 1 e ENTER para
8     ↪ sair")
9     r = input()
10    if r == '1':
11        break
12    doc = nlp(r)
13    print("As tags que o modelo prevê são: ")
14    print(doc)
15    sortedTags = sorted(doc.cats.items(), reverse=True, key=lambda x:x[1])
16    print("{:<15} {:<15}".format('tag', 'precision'))
17    i = 0
18    for t, p in sortedTags:
19        if i == 3:
20            break
21        print("{:<15} {:<15}".format(t, p))
22        i += 1

```

---

### Listagem 8: Ficheiro de exemplo para configuração de variáveis de ambiente

---

```

1 APP_HOSTNAME
2 PB_ADMIN_EMAIL
3 PB_ADMIN_PW
4 PB_FB_CLIENT_ID
5 PB_FB_CLIENT_SECRET
6 PB_SMTP_HOST
7 PB_SMTP_PW
8 PB_SMTP_PORT
9 PB_SMTP_USERNAME
10 PB_SMTP_TLS
11 PB_SMTP_AUTHMETHOD
12 PB_SMTP_SENDER_ADDRESS
13 PB_SMTP_SENDER_NAME
14 PB_FACEBOOK_API
15 PB_SENTIMENT_API
16 HTTPS_HOST_PORT
17 NEXT_PUBLIC_HTTPS_PUBLIC_PORT
18 NEXT_PUBLIC_APP_HOSTNAME
19 FACEBOOK_APP_SECRET
20 FACEBOOK_APP_ID
21 POCKET_BACKEND_URL

```

---

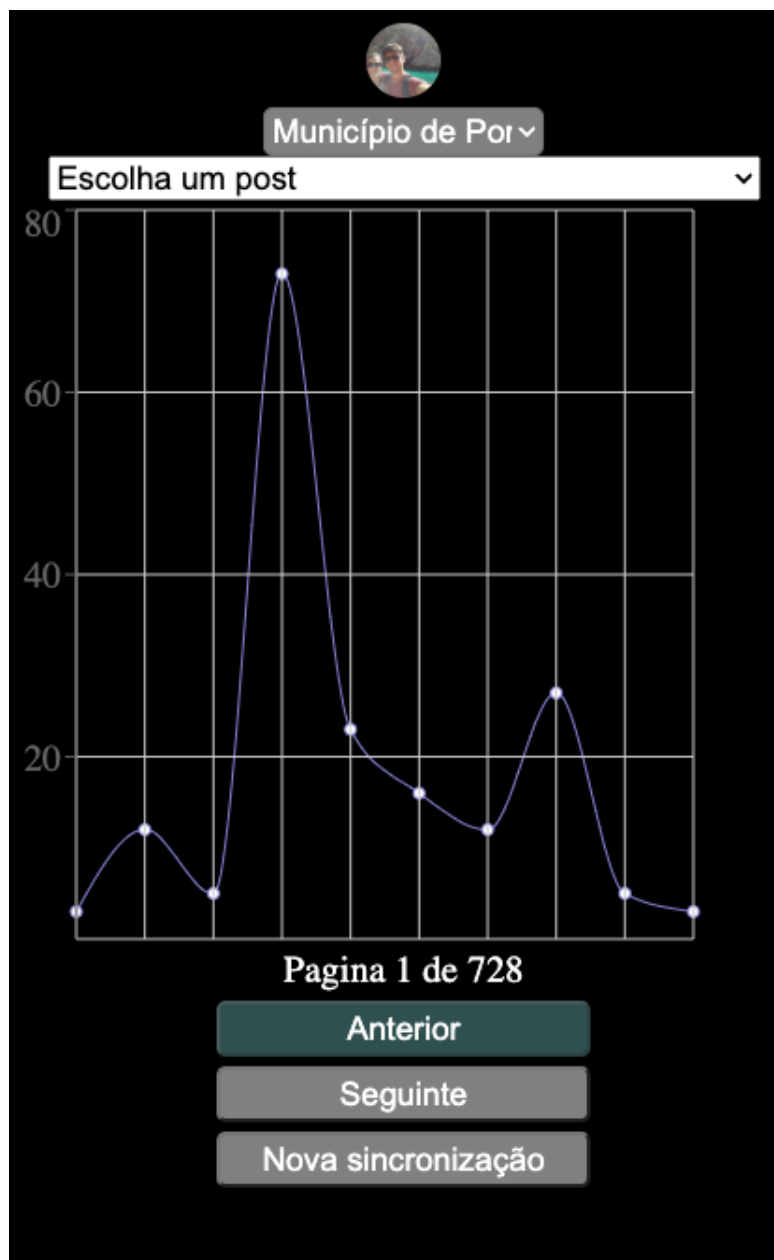
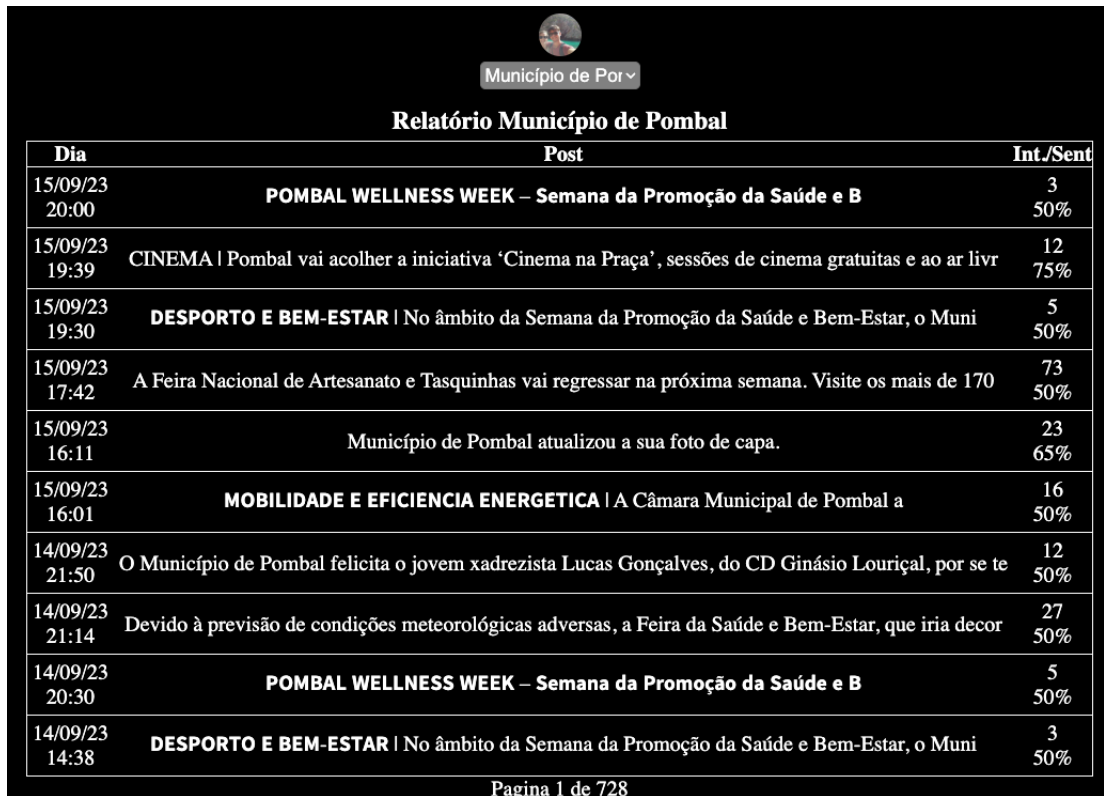


Figura 12: Gráfico temporal com os *posts* e métrica *engagement*



Dia	Post	Int./Sent
15/09/23 20:00	<b>POMBAL WELLNESS WEEK – Semana da Promoção da Saúde e B</b>	3 50%
15/09/23 19:39	CINEMA   Pombal vai acolher a iniciativa ‘Cinema na Praça’, sessões de cinema gratuitas e ao ar livre	12 75%
15/09/23 19:30	<b>DESPORTO E BEM-ESTAR</b>   No âmbito da Semana da Promoção da Saúde e Bem-Estar, o Muni	5 50%
15/09/23 17:42	A Feira Nacional de Artesanato e Tasquinhas vai regressar na próxima semana. Visite os mais de 170	73 50%
15/09/23 16:11	Município de Pombal atualizou a sua foto de capa.	23 65%
15/09/23 16:01	<b>MOBILIDADE E EFICIENCIA ENERGETICA</b>   A Câmara Municipal de Pombal a	16 50%
14/09/23 21:50	O Município de Pombal felicita o jovem xadrezista Lucas Gonçalves, do CD Ginásio Louriçal, por se te	12 50%
14/09/23 21:14	Devido à previsão de condições meteorológicas adversas, a Feira da Saúde e Bem-Estar, que iria decor	27 50%
14/09/23 20:30	<b>POMBAL WELLNESS WEEK – Semana da Promoção da Saúde e B</b>	5 50%
14/09/23 14:38	<b>DESPORTO E BEM-ESTAR</b>   No âmbito da Semana da Promoção da Saúde e Bem-Estar, o Muni	3 50%

Página 1 de 728

Figura 13: Relatório com médias da análise de sentimentos e *engagement*

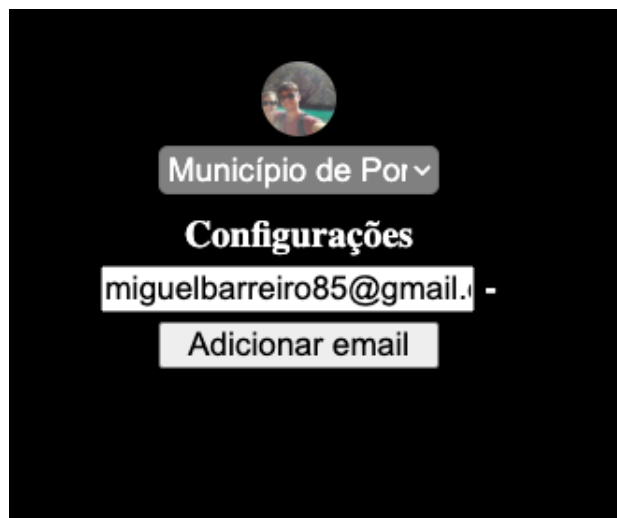


Figura 14: Página de configuração de emails para alertas

Município de Por

close

**Post:** A Feira Nacional de Artesanato e Tasquinhas vai regressar na próxima semana. Visite os mais de 170 expositores e usufrua das iguarias disponíveis nas 12 tasquinhas representativas do nosso território. Usufrua de todo o programa de animação preparado para si e assista em direto à emissão do programa "Aqui Portugal" que a RTP vai transmitir para todo o mundo os diferentes usos, costumes e tradições, acompanhados da atuação de artistas de renome nacional. Anote na sua agenda!

**Data:** 2023-09-15 16:42:14.000Z

**Partilhas:** 34

**ReactionsLove:** 1

**ReactionsLike:** 46

**ReactionsSad:** 1

**ReactionsTotal:** 48

**Comentários positivos:** 0

**Comentários negativos:** 0

Figura 15: Componente de detalhes de publicação



## DECLARAÇÃO

---

Declaro, sob compromisso de honra, que o trabalho apresentado nesta dissertação, com o título “*IMPLEMENTAÇÃO DE FERRAMENTA DE AVALIAÇÃO DE SATISFAÇÃO DE MUNÍCIPES ATRAVÉS DAS REDES SOCIAIS*”, é original e foi realizado por Luís Miguel da Silva Barreiro (2210781) sob orientação de Professor Nuno Filipe Pereira Salvador ([nuno.salvador@ipleiria.pt](mailto:nuno.salvador@ipleiria.pt)).

*Leiria, Setembro de 2023*

---

Luís Miguel da Silva Barreiro