Análise de publicações no Twitter nas Instituições do Ensino Superior do topo de Ranking Mundial

Tiago da Silva Coelho
Mestrado Integrado em Engenharia de Redes e Sistemas Informáticos
Departamento de Ciência de Computadores
2021

Orientador
Álvaro Figueira, Professor Auxiliar, Faculdade de Ciências da Universidade do Porto

U.PORTO

FACULDADE DE CIÊNCIAS
UNIVERSIDADE DO PORTO

Todas as correções determinadas pelo júri, e só essas, foram efetuadas.

O Presidente do Júri,

Porto, _____/_____/_____

Abstract

In recent years we have seen a large adherence to social media by various Higher Education Institutions (HEI) with the intent of reaching their target audiences and strengthen their brand recognition. These institutional publications are guided by a specific editorial strategy, designed to help them better accomplish their pre-established business goals. It is important for organizations to discover the true audience-aggregating themes resulting from their communication strategies, as it provides institutions with the ability to monitor their organizational positioning and identify opportunities and threats that, according to an internal evaluation of the strategies employed, may result in necessary strategic readjustments. In this dissertation we create an automatic system capable of identifying Higher Education Institutions Twitter communication strategies and the topics that emerge from their publication contents. Furthermore, with the current Covid-19 pandemic causing major consequences in many different fields (political, economic, social, educational) beyond the spread of the disease itself, we determine the impact of the pandemic on the identified HEI content strategies. We then gage the predictive capability of the obtained editorial models by attempting to predict the engagement and the dominant topic of a publication. We gathered and analyzed more than 18k Twitter publications from 12 of the top HEI according to the 2019 Center for World University Rankings (CWUR). Utilizing machine learning techniques, and topic modeling, we determined the emergent content topics across all HEI: Education, Faculty, Employment, Research, Health and Society. Then, we characterized the editorial strategy of each HEI before, and during, the Covid-19 pandemic and were able to observe significant differences between the specified periods. Lastly, with respect to the predictive capability of the models, we determine that the information gathered is not enough to be able to accomplish the proposed tasks.

Resumo

Nos últimos anos assistimos a uma grande adesão aos meios de comunicação social por parte de várias Instituições de Ensino Superior (HEI) com a intenção de atingir os seus públicos-alvo e fortificar o seu reconhecimento de marca. Estas publicações institucionais são orientadas por uma estratégia editorial específica, concebida para as ajudar a cumprir melhor com os seus objetivos empresariais preestabelecidos. É importante para as organizações a descoberta dos verdadeiros temas agregadores de público resultantes das suas estratégias de comunicação, na medida em que proporciona às instituições a capacidade monitorizar o seu posicionamento organizacional e identificar oportunidades e ameaças que, de acordo com uma avaliação interna das estratégias empregues, podem resultar em reajustes estratégicos necessários. Nesta dissertação criamos um sistema automático capaz de identificar as estratégias de comunicação no Twitter das instituições de ensino superior e os tópicos que emergem do conteúdo das suas publicações. Além disso, com a atual pandemia de Covid-19 a causar grandes consequências em muitas áreas diferentes (políticas, económicas, sociais, educacionais) para além da propagação da própria doença, determinamos o impacto da pandemia nas estratégias de conteúdo das HEI identificadas. Seguidamente, calculamos a capacidade de previsão dos modelos editoriais obtidos, tentando prever o envolvimento e o tópico dominante de uma publicação. Para tal, reunimos e analisámos mais de 18 mil publicações no Twitter de 12 das principais HEI de acordo com o Centro para o Ranking Mundial de Universidades (CWUR) de 2019. Utilizando técnicas de *machine learning*, e identificação de tópicos, determinámos os tópicos de conteúdo emergentes em todas as HEI: Educação, Corpo Docente, Emprego, Investigação, Saúde e Sociedade. De seguida, caracterizámos a estratégia editorial de cada HEI antes, e durante, a pandemia de Covid-19 e fomos capazes de observar diferenças significativas entre os períodos especificados. Finalmente, no que diz respeito à capacidade de previsão dos modelos, determinamos que a informação recolhida não é suficiente para poder realizar as tarefas propostas.

Agradecimentos

A conclusão deste projeto não seria possível sem a ajuda, assistência e apoio de muitas pessoas cujos nomes podem não ser todos enunciados. No entanto, a sua participação não foi esquecida e permanece um profundo reconhecimento à sua contribuição. Um agradecimento especial é dirigido ao meu supervisor Prof. Álvaro Figueira por todo o apoio e o discernimento que me prestou durante o desenvolvimento deste projeto. Quero agradecer também aos meus amigos, que desempenharam um papel decisivo, apoiando-me e dando-me motivação para nunca desistir. Não consigo agradecer suficientemente à minha família, em especial à minha mãe e as minhas irmãs, que me apoiaram durante todo o meu percurso académico e principalmente durante esta última fase de conclusão do meu ciclo de estudos.

Conteúdo

Lista de Tabelas

Lista de Figuras

Lista de Blocos de Código

Acrónimos

ANN Rede Neuronal Artificial

API Interface de Programação de Aplicações

BoW Bag-of-Words

CBOW Bag-of-Words Contínuo

CPU Unidade Central de Processamento

CWUR Centro para o Ranking Universitário Mundial

K-NN K-Nearest Neighbors

KPI Indicadores Chave de Performance

LDA Alocação de Dirichlet Latente

LR Regressão Logística

LSA Análise Semântica Latente

MAE Erro Médio Absoluto

MLP Perceptron Multicamadas

MSE Erro Quadrático Médio

NLP Processamento de Linguagem Natural

NMF Fatorização Matricial Não Negativa

OMS Organização Mundial de Saúde

pLSA Análise Probabilística de Semântica Latente

POS Part of Speech

RMSE Raíz do Erro Quadrático Médio

SVM Máquinas de Vetores de Suporte

TDIDT Top Down Induction of Decision Trees

TF-IDF Term Grequency–Inverse Document Frequency

UTC Tempo Universal Coordenado

UTF Formato de Transformação Unicode

Capítulo 1

Introdução

1.1 Contextualização e Motivação

Atualmente, a nossa sociedade encontra-se numa era digital, onde a tecnologia é alvo de imensos avanços e cada vez mais se acelera o seu desenvolvimento. Nesta era digital, deu-se a utilização generalizada da Internet e o aparecimento das plataformas online para a comunicação interpessoal. Embora a utilização inicial destas plataformas de comunicação social incidisse essencialmente na interação entre amigos e familiares, com a sua maturação, os seus utilizadores começaram a utilizar estas plataformas para um número cada vez maior de finalidades (partilhar notícias, organizar eventos, marketing, etc). Um grupo específico de utilizadores, as empresas/organizações, serviram-se destas tecnologias web polivalentes para revolucionar o processo de comunicação com os seus diversos públicos [72]. Atualmente, a variedade de ferramentas de comunicação à disposição das organizações permitem a formação de relações diretas com os públicos que pretendem atingir, contrastando com o anterior modelo segundo o qual se usava a intervenção de um intermediário entre as mesmas e o público alvo (TV, rádio, jornal, etc).

Uma das ferramentas que mais contribuiu para o panorama atual de gestão de comunicação foram essas redes online, que tiveram uma grande taxa de adesão por parte de indivíduos. As organizações têm vindo a reconhecer o potencial dessas plataformas e, como tal, têm seguido os seus públicos na adesão às mesmas, sendo que muitas delas o fazem de acordo com uma estratégia bem definida, construída de forma a conseguirem cumprir objetivos empresariais previamente estabelecidos [24]. Neste processo de adesão, recursos como tempo, esforço, tecnologia e competências vão sendo alocados cada vez mais, levantando a necessidade de legitimar este investimento no contexto de desenvolvimento organizacional. Para além destes fatores, é também importante para as organizações a descoberta dos verdadeiros temas agregadores de público resultantes das suas estratégias atuais. É através deles que o público forma uma perceção das instituições e, como tal, apenas através da sua identificação se torna possível dar mais relevo a esses temas numa nova estratégia e promover uma melhor imagem publica.

As Instituições de Ensino Superior estão também inseridas neste contexto. Com o aumento

da competitividade e das expectativas de atuais e futuros alunos e a diminuição de apoios governamentais e do número de novas inscrições, as instituições necessitam cada vez mais de agir de modo a garantir subsistência [56]. Como tal, passa a ser vital para uma Instituição de Ensino Superior conseguir financiamento adicional e isso passa por concentrar os seus esforços na obtenção de uma vantagem competitiva sob forma de diferenciação, alcançada através de ações de autopromoção e de maior interação com públicos mais amplos. Campanhas vantajosas nas redes sociais são os mais significativos condutores de fidelidade a uma instituição [93].

A diferenciação é um dos aspetos mais difíceis de obter para uma instituição de ensino superior (Higher Education Institution - HEI) visto que o produto que oferecem pode ser considerado semelhante a muitos dos seus competidores. A adesão das Instituições de Ensino Superior às redes sociais equipa os seus respetivos departamentos de comunicação com a capacidade de fortificar o reconhecimento da marca e melhorar a sua imagem através da comunicação direta com os seus diversos públicos, resultando na diferenciação pretendida. Esses públicos consistem em estudantes, os seus pais, investidores, docentes, investigadores, etc, e a interação com todos eles levanta diferentes necessidades de comunicação e gestão. De acordo com a estratégia ou modelo de comunicação empregue, as HEI tentam satisfazer estas necessidades de modo a serem mais competitivas no mercado global. Apesar de possuírem objetivos em comum, as estratégias, na sua especificidade, podem variar significativamente de instituição para instituição, levando a resultados diferentes.

Estratégias de conteúdo para redes sociais destinadas a organizações com tal diversidade de expectativas, distinção de serviços, intervenção esperada pela sociedade e correspondente pressão externa carecem de investigação. É assim importante pesquisar modos de intervenção que possam proporcionar o equilíbrio entre as suas necessidades institucionais e transacionais, para assegurar a sua sobrevivência e potencial competitivo [64].

Embora seja possível definir vertentes das publicações nas plataformas online comuns a instituições de forma generalizada através das suas necessidades partilhadas (imagem, ensino, investigação, missão social, etc), a análise da estratégia específica de cada instituição torna-se difícil devido ao elevado volume de publicações que são necessárias explorar de modo a identificar a forma como incidem sobre essas vertentes. A quantidade de informação a analisar seria suficiente para sobrecarregar qualquer individuo, o ser humano simplesmente não tem capacidade biológica para fazer esse tipo de análise de forma manual e autónoma, ou seja, capacidade para guardar e catalogar essa quantidade de informação, assim como fazer uma análise comparativa em tempo útil da mesma, com informação proveniente de outras instituições, resultante de outros modelos de comunicação. Esta análise é importante na medida em que proporciona às instituições a capacidade monitorizar o seu posicionamento organizacional e identificar oportunidades e ameaças que, de acordo com uma avaliação interna das estratégias empregues, podem resultar em reajustes estratégicos necessários [34].

1.2 Questões de Investigação

Nesta secção serão expostas as questões que sumarizam os principais problemas previamente contextualizados em análise ao longo da dissertação que a seguir se listam:

- Quais as estratégias empregues pelas HEI na comunicação através da rede social Twitter?
- Quais as estratégias com melhores resultados de interação/envolvimento?
- Quais as áreas editoriais de maior incidência? Quais as semelhanças estratégicas entre as instituições de acordo com essas áreas editoriais?
- Qual a influência do Covid-19 na comunicação das HEI?
- Será comum uma distribuição uniforme das publicações por área editorial ou existe uma baixa diversidade estratégica?
- Dado um Tweet de uma HEI, qual será a sua interação/envolvimento?
- Será possível prever a popularidade da publicação de uma HEI com base no seu conteúdo?
- Será possível prever o tópico da próxima publicação de uma HEI com base nas suas publicações anteriores?

1.3 Objetivos

De modo a responder às questões anteriores, nesta dissertação, pretendemos criar um sistema automático capaz de obter a estratégia de comunicação de uma determinada HEI, verificar as semelhanças estratégicas entre instituições, determinar o impacto da Covid-19 nas estratégias de comunicação identificadas, caracterizar as áreas editoriais com maior envolvimento por parte do publico e, por fim, determinar a possibilidade de prever a interação futura de publicações ou a área editorial de publicações, com base no seu conteúdo, ou conteúdo de publicações que as antecedem.

Numa fase inicial, após a devida aplicação de técnicas de análise exploratória começamos por nos servir de métodos de *machine learning* não supervisionados para realizar tarefas de identificação de tópicos e determinar as áreas editoriais presentes no conteúdo das publicações recolhidas.

De seguida, caracterizamos as estratégias de publicação de cada HEI consoante as áreas editoriais presentes no conteúdo da sua coleção de Tweets e realizamos uma análise comparativa das estratégias resultantes.

Por fim, realizamos a categorização de cada Tweet da nossa base de dados de acordo com o seu conteúdo e exploramos os diferentes modelos de aprendizagem supervisionada para obter o

melhor modelo de previsão de futura interação para novas publicações e o melhor modelo de previsão do próximo tópico a ser abordado numa publicação.

1.4 Organização da tese

A dissertação está dividida da seguinte forma: o capítulo atual apresenta uma breve introdução e contextualização do problema assim como a explicita a motivação para desenvolver a nossa solução e identifica os objetivos propostos e, de uma forma geral, a metodologia como pretendemos atingi-los. Neste capítulo é descrita a realidade atual do uso das redes socais como ferramentas de comunicação por parte das Instituições de Ensino Superior, o que motiva as instituições a comunicarem através deste meio, a importância da análise de estratégias de comunicação na obtenção dos resultados pretendidos e também o problema de grandes quantidades de informação não serem facilmente manipuláveis sem o uso de ferramentas de processamento automático. No capítulo 2 são apresentados conceitos de *machine learning* fundamentais à compreensão do trabalho desenvolvido ao longo da dissertação. De seguida, no capítulo 3, é realizamos uma revisão bibliográfica de literatura complementar assente no mesmo tema da nossa dissertação, explicamos o critério segundo o qual essa revisão foi realizada e discutimos as diferenças e benefícios da nossa abordagem em comparação com abordagens anteriores. No capítulo 4 é exposta em detalhe a metodologia utilizada durante o desenvolvimento e no capítulo 5 apresentamos todo o processo de desenvolvimento e descrevemos o trabalho realizado de modo a cumprir com os objetivos propostos. No capítulo 6 fazemos a análise dos resultados obtidos e, por fim, no capítulo 7 apresentamos as conclusões da dissertação, as limitações a ela associadas e as possibilidades de desenvolvimento futuro.

1.5 Dados utilizados

O Centro para o Ranking Universitário Mundial (CWUR)[1] publica anualmente a Classificação Internacional das Universidades, que mede um conjunto de parâmetros relativos à educação e ao prestígio da instituição, do corpo docente e da investigação produzida. O CWUR utiliza sete indicadores para classificar as 1000 melhores universidades do mundo: Qualidade da Educação, Empregabilidade de Ex-Alunos, Qualidade da Universidade, Número de Publicações, Qualidade das Publicações, Influência e Citações. Para a nossa investigação utilizámos o ranking disponibilizado para o período de 2019/2020 [4] e selecionamos 12 das 30 instituições mais bem posicionadas com contas do Twitter ativas, formando assim uma amostra representativa das melhores universidades do mundo.

De seguida, com o uso da Twitter API [10], foi feita a recolha de todas as publicações de cada uma das instituições selecionadas entre o período de 1 de fevereiro 2019 e 31 de Outubro de 2020. Estes tweets resultam em mais 18k publicações diferentes e é com base neste conjunto que

[1] https://cwur.org/

realizamos o nosso trabalho. Apesar de nem todas as HEI começarem e terminarem o seu ano letivo exatamente nestas datas, acreditamos que este período é o mais adequado na representação do ano letivo para a maioria das instituições.

Capítulo 2

Conceitos Fundamentais

2.1 Introdução a Machine Learning

O ramo de *machine learning* possui várias definições formais na literatura. Arthur Samuel [80] definiu *machine learning* como um ramo de estudo que proporciona aos computadores a capacidade de aprender sem qualquer programação explícita. No contexto de ciências de computação, Tom Mitchell [55] apresentou o ramo da seguinte forma: Um programa de computador é dito que aprende com a experiência (E) a respeito de uma classe de tarefas (T) e métricas de performance (P) se o seu desempenho na execução de tarefas em T, tal como medido por P, melhora com a experiência E. Ethem Alpaydin no seu livro 'Introduction to machine learning' [15] define *machine learning* como o ramo de programação de computadores que otimizam um critério de avaliação utilizando dados de exemplo ou experiência prévia. Estas várias definições partilham a noção do treino intencional de computadores para a realização de tarefas de forma inteligente que vão além do tradicional processamento de números através da aprendizagem do ambiente que os rodeia utilizando a repetição de exemplos.

Machine learning consiste essencialmente na tecnologia do desenvolvimento de algoritmos computacionais capazes de emular alguma forma de inteligência. Baseia-se em ideias de diferentes disciplinas como inteligência artificial, probabilidade e estatística, ciência de computadores, teoria da informação, psicologia, teoria de controlo e filosofia [15, 19, 55]. Esta tecnologia tem sido aplicada em áreas tão diversas como reconhecimento de padrões [55], visão computacional [17], engenharia aeroespacial [16], finanças [39], entretenimento [38, 92], ecologia [33], biologia computacional [57, 91] e aplicações biomédicas [29, 50]. A propriedade mais importante destes algoritmos é a sua capacidade distintiva de aprender o ambiente que os rodeia a partir de dados de input.

Um algoritmo de *machine learning* é um processo computacional que utiliza dados de input para alcançar um objetivo desejado sem necessidade de ser 'hard coded' para produzir um determinado resultado. Estes algoritmos são, de certa forma, 'soft coded' visto que alteram ou adaptam automaticamente a sua arquitetura através da repetição (ou seja, experiência) de modo

a tornarem-se cada vez melhores na realização da tarefa predefinida. O processo de adaptação tem o nome de treino, no qual são fornecidas amostras de dados de input juntamente com os resultados desejados. O algoritmo, então, configura-se otimamente de modo a poder não só produzir o resultado desejado quando apresentado com os dados de input de treino, mas também generalizar para produzir o resultado desejado a partir de novos e nunca antes observados dados de input. A fase de treino não necessita ser limitada a uma adaptação inicial durante um intervalo de tempo finito. Tal como com os seres humanos, um bom algoritmo pode praticar a aprendizagem ao longo do seu 'tempo de vida', ou seja, à medida que processa novos dados e aprende com erros prévios.

Existem diversas formas segundo as quais um algoritmo computacional se pode adaptar como resposta à sua fase de treino. Por exemplo, os dados de input podem ser selecionados e pesados de modo a fornecer os mais decisivos resultados. O algoritmo pode possuir parâmetros numéricos variáveis, ajustáveis através da otimização iterativa. Pode possuir uma rede de caminhos computacionais possíveis que organiza de modo a obter resultados ótimos ou pode até determinar as distribuições de probabilidade a partir dos dados de input e utilizá-los para prever resultados.

A capacidade de aprender através de dados de input do ambiente circundante é um dos aspetos principais no desenvolvimento de uma aplicação de *machine learning* de sucesso. Neste contexto, a aprendizagem é definida como uma estimativa das dependências a partir dos dados [27]. As áreas de *data mining* e *machine learning* estão altamente relacionadas. *Data mining* é um processo que incorpora dois elementos: a base de dados e *machine learning*. O primeiro fornece técnicas de gestão de dados, enquanto o segundo fornece técnicas de análise de dados. Assim, enquanto *data mining* necessita de *machine learning*, *machine learning* não necessita necessariamente de *data mining*. Apesar disso, vários algoritmos de *machine learning* utilizam métodos de *data mining* no pré-processamento dos dados antes da aprendizagem das tarefas desejadas [45].

O ramo de *machine learning* pode ser dividido de acordo com a natureza da rotulagem dos dados de input em aprendizagem supervisionada, não supervisionada, semi-supervisionada, como mostra a figura 2.1. A aprendizagem supervisionada é utilizada na estimação de um mapeamento desconhecido (input, output) através de amostras conhecidas (input, output) onde o output é catalogado. Na aprendizagem não supervisionada, apenas são fornecidas amostras de input ao sistema durante o treino. A aprendizagem semi-supervisionada é uma combinação das abordagens anteriores, onde parte dos dados são parcialmente catalogados e essa mesma parte é utilizada para inferir a parte restante.

Numa perspetiva de *concept learning*, *machine learning* pode ser categorizada em aprendizagem transdutiva e indutiva [85]. A aprendizagem transdutiva envolve a inferência de casos de treino específicos para casos de teste específicos utilizando rótulos discretos ou contínuos. Em contrapartida, a aprendizagem indutiva ambiciona a previsão de outputs através de dados de input aos quais o sistema não tenha estado exposto. Seguindo este raciocínio, Mitchell [54] defende a necessidade da existência de um viés (*bias*) indutivo no processo de treino de modo a

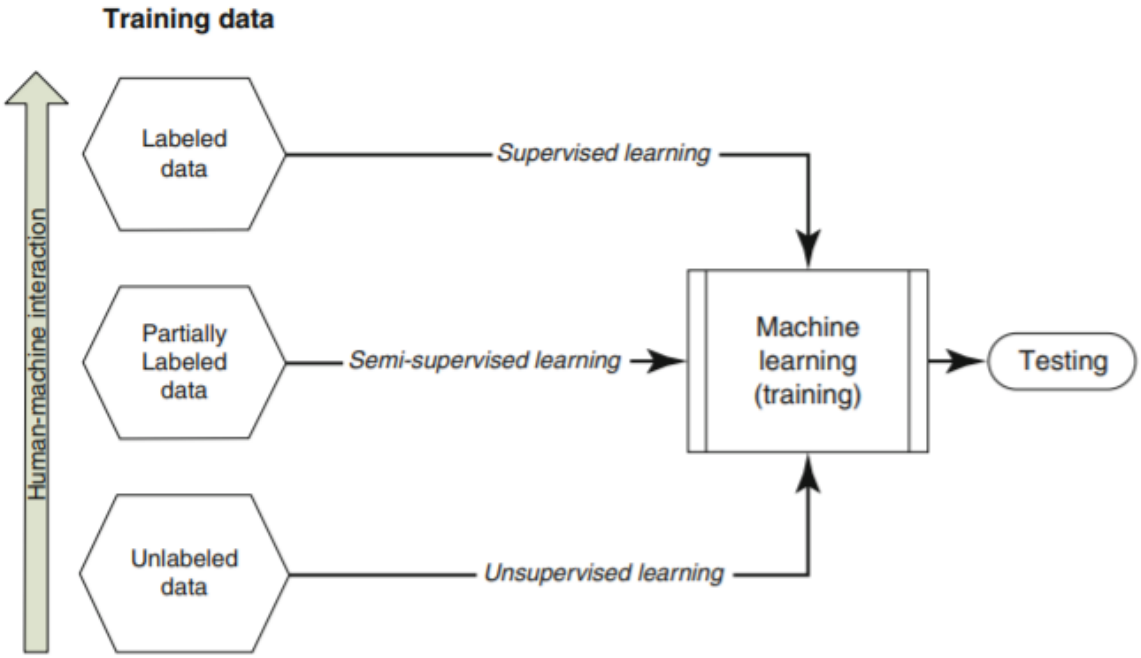

Figura 2.1: Categorias de algoritmos de *machine learning* de acordo com os seus dados de input [31]

permitir o algoritmo de *machine learning* de generalizar para além dos dados observados. De um ponto de vista probabilístico, os algoritmos de *machine learning* podem ser divididos em modelos discriminantes ou generativos. Um modelo discriminante mede a probabilidade condicional de um output dado um input tipicamente determinístico. Por outro lado, um modelo generativo é completamente probabilístico, no sentido em que utiliza distribuição de probabilidade conjunta para fazer as suas previsões.

Para uma aplicação bem sucedida de *machine learning* é primeiramente necessário uma boa caracterização do problema e da sua natureza em termos dos dados de input e de output. Em segundo lugar, apesar da robustez ao ruído, um bom modelo não consegue substituir uma má qualidade de dados, visto que é primeiramente construído sobre aproximações aos dados. Nas palavras de George Box: 'All models are wrong; some models are useful' [22]. Por fim, o modelo necessita ser capaz de generalizar para além dos dados observados. Para cumprir com esse objetivo, o modelo necessita ser mantido tão simples quanto possível mas não ser tornado mais simples, uma qualidade se segue o princípio de *Occam's razor* segundo o qual, entre hipóteses concorrentes, deve ser selecionada a hipótese com menor número de premissas/suposições.

O erro de viés/*bias* é resultante de suposições erradas no algoritmo de aprendizagem. Um alto viés pode fazer com que um algoritmo não identifique corretamente relações relevantes entre os atributos e as variáveis-alvo (*underfitting*).

O erro de variância resulta da sensibilidade a pequenas flutuações no conjunto de treino. Uma alta variância pode resultar na modelação de ruído aleatório nos dados de treino (*overfitting*).

O dilema de viés-variância é o conflito que ocorre na tentativa de, simultaneamente, minimizar estas duas fontes de erro que impedem algoritmos de aprendizagem supervisionada de generalizar para além dos seus dados de treino [46, 86].

Se o nosso modelo é demasiado simples em comparação com função a modelar, ele pode possuir um alto viés e baixa variância, o modelo não se ajusta corretamente aos dados e ocorre o *underfit*, onde o modelo se revela incapaz de modelar tanto os dados de treino como novos dados nunca antes vistos. Por outro lado, se, como resposta, a complexidade for aumentada e se o modelo se verificar demasiado complexo, este poderá possuir um baixo viés e alta variância, ficando sujeito a dar *overfit* e perder a capacidade de generalizar corretamente. Para otimizar a performance no contexto da generalização, a complexidade da hipótese deve corresponder à complexidade da função subjacente aos dados.

A secção que se segue pretende ser uma introdução em maior detalhe de alguns dos conceitos mencionados previamente assim como novos conceitos, considerados essenciais para a compreensão do trabalho realizado no contexto desta dissertação. Consiste numa breve introdução às métricas e métodos de avaliação de modelos de *machine learning*, introdução à área de processamento de linguagens naturais, as suas tarefas e aplicações, aprendizagem supervisionada, aprendizagem não supervisionada e os diversos algoritmos utilizados nestes dois contextos.

2.2 Avaliação de Modelos

A forma como um modelo generaliza para novas observações é um aspeto importante que deve ser considerado. São necessários mecanismos que avaliem essa capacidade de generalização de modo a conseguirmos verificar se o modelo funciona corretamente e se podemos confiar nas suas previsões. Nesta secção, apresentamos as técnicas utilizadas na avaliação de modelos de *machine learning* e ilustramos como as métricas de avaliação mais comuns são implementadas em problemas de classificação e regressão.

Os métodos de avaliação de modelos podem ser divididos em duas categorias, nomeadamente métodos de *holdout* e *cross-validation*. Ambos os métodos utilizam um conjunto de dados de teste (i.e. dados nunca antes observados pelo modelo) na avaliação da performance do modelo. Não é recomendável a utilização dos dados fornecidos ao modelo durante o seu treino (conjunto de dados de treino) na avaliação do seu desempenho, isto porque, como o modelo teve acesso a esses dados na sua aprendizagem, a suas previsões não serão representativas da sua capacidade de generalizar, serão sempre corretas.

O objetivo da validação *holdout* é a testagem do modelo em dados diferentes daqueles onde o modelo foi treinado, visto que esta fornece uma estimativa imparcial do desempenho da aprendizagem. Este método divide o conjunto de dados original em 3 subconjuntos:

- O **conjunto de treino** constitui o subconjunto utilizado na aprendizagem dos modelos preditivos.

- O **conjunto de validação** consiste num subconjunto de dados utilizados na avaliação da performance preditiva do modelo criado na fase de treino. Fornece uma plataforma capaz de aperfeiçoar os parâmetros do modelo e selecionar o modelo com melhor performance-
- O **conjunto de teste** é o subconjunto de dados não observados pelo modelo na fase de treino que são utilizados para estimar o desempenho provável do modelo no futuro. Se um modelo se ajusta muito melhor ao conjunto de treino do que ao conjunto de teste, o modelo pode ser ter sido alvo de *overfitting*.

A validação *holdout* é útil em várias situações graças à sua rapidez, simplicidade e flexibilidade. No entanto, esta técnica é normalmente associada a uma grande variabilidade, visto que diferenças no conjunto de treino e de teste podem resultar em diferenças significativas na estimativa de desempenho.

Á semelhança da validação *holdout*, a *cross-validation* é uma técnica que envolve a divisão do conjunto de observações original num conjunto de treino, utilizado para treinar o modelo e um conjunto de validação utilizado para avaliar o modelo. Existem várias vertentes desta técnica, sendo que a mais comum é a *k-fold cross validation*, onde o conjunto de dados original é dividido em k subconjuntos do mesmo tamanho, sendo k um valor especificado pelo utilizador. Dos k subconjuntos criados, apenas um é retido como o conjunto de validação e os restantes são utilizados como dados de treino. O processo de *cross-validation* é repetido k vezes, com cada um dos k subconjuntos utilizados exatamente uma vez como o conjunto de validação. A estimativa do desempenho do modelo pode ser calculado através da média do k resultados obtidos. Como se pode observar, cada instância de dados pertence ao conjunto de teste exatamente uma vez e pertence ao conjunto de treino k-1 vezes, o que reduz significativamente o viés/*bias*, visto que estamos a utilizar todos os dados para aprendizagem, e também reduz a variância, já que a todos dados estão também a ser utilizados no conjunto de validação.

Para conseguir quantificar o desempenho de um modelo é necessário utilizar métricas de avaliação. A escolha da métrica de avaliação depende da natureza da tarefa a executar (como classificação, regressão, *clustering*, identificação de tópicos, etc). As tarefas de aprendizagem supervisionada de classificação e regressão constituem a maioria das aplicações de *machine learning* e foram dois dos três tipos de tarefas abordados no nosso projeto. Como tal, vamos incidir sobre as diferentes métricas existentes para este tipo de problemas (as métricas de avaliação de problemas de identificação de tópicos serão apenas abordadas na secção 2.4.1).

2.2.1 Métricas de Classificação

Uma **matriz de confusão** (também conhecida por matriz de erro) é uma tabela específica que facilita a visualização da performance de um algoritmo. Cada linha desta matriz representa as instâncias pertencentes a uma classe real enquanto cada coluna representa as instâncias pertencentes uma classe prevista, ou vice versa [73]. O seu nome surge da facilidade que proporciona em observar se o modelo 'confunde' as diferentes classes, ou seja, se o modelo

frequentemente classifica instâncias pertencentes a uma classe como pertencentes a outra). A matriz possui duas dimensões ('classificação real' e 'classificação prevista') e conjuntos idênticos de classes nas duas dimensões (cada combinação de uma destas dimensões e classes representa uma nova variável na matriz).

Em termos de sensibilidade e especificidade, a matriz de confusão, no caso de uma variável alvo binária, é representada da seguinte forma:

		Classificação Real	
		P	N
Classificação Prevista	P	VP	FP
	N	FN	VN

Tabela 2.1: Representação da matriz de confusão

P corresponde à condição positiva, o número de casos positivos no conjunto de dados e N corresponde à condição negativa, ou seja, o número de casos negativos no conjunto de dados. VP corresponde a 'verdadeiros positivos', i.e. o número de casos com classificação real positiva cuja classificação prevista é também positiva (casos positivos classificados corretamente) e FP a 'falsos positivos', o número de casos classificados como positivos pelo modelo mas que na realidade possuem uma classificação negativa. Do mesmo modo, FN (falsos negativos) corresponde ao número de casos classificados como negativos pelo modelo mas que na realidade possuem classificação positiva e VN (verdadeiros positivos) corresponde aos casos classificados como negativos pelo modelo com classificação real negativa (casos negativos classificados corretamente).

A **sensibilidade** (ou *recall*) de um modelo mede a proporção de casos positivos que foram corretamente identificados em relação a todos os casos identificados como positivos,

$$Recall = \frac{TP}{P} \tag{2.1}$$

enquanto a **especificidade** mede a proporção de casos negativos que foram corretamente identificados em relação a todos os casos identificados como negativos.

$$Especificidade = \frac{TN}{N} \tag{2.2}$$

A **precisão** mede a proporção de casos classificados que foi identificada corretamente, e é calculada através de

$$Precisão = \frac{TP}{TP + FP} \tag{2.3}$$

A **exatidão** (ou *accuracy*, em inglês) mede a proporção do número de previsões corretas em relação ao número total de previsões efetuadas:

$$Exatidão = \frac{TP + TN}{P + N} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.4}$$

Por fim, o temos o **F1 *score*** que pode ser interpretado como a média harmónica da precisão e da sensibilidade e possui um alcance de [0,1]. Esta métrica tenta encontrar o melhor equilíbrio entre a precisão e a sensibilidade. Quanto maior o valor do F1 *score*, melhor a sua performance. Matematicamente, pode ser representado pela expressão:

$$F1 = \frac{2 * precisão * sensibilidade}{precisão + sensibilidade} = \frac{2TP}{2TP + FP + FN} \tag{2.5}$$

2.2.2 Métricas de Regressão

O **erro quadrático médio**, ou *mean squared error* em inglês (MSE) é uma métrica de erro comum para problemas de regressão. O MSE é calculado como a média do quadrado das diferenças entre os valores previstos e os valores reais da variável-alvo no determinado conjunto de dados

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \tilde{Y}_i)^2 \tag{2.6}$$

onde Y_i representa o i-ésimo valor real no conjunto de dados, o $\tilde{Y}_i$ o i-ésimo valor previsto e o n o número de observações no conjunto de dados. A diferença entre estes dois valores (Y_i e $\tilde{Y}_i$) é colocada ao quadrado, o que resulta num valor final de erro sempre positivo. A elevação ao quadrado possui também o efeito de magnificação de erros maiores. Isto é, quanto maior a diferença entre o valore previsto e o valore real, maior será o erro resultante do quadrado da sua diferença. Desde modo, o MSE penaliza modelos com erros maiores.

As unidades do MSE são unidades ao quadrado. Por exemplo, se a variável-alvo de um problema de regressão está expressa em euros ou metros, o MSE retorna um valor em 'euros quadrados' ou metros quadrados. Esta caraterística pode afetar a interpretabilidade do valor de erro, tornando-o confuso de entender e contextualizar. Normalmente, calcula-se a raiz quadrada do MSE que retorna um valor nas mesmas unidades da variável-alvo e resolve este problema. Esta nova métrica tem o nome de **raiz do erro quadrático médio**, ou *root mean squared error* em inglês (RMSE) e funciona como uma extensão do MSE.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i - \tilde{Y}_i)^2} \tag{2.7}$$

O **erro médio absoluto**, ou *mean absolute error* em inglês (MAE) é uma métrica de avaliação popular porque, tal como o RMSE, as unidades do erro correspondem às unidades da variável-alvo.

Isto é, tanto o MSE como o RMSE penalizam erros maiores mais do que penalizam erros menores, inflacionando o valor de erro médio através do uso da elevação ao quadrado dos valores intermédios. Por outro lado, o MAE não penaliza mais ou menos os diferentes tipos de erros e graças a isso os valores aumentam de forma linear de acordo com o aumento do erro.

Como o seu nome sugere, o valore de MAE é calculado como a média dos valores absolutos dos valores de erro. Como tal, a diferença entre o valor real e o valor previsto pode resultar valores positivos e negativos mas ser forçada a assumir valores positivos durante o cálculo.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|Y_i - \tilde{Y}_i| \tag{2.8}$$

Por último, temos o coeficiente de determinação, denotado por R^2, que mede a proporção da variância na variável-alvo dependente que é possível prever a partir dos restantes atributos. Deste modo, o R^2 mede a força da relação entre o modelo e a variável-alvo numa escala de 0 a 1. Valores de R^2 fora do alcance [0,1] podem ocorrer quando o modelo treinado se ajusta aos dados de forma pior do que um hiperplano horizontal (a hipótese nula), ou seja, o modelo tem previsões piores do que simplesmente prever o valor médio da variável-alvo em cada instância. Isto pode acontecer devido a uma escolha errada do modelo e das restrições que lhe foram aplicadas.

O R^2 pode ser calculado através da expressão seguinte, onde $\bar{Y}$ representa o valor médio da variável-alvo:

$$R^2 = \frac{\text{Variância explicada pelo modelo}}{\text{Variância total}} = \frac{\sum(\tilde{Y}_i - \bar{Y})^2}{\sum(Y_i - \bar{Y})^2} \tag{2.9}$$

2.3 Aprendizagem Supervisionada

A aprendizagem supervisionada é a tarefa de *machine learning* que incide na aprendizagem de uma função que mapeia um input a um output com base em exemplos de pares input-output [79]. Na aprendizagem supervisionada, cada exemplo é um par constituído por um objeto de input e um valor de output desejado. Um algoritmo de aprendizagem supervisionada analisa os dados etiquetados de treino, $D = \{(x1, y1), ..., (xn, yn)\}$ (onde x_i representa o vetor de atributos da observação i e y_i o seu valor de output) , com o objetivo de inferir uma função

$$y = f(x, \beta) + \epsilon. \tag{2.10}$$

que representa os padrões ocultos nos dados e que pode ser usada para mapear novos exemplo. β representa os parâmetros desconhecidos no mapeamento a ser estimados e ϵ o erro associado a esta estimação. Se y for numérico, $y \in \mathbb{R}$, o problema é considerado como um problema de regressão e se o y for categórico, o problema é considerado um problema de classificação. Devido à natureza das nossas variáveis alvo (categóricas e numéricas) nos diferentes problemas que nos

propusemos a resolver, executamos tanto tarefas de regressão e classificação ao longo do nosso desenvolvimento. Na secção seguinte apresentamos os algoritmos diferentes de aprendizagem supervisionada utilizados.

2.3.1 Regressão Linear

A regressão linear é uma abordagem linear na modelação da relação entre uma resposta escalar e uma ou mais variáveis explicativas (também conhecidas como variáveis independentes). Este método é aplicável quando tanto a variável alvo e as variáveis explicativas/atributos são numéricos. Para o caso de apenas uma variável explicativa o processo assume o nome de regressão linear simples enquanto para mais do que uma variável, o processo é chamado de regressão linear múltipla. O modelo é expresso como uma combinação linear dos atributos (variáveis explicativas) com pesos calculados a partir dos dados de treino,

$$y = x\beta + \epsilon \tag{2.11}$$

onde o x representa o vetor de observações de treino, β os parâmetros desconhecidos da regressão e ϵ o erro aleatório associado. Podemos ainda representar esta expressão como

$$y = w_0 + w_1x_1 + w_2x_2 + ... + w_nx_n + \epsilon \tag{2.12}$$

onde $x_1, x_2, ..., x_n$ são os valores do vetor de atributos e $w_0, w_1, w_2, ..., wn$ os pesos calculados. Sendo $y^{(1)}$ o valor de output para a primeira instância e $x_1^{(1)}, x_2^{(1)}, ..., x_n^{(1)}$ os valores dos atributos dessa mesma instância, o *predicted value* para a primeira instância pode ser obtido através de

$$\textit{Predicted Value}_{(1)} = w_0 + w_1x_1^{(1)} + w_2x_2^{(1)} + ... + w_nx_n^{(1)} = \sum_{j=0}^{n} w_jx_j^{(1)} \tag{2.13}$$

A regressão linear seleciona os valores de $w_0, w_1, ..., w_n$ que minimizam a diferença entre o valor real e os *predicted values* para todas as instâncias de treino. Normalmente, os modelos de regressão linear obtêm essa diferença através do método de *least squares*, mas é possível também utilizar outros métodos como a minimização de uma versão penalizada da função de *least squares*, ao qual se dá o nome de regressão de ridge (penalizando a norma de L^1) e regressão de lasso (penalizando a norma de L^2).

Este método revela-se mais útil quando usado em cenários nos quais a função a estimar assume dependências lineares, visto que este encontra a linha reta mais bem ajustada.

2.3.2 Regressão Logística

A regressão logística (LR) é um método estatístico semelhante à regressão linear já que este método encontra uma função que prevê o valor de uma variável binária, Y, a partir de uma ou mais variáveis explicativas, X. Porém, ao contrário da regressão linear as variáveis explicativas podem ser categóricas ou contínuas, o modelo não requer dados de natureza estritamente contínua.

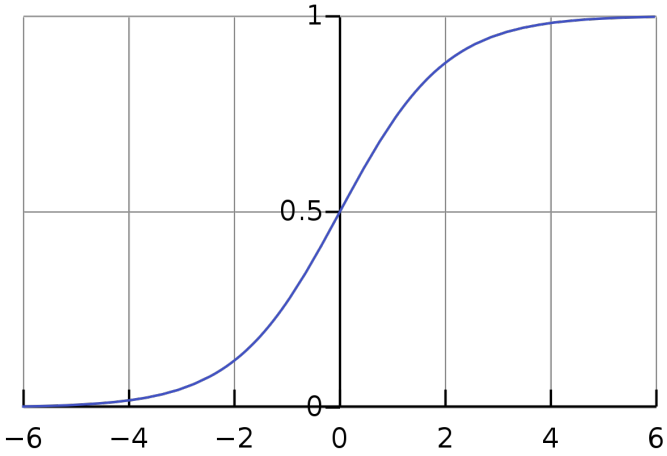

Figura 2.2: A função logística [1]

A regressão logística recebe o seu nome graças à função utilizada no núcleo deste método, a função logística. A função logística, também denominada de função sigmoide, foi desenvolvida com o intuito de descrever as propriedades do crescimento populacional na área da ecologia [43] e definida através da seguinte expressão, onde e é a base dos logaritmos naturais e t os valores numéricos de input que tencionamos transformar.

$$\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}} \tag{2.14}$$

Como podemos observar na figura 2.2, são funções em forma de S que podem receber qualquer valor $\in \mathbb{R}$ e mapeá-lo num valor entre 0 e 1, mas nunca exatamente nesses limites.

Assumindo que t é uma função linear de uma única variável explicativa x (quando t é uma combinação linear de mais do que uma variável explicativa o raciocínio seria o mesmo), podemos exprimir t da seguinte forma:

$$t = \beta_0 + \beta_1 x \tag{2.15}$$

onde β_0 é o *bias* e o β_1 o coeficiente para o único valor de input (x) que terá de ser aprendido a partir dos dados de treino. De acordo com este pressuposto, a função logística geral pode ser descrita como

$$p(x) = \sigma(t) = \frac{1}{1 + e^{-\beta_0 + \beta_1 x}} \tag{2.16}$$

No modelo logístico, $p(x)$ é interpretada como a probabilidade da classe padrão ($Y = 1$). Estamos a modelar a probabilidade do input (x) pertencer à classe padrão ($Y = 1$), que é dada por $P(x) = P(Y = 1|X)$.

Podemos agora definir a função 'logit' ou '*log odds*' como o inverso da função logística padrão, e aplicando-a podemos observar que

$$g(p(x)) = \sigma^{-1}(p(x)) = logit(px) = ln(\frac{p(x)}{1 - p(x)}) = \beta_0 + \beta_1 x \Leftrightarrow \frac{p(x)}{1 - p(x)} = e^{\beta_0 + \beta_1 x} \tag{2.17}$$

As *odds* da variável dependente igualar um caso (dada alguma combinação linear dos atributos) é equivalente à função exponencial da expressão da regressão linear. Isto ilustra como o logit serve de função de ligação entre a probabilidade e a expressão de regressão linear [43].

Para prever a pertença a uma classe, a regressão logística usa as '*log odds*' em vez das probabilidades e um método de *iterative maximum likelihood* em vez do *least squares* para dar *fit* ao modelo final. A regressão logística é mais apropriada para conjuntos de dados que não sigam distribuições normais ou quando as amostras têm matrizes de covariância diferentes. Este método assume a independência entre variáveis, o que nem sempre se verifica em conjuntos de dados reais. Mesmo assim a aplicabilidade do método e quão bem ele funciona normalmente excedem as suas expectativas estatísticas.

2.3.3 Árvores de Decisão

A aprendizagem de árvores de decisão ou indução de árvores de decisão é uma abordagem de modelação preditiva que utiliza uma árvore de decisão (como modelo preditivo) para ir desde observações sobre um item (representado nos ramos) até conclusões sobre o valor-alvo do item (representado nas folhas). Os modelos em árvores onde a variável-alvo pode tomar um conjunto discreto de valores são chamados árvores de classificação; nestas estruturas de árvores, as folhas representam *labels* de classes e os ramos representam conjunções das características que conduzem a essas *labels*. Árvores de decisão onde a variável alvo pode assumir valores contínuos (tipicamente números reais) são denominadas árvores de regressão. As árvores de decisão estão entre os algoritmos de *machine learning* mais populares, dada a sua inteligibilidade e simplicidade [89].

Por uma questão de simplicidade vamos assumir que tratamos de uma árvore de classificação com uma única variável-alvo categórica e cujos atributos de input possuem domínios finitos e discretos. Cada elemento no domínio da variável-alvo constitui uma classe. Uma árvore de decisão é uma árvore na qual a cada nó interno (não-folha) é atribuído um dos atributos de input. Aos arcos com origem num nó com um atributo de input são também a eles lhes atribuído cada um dos valores possíveis da variável-alvo ou o arco leva a um nó de decisão subordinado sobre um

atributo de input diferente. A cada nó folha da árvore é atribuída uma classe ou a distribuição de probabilidades sobre as classes, significando que o conjunto de dados foi classificado corretamente pela árvore numa classe ou numa distribuição de probabilidade em específico.

A árvore é construída através da divisão do conjunto original, que constitui a raiz da árvore, em subconjuntos que constituem os seus nós sucessores. Esta divisão é realizada de acordo com determinadas métricas baseadas nos atributos do conjunto de dados. Este processo é repetido em cada um dos subconjuntos criados de forma recursiva. A recursão diz-se completa quando um subconjunto presente num determinado nó possui todos os mesmos valores da variável-alvo ou quando uma nova divisão não acrescenta valor às previsões. Este processo é chamado de *top-down induction of decision trees* (TDIDT) [74] e constitui uma das estratégias mais comuns na aprendizagem de árvores de decisão. As métricas utilizadas para fazer a melhor divisão são diferentes dependendo do algoritmo em questão, sendo que normalmente medem a homogeneidade da variável-alvo dentro dos subconjuntos criados. Estas métricas são aplicadas a cada subconjunto candidato e os valores resultantes são combinados de modo a obter uma medida geral da qualidade da divisão.

2.3.3.1 Métodos de Combinação

Métodos de combinação (ou *ensemble methods* em inglês) utilizam múltiplos algoritmos de aprendizagem para obter melhores resultados preditivos do que seria possível utilizando apenas um único dos algoritmos que o constituem [65, 70, 77]. Empiricamente, métodos de combinação proporcionam melhores resultados quando existe uma diversidade significativa entre os modelos constituintes e como tal, vários métodos de combinação são desenvolvidos com a promoção da diversidade entre modelos em mente [47, 82]. Uma abordagem comum neste tipo de métodos é a utilização de árvores de decisão como os modelos base, sendo que existem diferentes formas de as combinar de modo a criar métodos de combinação com diferentes características.

Bagging (*Boostrap Aggregation*) é utilizado quando o objetivo passa por reduzir a variância de uma única árvore de decisão [23]. Dado um conjunto de dados de treino D de tamanho n, bagging gera m novos conjuntos de dados de treino D_i de tamanho n' por amostragem aleatória em D de forma uniforme e com substituição. Ao realizar esta amostragem com substituição algumas observações podem acabar por ser repetidas em cada D_i. Este tipo de amostragem é chamado de *bootstrap sampling*. O facto de este ser realizado com substituição garante que cada novo conjunto de dados é independente dos restantes, visto que cada conjunto não depende das observações previamente selecionadas. Após este processo, são criados m modelos utilizando os m conjuntos de dados criados e estes são posteriormente combinados através de uma média dos valores de output (no caso da regressão) ou através do voto por maioria (no caso da classificação).

Random Forest é um outro método de combinação que pode ser considerado como uma extensão do bagging. Para além de selecionar as observações do conjunto de dados D de forma aleatória para formar os conjuntos D_i, este método seleciona também um número aleatório de atributos, em contraste com um uso tradicional de bagging, que utiliza todas os atributos

presentes para treinar as árvores. Apesar de não ser algo intuitivo, algoritmos mais aleatórios podem ser utilizados de modo a produzir um modelo de combinação melhor em comparação a algoritmos mais deliberados [41].

Boosting envolve a construção incremental de um método de combinação através do treino de cada nova iteração do modelo base com ênfase nas instâncias com maior erro/mal classificadas pelos modelos anteriores. Assumindo um problema de classificação por questões de simplicidade (lógica semelhante para um problema regressão), são atribuídos pesos iguais as instâncias presentes nos dados de treino no início da sua execução. Estes dados são então fornecidos a um modelo base (neste caso uma árvore de decisão). Às instâncias mal classificadas pelo primeiro modelo base são atualizados os pesos de modo a que estes sejam maiores do que os pesos das instâncias corretamente classificadas. Por sua vez, este conjunto de dados com os pesos atualizados é fornecido a uma nova iteração do modelo base e assim por diante. Os resultados são então combinados sob a forma de voto por maioria.

O **Gradient Boosting** é uma extensão do método de boosting. Ele utiliza o algoritmo de *gradient descent* que é capaz de otimizar qualquer função de perda diferenciável [71]. Um conjunto de árvores é construído um a um e as árvores individuais são somadas sequencialmente. A árvore seguinte tenta recuperar a perda (diferença entre os valores reais e os valores previstos).

2.3.4 K-Nearest Neighbours (k-NN)

O k-nearest neighbors (k-NN) é um dos algoritmos de *machine learning* mais simples, fáceis de compreender, versáteis e de maior relevância. K-NN assume a similaridade entre novos casos/dados e os casos que já possui armazenados, e obtém uma previsão para um novo caso utilizando uma combinação dos valores dos seus k vizinhos mais próximos. O algoritmo é normalmente utilizado para classificação mas pode também ser utilizado em problemas de regressão. k-NN é um algoritmo não paramétrico, o que significa que não faz qualquer tipo de suposição sobre os dados subjacentes. É também caracterizado como um '*lazy learner*' já que não aprende com os dados de treino de forma imediata, apenas os armazena e na altura da classificação, executa uma ação sobre os dados armazenados.

O k-NN realiza um tipo de classificação onde a função é apenas aproximada localmente e toda a computação é diferida até o momento de avaliação. Visto que o algoritmo depende da noção de distância para a classificação, se os atributos representam diferentes unidades físicas ou possuem escalas muito diferentes, a normalização dos dados de treino pode melhorar significativamente a sua exatidão [40, 69].

Uma técnica útil tanto para classificação e regressão consiste na atribuição de pesos às contribuições dos vizinhos (observações mais próximas) de modo a que os vizinhos mais próximos contribuam mais para o resultado do que os vizinhos mais distantes. Um esquema comum para a atribuição de pesos é, por exemplo, a atribuição do peso $1/d$ a cada vizinho, sendo que d representa a distância correspondente a esse vizinho.

As métricas de distância é o hiperparâmetro através do qual medimos a distância entre os valores dos atributos dos dados armazenados e as novas entradas de teste.

Algumas das métricas de distâncias mais comuns incluem a distância euclidiana:

$$d(i,j) = \sqrt{|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + ... + |x_{ip} - x_{j1p}|^2} \tag{2.18}$$

Uma alternativa à distância euclidiana é a distância de Manhattan, onde as diferenças entre os valores dos atributos não são colocadas ao quadrado, apenas somadas (após determinar o valor absoluto):

$$d(i,j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + ... + |x_{ip} - x_{j1p}| \tag{2.19}$$

Outras métricas de distância podem ser obtidas assumindo potências mais altas do que 2. Potências mais altas aumentam a influência de maiores diferenças com o sacrifício das menores diferenças. Geralmente, a distância Euclidiana representa um bom compromisso apesar de outras métricas de distância poderem ser mais apropriadas em contextos específicos.

A escolha do número de vizinhos mais próximos a considerar k é também um hiperparâmetro crítico na criação de um modelo k-NN. Um valor demasiado baixo aumenta a influência do ruído no resultado e pode causar *overfitting* ao tornar as fronteiras entre classes menos distintas. Por outro lado, um valor excessivamente alto de k necessita de mais poder computacional e pode causar *underfitting* pois o algoritmo acaba por ignorar as estruturas locais e a noção de vizinhanças. Um bom valor de k pode ser selecionado através de determinadas técnicas de otimização de hiperparâmetros. Podemos, por exemplo, treinar vários modelos com valores de k diferentes e observar a evolução do erro de teste de acordo com o aumento do valor de k a partir de um determinado valor inicial. O melhor valor de k será corresponde ao valor que minimiza a curva do erro de teste.

2.3.5 Máquinas de Vetores de Suporte (SVM)

As máquinas de vetores de suporte, ou *support vector machines* em inglês, (SVM) são algoritmos relativamente simples utilizados para regressão e classificação (apesar de geralmente mais utilizadas em problemas de classificação). Fundamentalmente, a SVM encarrega-se de determinar o hiperplano que cria uma fronteira entre os tipos de dados.

Cada instância do conjunto de dados é visto como um vetor com N dimensões, onde N corresponde ao número de atributos dos dados. O objetivo do modelo é identificar se é possível separar os dados através de um subespaço plano de $(N-1)$ dimensões, um hiperplano. Podem existir vários hiperplanos que separam corretamente os dados de treino. Uma escolha razoável do melhor hiperplano é aquele apresente a maior separação/margem entre as diferentes classes. Sendo assim, é escolhido o plano do modo a que a distância entre ele e o ponto de cada classe mais

próximo seja maximizada, como se pode observar no exemplo da figura 2.3. H3 não separa as classes. Ambos H1 e H2 separam as classes mas, como H2 as separa com margem máxima, deve ser selecionado como o melhor hiperplano neste exemplo. As instâncias/pontos mais próximos do hiperplano são chamados de vetores de suporte, e é a eles que o método deve o seu nome.

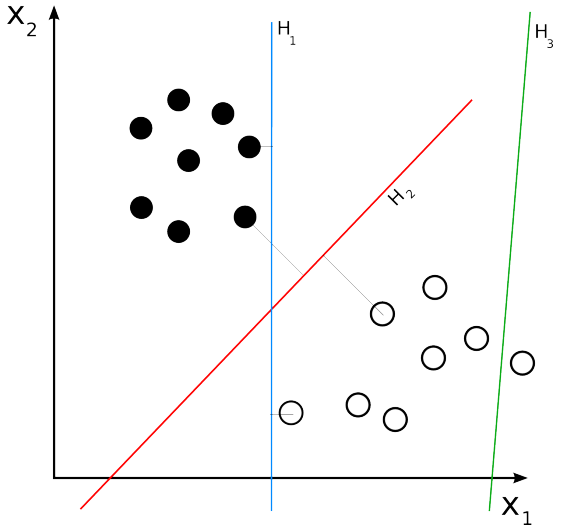

Figura 2.3: Conjunto de dados de exemplo e os seus possíveis hiperplanos de separação [2]

No caso de classificação *mutliclass*, o modelo cria um classificador binário para cada classe da variável-alvo (com output pertence/não pertence à classe especificada). O classificador com os melhores resultados é escolhido como output do SVM para a instância em questão.

Segundo a definição anterior, é fácil de perceber que as máquinas de vetores de suporte têm bom desempenho para dados linearmente separáveis, sem recorrer a nenhum tipo de manipulação/modificação destes dados. No caso de dados mais complexos, as SVM utilizam o seu núcleo, ou *kernel* em inglês, para obter desempenhos semelhantes.

Um núcleo/*kernel* permite a este modelo projetar os atributos num espaço estendido (de maiores dimensões) onde os dados são linearmente separáveis. No entanto, a SVM não necessita de realmente efetuar esta transformação dispendiosa para um espaço de maiores dimensões para determinar o hiperplano de separação ótimo. Isto é conhecido como o truque do núcleo, ou *kernel trick* em inglês. Internamente, o SVM com núcleo consegue calcular estas transformações complexas em termos de cálculos de similaridade entre pares de pontos no espaço de atributos projetado utilizando coordenadas do espaço original.

Existem várias funções de núcleo diferentes possíveis de utilizar em SVMs. Três das mais utilizadas incluem:

$$\text{O núcleo linear: } K(x, y) = x \cdot y \tag{2.20}$$

$$\text{O núcleo polinomial: } K(x, y) = (1 + x \cdot y)^{degree} \tag{2.21}$$

$$\text{O núcleo RBF: } K(x, y) = exp(-\gamma||x - y||^2) \tag{2.22}$$

A escolha da função de núcleo a utilizar depende de problema para problema, consoante o conjunto de dados que este apresenta. Por exemplo, quando o conjunto de dados é linearmente separável, a melhor decisão seria o núcleo linear. Neste caso, utilizar um núcleo não-linear como o polinomial ou o RBF seria mais dispendioso, mais demorado (devido há existência de mais hiperparâmetros para otimizar) e também mais suscetível a *overfitting*.

2.3.6 Redes Neuronais Artificiais (ANNs)

Redes neuronais artificiais são modelos inspirados pelas redes neuronais biológicas que constituem o cérebro humano. Uma rede neuronal é essencialmente uma coleção de unidades ou nós chamados de neurónios artificiais que tentam modelar os neurónios biológicos presentes num cérebro. Cada conexão é capaz de transmitir um sinal a outros neurónios, funcionando de forma semelhante a uma sinapse num cérebro biológico. Um neurónio artificial recebe um sinal, processa-o e é também capaz de transmitir um sinal aos seus neurónios adjacentes. O output de cada neurónio é calculado através da aplicação de uma determinada função de ativação à soma de todos os seus dados de input. As conexões possuem pesos que são ajustados durante o processo de aprendizagem. O peso aumenta ou diminui a intensidade do sinal proveniente dessa conexão. Os neurónios podem possuir um determinado limite que garante que um sinal é apenas transmitido se o agregado resultante dos sinais de input o ultrapassar.

Tipicamente, os neurónios são organizados em camadas. Camadas diferentes podem aplicar diferentes transformações aos seus inputs. As redes neuronais aprendem ao processar exemplos, cada um deles possuindo os valores dos atributos e o valor da variável-alvo. As redes formam associações com determinados pesos entre estes dois valores que são armazenadas na própria estrutura da rede.

A fase de treino de uma rede neuronal a partir de um conjunto de exemplos está associada ao cálculo da diferença entre os valores de output processados (previsões) e os valores de output reais (valores da variável-alvo), ou seja, a obtenção do erro do modelo. Após isso, o algoritmo de retropropagação é utilizado no ajuste dos pesos das conexões para compensar cada erro encontrado durante a aprendizagem. A quantidade de erro é dividida entre as conexões. Tecnicamente, a retropropagação calcula o gradiente (a derivada) da função de perda (função que mede o erro) associada a um determinado estado no que diz respeito aos pesos. As atualizações são frequentemente feitas através da descida gradiente estocástica (método iterativo utilizado para otimizar uma função de perda) [78]. O algoritmo recebe o seu nome devido aos pesos, que são atualizados de trás para a frente, do output para o input. Ajustes sucessivos dos pesos fazem com que a rede produza outputs cada vez mais semelhantes ao valor real da variável-alvo. Após um número suficiente de iterações, a fase de treino pode ser terminada com base num determinado critério, definido para garantir o melhor desempenho da rede.

A perceptron multicamadas, ou *multilayer perceptron* em inglês (MLP) é um tipo de redes neuronais artificiais *feedforward* capaz de aproximar qualquer função não linear. Uma rede *feedforward* é uma rede onde as conexões entre os nós não formam um ciclo, ou seja, existe

apenas uma direção de propagação dos sinais. Uma MLP consiste em pelo menos 3 camadas: uma camada de input, uma camada oculta e uma camada de output. Entre a camada de input e a camada de output, uma MLP pode possuir um número arbitrário de camadas ocultas. Uma outra característica das MLPs é que são completamente conectadas, ou seja, cada nó numa camada conecta com um determinado peso w_{ij} a cada nó da camada seguinte.

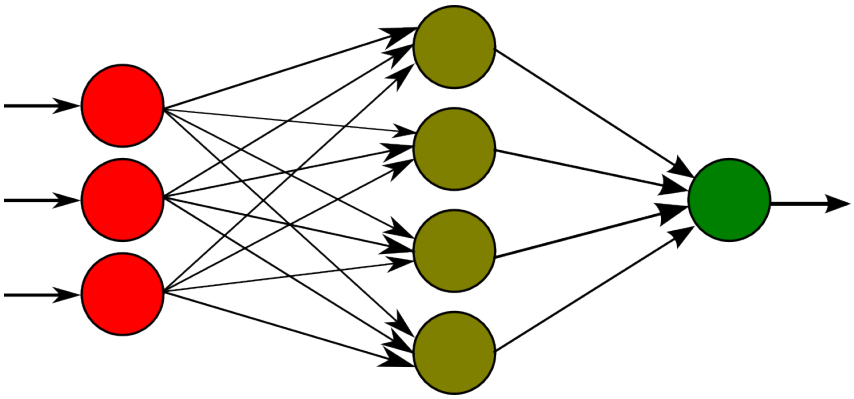

Figura 2.4: Exemplo de uma MLP [3]

Exceto os nós da camada de input, cada nó de uma MLP utiliza funções de ativação não lineares. Cada função recebe um único valor e efetua uma certa operação matemática sobre ele. Esta são algumas das funções de ativação mais comuns para MLPs:

A função **sigmoide** recebe um valor real e transforma-o num valor entre 0 e 1 (como já observamos anteriormente na secção 2.3.2). Esta função de ativação é geralmente utilizada quando temos de prever a probabilidade como output em cenários de regressão logística para determinar a probabilidade de ocorrência de classes. O principal problema ao utilizar a função sigmoide como função de ativação é que esta função é afetada pelo problema da dissipação do gradiente, pelo que devem ser utilizadas apenas para redes compostas por um baixo número de camadas.

O problema de dissipação do gradiente é encontrado no treino de redes neuronais artificiais com métodos de aprendizagem baseados em gradientes e retropropagação. Nestes métodos, cada um dos pesos da rede neuronal recebe uma atualização proporcional à derivada parcial da função de erro em relação ao peso atual em cada iteração de treino. O problema é que, nalguns casos, o gradiente será cada vez mais pequeno, impedindo efetivamente que o peso altere o seu valor. No pior dos casos, isto pode impedir completamente a rede neural de continuar o treino.

A **tangente hiperbólica (tanh)** recebe um valor real e transforma-o num valor entre -1 e 1.

$$tanh(x) = 2\sigma(2x) - 1 \tag{2.23}$$

A tangente hiperbólica é graficamente semelhante à função sigmoide mas centrada entre -1 e 1. A maior vantagem desta característica em relação à sigmoide é que os valores negativos

de input são mapeados fortemente negativos e os valores de input zero são mapeados perto de zero. Esta função é maioritariamente utilizada em cenários em que se pretende efetuar uma classificação entre duas classes.

Assim como a sigmoide, a tangente hiperbólica também é afetada pelo problema da dissipação do gradiente, é por isso também necessário ter em conta o número de camadas da rede na sua utilização.

A **Unidades Lineares Retificadas (ReLu)** recebe um valor real e limita-o a 0 (substitui valores negativos por 0).

$$ReLu(x) = max(0, x) \tag{2.24}$$

A função ReLU retifica o problema da dissipação do gradiente, algo que a tornou na função de ativação mais popular nos dias de hoje [78]. O gradiente para a ReLU é dado por:

$$fRelu(x) = \begin{cases} 0 & \text{if } x \leq 0 \\ 1 & \text{if } x > 0 \end{cases} \tag{2.25}$$

Como a derivada da ReLU apenas pode assumir os valores 0 ou 1, multiplicar pela própria não faz com que os pesos que estão mais longe da camada de output fiquem gradualmente mais pequenos e é graças a isto que o problema da dissipação do gradiente é drasticamente reduzido [37]. Esta característica também permite obter representações esparsas da rede neuronal com um número reduzido de conexões úteis, visto que a ReLU retorna 0 para todos os valores de input negativos, o que torna provável que determinados neurónios não sejam sequer ativados (tenham um output diferente de 0).

Esta representação esparsa é benéfica para uma rede neuronal porque resulta em modelos mais concisos, com menos *overfit* e melhor exatidão na previsão dos valores de output. Numa rede esparsa, os neurónios são garantidos de processar informação relevante relativa ao problema a resolver. A função ReLU é também mais eficiente de calcular, tendo apenas de escolher o máximo entre 0 e o valor de input, e não necessitando de executar operações dispendiosas como por exemplo a exponenciação, no caso da função sigmoide.

2.4 Aprendizagem Não-Supervisionada

A aprendizagem não supervisionada engloba um tipo de algoritmos que focam na aprendizagem de padrões através de dados não catalogados. Em contraste com a aprendizagem supervisionada onde os dados são catalogados por humanos, a aprendizagem não supervisionada é utilizada na descoberta das estruturas subjacentes num conjunto de dados, no agrupamento de dados de

acordo com similaridades e na representação do dataset de uma forma reduzida, tudo isto sem necessidade da intervenção humana. A sua capacidade em identificar semelhanças e diferenças nos dados tornam-na na solução ideal para tarefas de processamento mais complexas. Este tipo de aprendizagem pode ser mais fácil, mais rápida e menos dispendiosa visto que não requer o trabalho manual associado à catalogação dos dados necessária para aplicar aprendizagem supervisionada. Por outro lado, possui também algumas desvantagens como a dificuldade em verificar a exatidão dos outputs de aprendizagem não supervisionada, já que não existem dados catalogados para comparar aos resultados, ou a necessidade de empregar mais tempo dos profissionais envolvidos (engenheiros, cientistas de dados, etc) na interpretação dos resultados.

Os métodos de aprendizagem não supervisionada utilizados nesta dissertação são *clustering* e modelos de variáveis latentes. Como o nosso problema incide no domínio do processamento de linguagens naturais e as tarefas de aprendizagem não supervisionada presentes na dissertação estão inseridas num contexto de NLP, nas secções seguintes apenas são abordados os conceitos gerais dos métodos mencionados, sendo que na secção 2.5 é abordado em mais detalhe as especificidades dos algoritmos utilizados.

2.4.1 Clustering

A análise de *clusters* ou *clustering* é a tarefa de agrupar um conjunto de objetos de tal forma que os objetos do mesmo grupo (chamado *cluster*) são mais semelhantes (num determinado sentido) uns aos outros do que a aqueles pertencentes a outros grupos (*clusters*).

A análise de *clusters* em si não é um algoritmo específico, mas uma tarefa geral a ser resolvida. Pode ser alcançada através de vários algoritmos que diferem significativamente na sua compreensão do que constitui um *cluster* e de como os encontrar eficientemente. Noções populares de *clusters* incluem grupos com pequenas distâncias entre os membros do *cluster*, áreas densas do espaço de dados, intervalos ou distribuições estatísticas particulares. *Clustering* pode, portanto, ser formulado como um problema de otimização multi-objetivo. O algoritmo de *clustering* apropriado e definições de parâmetros (incluindo parâmetros tais como a função de distância a utilizar, um limiar de densidade ou o número de agrupamentos esperados) dependem do conjunto de dados individual e da utilização pretendida dos resultados. A análise de *clusters*, como tal, não é uma tarefa automática, mas um processo iterativo de descoberta de conhecimento ou otimização interativa multi-objetivo que envolve tentativa e falha. É frequentemente necessário modificar o pré-processamento de dados e os parâmetros do modelo até que o resultado atinja as propriedades desejadas.

2.4.2 Modelos de Variáveis Latentes

Em estatística, as variáveis latentes, em oposição às variáveis observáveis, são variáveis que não são diretamente observadas mas sim inferidas (através de um modelo matemático) a partir de outras variáveis que são observadas (diretamente medidas). Os modelos matemáticos que

visam explicar as variáveis observadas em termos de variáveis latentes são chamados modelos de variáveis latentes

Um modelo de variável latente é um modelo estatístico que relaciona um conjunto de variáveis observáveis (as chamadas variáveis manifestas) com um conjunto de variáveis latentes. Assume-se que as respostas sobre os indicadores ou variáveis manifestas são o resultado da posição de um indivíduo sobre a(s) variável(s) latente(s), e que as variáveis manifestas não têm nada em comum depois de controladas para a variável latente (independência local).

Um exemplo de modelo de variável latente utilizado nesta dissertação é o *clustering* baseado na distribuição, um modelo de variável latente para *clustering*. O *clustering* baseado na distribuição é modelo de *clustering* mais estreitamente relacionado com a estatística visto que os *clusters* são modelados de acordo com distribuições estatísticas. Ao contrário dos 'hard' *clusters*, onde cada objeto apenas pertence a um *cluster*, este modelo possui 'soft' *clusters* que permitem que cada objeto pertença a cada *cluster* com um determinado grau (por exemplo, a probabilidade de pertencer ao *cluster* em questão). Os *clusters* podem então ser facilmente definidos como objetos pertencentes com maior probabilidade à mesma distribuição. O cálculo destas probabilidades de pertença pode ser obtido de diferentes formas dependendo do algoritmo em questão.

2.5 Processamento de Linguagem Natural (NLP)

O processamento de linguagem natural é um subcampo da linguística, informática e inteligência artificial relacionado com as interações entre computadores e a linguagem humana, em particular com a forma de programar computadores para processarem e analisarem grandes quantidades de dados de linguagem natural. O objetivo deste campo é prover a um computador as ferramentas necessárias de modo a que este se torne capaz de 'compreender' o conteúdo de um conjunto de documentos, compreender as nuances contextuais da linguagem neles presente. Esta tecnologia trata então de extrair a informação e conhecimento contidos numa coleção de documentos, sendo também capaz de categorizar e organizar os próprios, tudo isto de uma forma mais rápida, mais precisa e consistente do que através de agentes humanos.

No processamento de linguagem natural, a linguagem humana é separada em fragmentos de forma a que a estrutura gramatical das frases e o significado das palavras possam ser analisados e compreendidos de acordo com o contexto. Esta fase normalmente é chamada de pré-processamento, e a qualidade da mesma pode influenciar os resultados de qualquer modelo ao qual estes dados sejam fornecidos.

Aqui estão algumas tarefas fundamentais de pré-processamento que utilizamos no nosso desenvolvimento e que são normalmente necessárias de executar de modo a que as ferramentas de NLP possam fazer sentido da linguagem humana:

- *Tokenization*: Processo de divisão do texto em unidades semânticas mais pequenas.

- *Part-of-speech (POS) tagging*: Processo através do qual cada palavra é atribuída uma *label* de *part-of-speech*. Esta *label* pode ser substantivo (NN), advérbio (RB), verbo (VB), ou uma parte mais especializada do discurso como substantivo plural próprio (NNPS), advérbio superlativo (RBS), verbo na 3ª pessoa (VBZ), etc. O objetivo da *POS tagging* no pré-processamento é excluir algumas partes de discurso que não contenham qualquer utilidade para a aplicação em questão.

- *Stemming*: Processo de remover as terminações das palavras a fim de detetar a sua forma de raiz. Fazendo isso, muitas palavras são agrupadas numa só e a dimensionalidade é reduzida.

- *Lemmatization*: Uma alternativa na redução de dimensionalidade. No processo de *lemmatization*, muitas palavras são também agrupadas numa só. Este processo analisa uma palavra morfologicamente e remove a sua terminação, produzindo assim a sua forma base ou *lemma*. Ao contrário do *stemming*, a *lemmatization* gera termos existentes na linguagem.

- *Stop word removal*: Processo de remoção de *stop words*. *Stop words* são palavras com altas frequências de presença em todas as frases mas consideradas desnecessárias devido ao facto de não conterem informação útil para posterior análise. O conjunto destas palavras não está completamente predefinido e pode ser alterado através da remoção ou adição de palavras a ele, de acordo com as necessidades da aplicação.

- *Lowercasing*: Processo de conversão de todas as palavras existentes no texto para letra minúscula com o intuito de reduzir dimensionalidade (as mesmas palavras em letra maiúscula e minúscula passam a ser agrupadas numa só).

Depois da fase de pré-processamento, é necessário construir um algoritmo de NLP e treiná-lo de modo a que ele consiga interpretar linguagem natural e executar tarefas específicas. Existem 2 tipos principais de algoritmos utilizados na resolução de problemas de NLP:

- Algoritmo simbólico (baseado em regras): Os algoritmos baseados em regras dependem de regras gramaticais manualmente criadas por peritos idiomáticos com conhecimento sobre o domínio da linguística assim como da consulta de um dicionário. Este tipo de algoritmos foi um dos primeiros algoritmos de NLP concretizados.

- Algoritmo de *machine learning*: Os algoritmos de *machine learning* possuem algumas vantagens em comparação com os seus predecessores. Os procedimentos de aprendizagem automática podem fazer uso de algoritmos de inferência estatística para produzir modelos que são robustos a dados de input não familiares e a dados errados. Geralmente, o tratamento destes dados com regras manuscritas é extremamente difícil, sujeito a erros e demorado. Sistemas baseados na aprendizagem automática das regras podem também ser tornados mais exatos simplesmente através do fornecimento de mais dados de input. Contudo, os sistemas baseados em regras manuscritas só podem ser tornados mais exatos aumentando a complexidade das regras, o que é uma tarefa muito mais difícil.

Apesar da popularidade dos algoritmos de *machine learning*, os algoritmos simbólicos ainda são regularmente utilizados atualmente em diversas situações. Embora as tarefas de NLP serem estreitamente interligadas, elas podem ser subdivididas de acordo com a sua utilização e objetivo, sendo que as tarefas apresentadas de seguida constituem o conjunto de tarefas de NLP utilizadas no contexto da dissertação.

2.5.1 Classificação de Texto

A classificação de texto é uma técnica que atribui um conjunto de categorias pré-definidas a texto em aberto. Os classificadores de texto podem ser utilizados para organizar, estruturar e categorizar praticamente qualquer tipo de texto - desde documentos, ficheiros, publicações nas redes sociais, etc.

Estima-se que cerca de 80% de toda a informação é não-estruturada [87], sendo o texto um dos tipos mais comuns de dados não-estruturados. Devido à natureza de dados em formato de texto, analisar, compreender, organizar e ordená-los é uma tarefa possivelmente difícil e demorada. É aqui que entra a classificação de texto com *machine learning*. A classificação manual de texto envolve um anotador humano, que interpreta o conteúdo do texto e o classifica em conformidade. Este método pode dar bons resultados, mas é demorado e dispendioso. A classificação automática de texto utiliza *machine learning*, o processamento de linguagem natural e outras técnicas guiadas por IA. Utilizando estes classificadores de texto é possível estruturar automaticamente todo o tipo de texto relevante, desde e-mails, documentos legais, *chatbots*, inquéritos, e muito mais de uma forma mais rápida, mais económica e mais precisa.

A aplicação da classificação de texto da qual nos servimos neste projeto foi a análise de sentimento. A análise de sentimento é uma ferramenta de *machine learning* que analisa dados de texto de modo a determinar o tom emocional subjacente. O sentimento é classificado como positivo, neutro ou negativo é obtido através de uma métrica chamada polaridade, que pode assumir valores entre [-1,1]. De uma forma muito simplificada, a análise do sentimento ajuda determinar a atitude do autor do texto a ser analisado para com o tema abordado no mesmo.

Existem várias abordagens à classificação automática de textos, mas todas elas se enquadram em três tipos de sistemas: sistemas baseados em regras, sistemas baseados em *machine learning* e sistemas híbridos. As abordagens baseadas em regras classificam o texto em grupos organizados utilizando um conjunto de regras linguísticas criadas manualmente. Estas regras instruem o sistema a utilizar elementos semanticamente relevantes de um texto para identificar categorias relevantes com base no seu conteúdo. Cada regra consiste de um antecedente ou padrão e uma categoria prevista.

Em vez de confiar em regras criadas manualmente, a classificação de textos de *machine learning* aprende a fazer classificações com base em observações passadas. Utilizando exemplos pré-catalogados como dados de treino, os algoritmos de aprendizagem automática podem aprender as diferentes associações entre secções de texto, e que um determinado resultado (um rótulo ou

label) é esperado para um determinado input (texto).

Os sistemas híbridos utilizam as duas abordagens anteriores e combinam um classificador base de *machine learning* com um sistema baseado em regras.

Na nossa implementação, utilizamos uma abordagem baseada em regras através da biblioteca TextBlob[1], uma biblioteca de processamento de linguagem natural para a linguagem Python que suporta operações complexas sobre dados textuais. A sua abordagem pode também ser caracterizada como uma abordagem baseada no léxico, um sentimento é definido pela sua orientação semântica e pela intensidade de cada palavra presente na frase. Isto requer que a biblioteca possua um dicionário pré-definido que classifique as palavras como negativas e positivas. Geralmente, uma mensagem de texto será representada por um *bag of words* (multiconjunto das suas palavras, ignorando a gramática e a ordem das mesmas). Ao calcular um sentimento para uma única palavra é utilizada a média aplicada sobre valores de polaridade na atribuição de uma pontuação de polaridade à palavra em questão. Isto é necessário pois podem existir no dicionário lexical mais do que uma entrada para determinadas palavras. Por exemplo, a palavra 'great' possui 4 entradas diferentes no dicionário da biblioteca TextBlob, cada uma delas com um sentido diferente: 'very good', 'of major significance or importance', 'relatively large in size or number or extent' e 'remarkable or out of the ordinary in degree or magnitude or effect'. Quando tentamos obter a polaridade da palavra 'great' individualmente, o modelo retorna uma média das polaridades associadas às 4 entradas de dicionário previamente mencionadas. Depois de atribuir pontuações individuais a todas as palavras, o sentimento final de um documento é novamente calculado por alguma operação de agregação, como obter a média de polaridade de todas as palavras analisadas.

Associados a cada entrada do dicionário da biblioteca TextBlob temos os valores de polaridade (como já mencionamos anteriormente) mas também outros valores que nos dão mais informação sobre a palavra e permitem algumas operações internas interessantes. Uma dessas operações/aspetos de destaque desta biblioteca é a sua capacidade de lidar com modificadores, também conhecidos como intensificadores porque intensificam o significado do texto de acordo com o seu padrão. Quando um intensificador é utilizado, a biblioteca ignora a sua polaridade e utiliza apenas um novo campo, a intensidade, de modo a calcular o sentimento do texto em questão. Utilizando novamente o exemplo 'great', se requisitarmos o cálculo do sentimento do texto 'very great', o resultado obtido passa pela multiplicação da polaridade da palavra 'great' pela intensidade da palavra 'very'. A biblioteca TextBlob possui também a capacidade de lidar com negações. Utilizando o texto de exemplo 'not great', a polaridade retornada consiste na polaridade da palavra 'great' multiplicada por -0.5. Um último facto interessante sobre esta biblioteca é que consegue também suportar ambas estas variantes em simultâneo, uma negação e um modificador. Neste caso, para além de multiplicar a polaridade da palavra por -0.5, o inverso da intensidade do modificador é também equacionado na multiplicação. Ou seja, no caso de 'not very great', a polaridade resultante consiste na polaridade da palavra 'great' multiplicada pelo valor constante de -0.5 assim como pelo valor inverso da intensidade de 'very'.

[1] https://textblob.readthedocs.io/en/dev/

No contexto da nossa dissertação, a análise de sentimento revela-se importante pois pode permitir-nos identificar algum tipo de relação entre o sentimento da publicação e a área editorial na qual se insere, tornando possível a caracterização da opinião das equipas de comunicação em relação às áreas editoriais nas quais publicam, mas também, e de uma forma muito mais simples, identificar uma possível relação entre o sentimento presente na publicação o envolvimento demonstrado pelo seu público, refletido no número de favoritos e de retweets da publicação em questão.

2.5.2 Word Embeddings

No processamento de linguagem natural, *word embeddings* ou incorporação de palavras é um termo utilizado para a representação de palavras para análise de texto, normalmente sob a forma de um vetor de valores reais que codifica o significado da palavra de modo a que as palavras que estão mais próximas num espaço vetorial pré-definido sejam expectáveis de possuir um significado semelhante [44].

A chave para esta abordagem é a ideia de utilizar uma representação densa e distribuída para cada palavra. Cada palavra é representada por um vetor de valores reais, muitas vezes com dezenas ou centenas de dimensões. Isto é contrastado com os milhares ou milhões de dimensões necessárias para representações esparsas de palavras, tais como *one-hot enconding*.

A representação distribuída é aprendida com base na utilização das palavras. Isto permite que palavras que são utilizadas de forma semelhante resultem em representações semelhantes, capturando naturalmente o seu significado. Isto pode ser contrastado com a representação nítida mas frágil num modelo de *bag-of-words* onde, a menos que explicitamente gerido, palavras diferentes têm representações diferentes, independentemente da forma como são utilizadas.

Os métodos de *embeddings* de palavras produzem vetores de palavras a partir de um corpus de texto. Primeiramente, é construído um vocabulário a partir dos dados de texto e de seguida é aprendida a representação vetorial das palavras presentes no vocabulário. O ficheiro vetorial de palavras resultante pode ser utilizado como características/atributos em muitas aplicações de nlp e *machine learning*.

Esta secção revê duas técnicas que podem ser utilizadas para aprender *embeddings* de palavras a partir de dados de texto e que consideramos aplicáveis ao nosso projeto:

2.5.2.1 Word2Vec

Word2Vec é um método estatístico para a aprendizagem de *embeddings* de palavras a partir de um corpo de texto. Foi desenvolvido por Tomas Mikolov, et al. na Google em 2013 [51] com o intuito de tornar o treino de *embeddings* baseado em redes neuronais mais eficiente e, desde então, tornou-se uma tecnologia padrão no desenvolvimento deste tipo de modelos.

Para além disso, estudos sobre análise dos vetores aprendidos e a exploração da matemática vetorial aplicada às representações das palavras, como [53] e [52], mostram que os vetores de palavras conseguem captar regularidades semânticas. Por exemplo, operações como

$$vetor('Paris') - vetor('France') + vetor('Italy')$$

resultam num vetor bastante semelhante ao $vetor('Rome')$, assim como a operação

$$vetor('king') - vetor('man') + vetor('woman')$$

resulta num vetor próximo do $vetor('queen')$.

Para obter as representações vetoriais apresentadas, foram introduzidos dois modelos de aprendizagem diferentes que podem ser utilizados como parte da abordagem do word2vec, sendo eles o *Bag-of-Words* Contínuo, ou modelo CBOW, e o modelo de Skip-Gram contínuo.

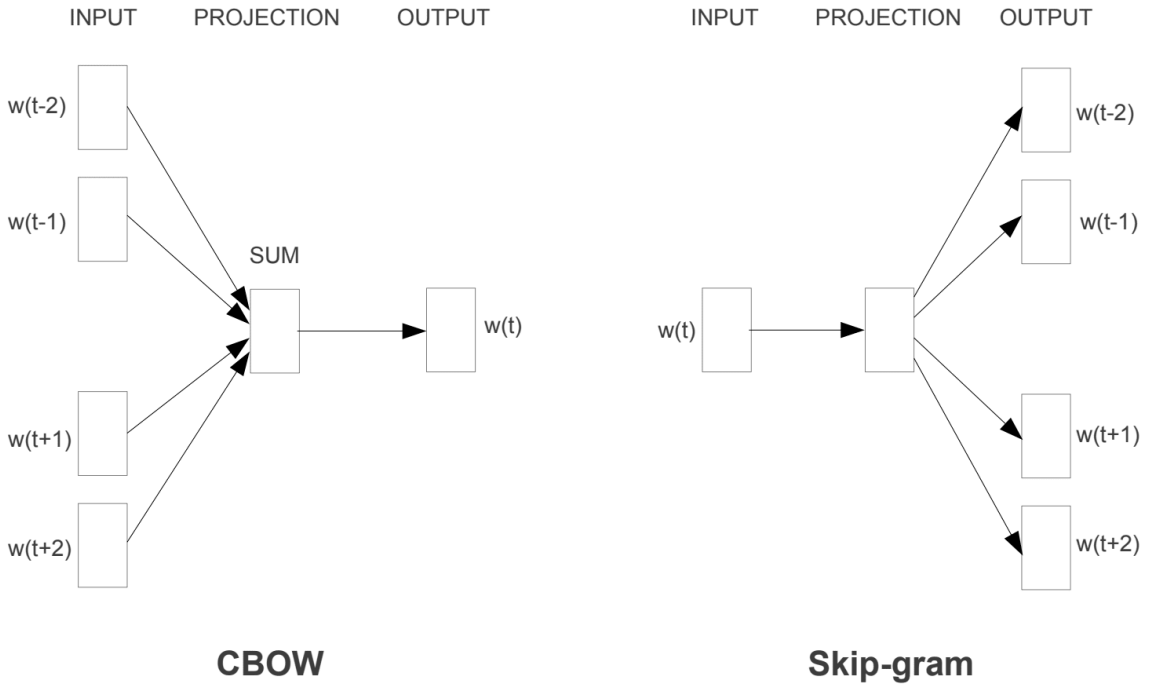

Figura 2.5: Modelos de aprendizagem do Word2Vec [51]

Na arquitetura do CBOW, o modelo prevê a palavra atual a partir de uma janela de palavras de contexto semelhante. Na arquitetura de skip-gram contínuo, o modelo utiliza a palavra atual para prever a janela de palavras de contexto mais próximas. A arquitetura do skip-gram atribui pesos maiores às palavras de contexto mais próximo do que às palavras mais distantes [51, 52].

Ambos os modelos estão centrados na aprendizagem das palavras dado o seu contexto de utilização local, onde o contexto é definido por uma janela de palavras vizinhas. Esta janela

é um parâmetro configurável do modelo. O tamanho da janela tem um forte efeito sobre as semelhanças vetoriais resultantes. Janelas grandes tendem a produzir mais semelhanças tópicas, enquanto janelas mais pequenas tendem a produzir mais semelhanças funcionais e sintáticas.

A principal vantagem deste método é que podem ser aprendidas *embeddings* de palavras de alta qualidade de forma eficiente (baixa complexidade de espaço e tempo), permitindo que *embeddings* maiores sejam aprendidas (mais dimensões) a partir de um corpus de texto muito maior (milhares de milhões de palavras).

2.5.2.2 GloVe

O algoritmo Global Vectors for Word Representation, ou GloVe, é uma extensão do método word2vec para a aprendizagem eficiente de vetores de palavras, desenvolvido por Pennington, et al. em 2014 [67]. Os modelos clássicos de representação do espaço vetorial de palavras foram desenvolvidos utilizando técnicas de fatorização matricial que fazem um bom trabalho de utilização de estatísticas de texto globais, mas não são tão bons como os métodos de aprendizagem, como o word2vec, na captura de significado e na sua demonstração em tarefas como o cálculo de analogias (por exemplo, as operações aplicadas na secção anterior aos vetores de 'king' e 'queen').

GloVe é uma abordagem que combina as estatísticas globais de técnicas de fatorização matricial com a aprendizagem local baseada no contexto como no word2vec. Em vez de usar uma janela para definir o contexto local, GloVe constrói um contexto explícito de palavras ou matriz de coocorrência de palavras utilizando estatísticas em todo o corpus de texto. O resultado é um modelo de aprendizagem que pode resultar em *embeddings* de palavras geralmente melhores.

A distância euclidiana (ou semelhança cosseno) entre dois vetores de palavras fornece um método eficaz para medir a semelhança linguística ou semântica das palavras correspondentes. As métricas de similaridade utilizadas para as avaliações dos vizinhos mais próximos produzem um único escalar que quantifica a afinidade de duas palavras. Esta simplicidade pode ser problemática uma vez que duas palavras dadas quase sempre exibem relações mais complexas do que as que podem ser capturadas por um único valor. Por exemplo, a palavra 'homem' pode ser considerada como semelhante à palavra 'mulher', na medida em que ambas as palavras descrevem seres humanos. Por outro lado, as duas palavras são frequentemente consideradas opostas, uma vez que destacam um eixo primário ao longo do qual os seres humanos diferem uns dos outros. O conceito subjacente que distingue 'homem' de 'mulher', isto é, sexo ou género, pode ser equivalentemente especificado por vários outros pares de palavras, tais como 'rei' e 'rainha' ou 'irmão' e 'irmã'. Para afirmar esta observação matematicamente, podemos esperar que as operações: homem - mulher, rei - rainha, e irmão - irmã possuam todas resultados aproximadamente iguais.

A fim de capturar de uma forma quantitativa a nuance necessária para distinguir 'homem' de 'mulher', é necessário que um modelo associe mais do que um único número ao par de palavras. Um candidato natural e simples para um conjunto alargado de números discriminatórios é a diferença vetorial entre os dois vetores de palavras. GloVe é concebido de modo a que tais

diferenças vetoriais captem ao máximo possível o significado especificado pela justaposição de duas palavras.

O modelo GloVe é treinado nas entradas não nulas de uma matriz global de coocorrência de palavras, que tabula a frequência com que as palavras coocorrem umas com as outras num determinado corpus. O preenchimento desta matriz requer uma única passagem por todo o corpus para recolher as estatísticas. Para grandes corpora, esta passagem pode ser computacionalmente dispendiosa, mas é um custo inicial único. As iterações de treino subsequentes são muito mais rápidas pois o número de entradas de matriz diferentes de zero é tipicamente muito menor do que o número total de palavras no corpus.

O objetivo do treino do modelo é aprender vetores de palavras de modo a que o seu produto escalar seja igual ao logaritmo da probabilidade de concorrência das palavras. Sabendo que o logaritmo de um rácio é igual à diferença de logaritmos, este objetivo associa rácios de probabilidades de concorrência com diferenças vetoriais. Como estes rácios podem possuir alguma forma de significado, esta informação também é codificada como diferenças vetoriais.

2.5.3 Identificação de Tópicos

Um dos objetivos mais críticos da análise de dados é determinar as características que estes partilham. Na análise de texto, isto significa muitas vezes determinar os eventos ou conceitos discutidos num documento. Esta informação é clara para um leitor humano, mas a um programa apenas é fornecido o texto tal como está escrito e não o assunto de cada documento. A fim de realizar esta tarefa através de um programa, os cientistas de dados utilizam um método chamado identificação de tópicos. A identificação de tópicos é uma ferramenta estatística popular utilizada na extração de variáveis latentes de grandes conjuntos de dados [21]. É particularmente bem adaptado para a utilização com dados de input no formato de texto, contudo, também tem sido utilizado para análise de dados bioinformáticos [49], dados de redes sociais [42] e dados ambientais [36]. É importante notar que os modelos de tópicos não substituem a interpretação humana de um texto. Em vez disso, são uma forma de fazer suposições educadas sobre como as palavras coexistem em diferentes temas latentes, identificando padrões na forma como coocorrem dentro dos documentos.

Por uma questão de simplicidade, iremos discutir a identificação de tópicos no contexto no qual a utilizamos, com dados de input em formato de texto. Neste contexto mais restrito, podemos definir a identificação de tópicos como uma técnica de *machine learning* não supervisionada capaz de analisar um conjunto de documentos, detetando padrões de palavras e frases dentro deles, e agrupar automaticamente grupos de palavras e expressões similares que melhor caracterizam um conjunto de documentos.

No seguimento deste documento, uma 'palavra' ou 'termo' representa a unidade fundamental de dados individuais, um 'documento' representa uma string composta por N palavras, um 'corpus' representa um conjunto constituído por M documentos geralmente abrangendo todo o

conjunto de dados e um 'vocabulário' é definido como uma coleção de todas as palavras distintas presentes num corpus.

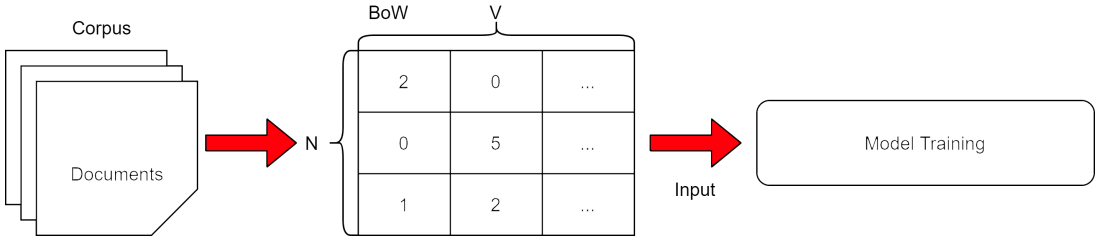

Figura 2.6: Diagrama dos passos de Topic Modeling

Para compreender melhor o funcionamento de um modelo de tópicos descrevemos primeiro as ideias básicas por detrás da identificação de tópicos através de um diagrama. A figura 2.6 ilustra os passos chave da identificação de tópicos. Primeiro, assumimos que existem N documentos, V palavras e K tópicos num corpus. De seguida, descrevemos cada componente deste diagrama em detalhe.

No processamento de linguagem natural, um documento é normalmente representado por um *bag of words* (como já mencionamos anteriormente) que é na realidade uma matriz de palavras e documentos. Como assumimos N documentos e V palavras, o *bag of words* de um corpus é uma matriz de N x V. Um exemplo de um BoW é apresentado na tabela seguinte.

	d1	d2	d3	d4	d5	d6	d7	d8
palvara1	2	0	3	0	0	0	0	0
palavra2	0	5	0	0	0	0	0	0
palavra3	1	2	0	0	0	0	0	0
palavra4	0	0	3	6	0	0	1	0
palavra5	0	0	0	0	3	4	0	1

Figura 2.7: Exemplo de um Bag of Words

De acordo com a tabela apresentada na figura 2.7 como exemplo, existem cinco palavras (palavra1, palavra2, palavra3, palavra4 e palavra5) e oito documentos (d1-d8) neste corpus. Os valores $W_{i,j}$ na matriz representam a frequência da palavra i no documento j. Por exemplo, $W_{3,1}$ = 1 significa que a palavra3 ocorre 1 vez no documento d1. Este conjunto de documentos constitui o nosso corpus e este conjunto de palavras o nosso vocabulário. O BoW é uma representação simplificada do corpus que serve como dado de input na construção de um modelo de tópicos. A partir da representação anterior podemos deduzir que a ordem das palavras de um documento não afeta a representação do BoW. Dito de outra forma, as palavras no documento são permutáveis. Além disso, os documentos de um corpus são independentes: não há qualquer relação entre os documentos. A permutabilidade de palavras e documentos podem ser considerados como os dois

pressupostos fundamentais da identificação de tópicos.

Num BoW, a dimensionalidade do espaço de palavras pode ser enorme, e o BoW reflete apenas as palavras dos textos originais. Em contraste, o aspeto mais importante que se espera saber sobre um documento são os temas que aborda e não as palavras que o constituem. O objetivo da identificação de tópicos é descobrir os temas que percorrem um corpus através da análise das palavras dos textos originais. Chamamos a estes temas obtidos durante o treino do modelo de 'tópicos'. Na identificação de tópicos, um tópico é visto como uma distribuição de probabilidade sobre um vocabulário fixo. Cada tópico é uma mistura de palavras num vocabulário. Da mesma forma, cada documento é uma mistura de tópicos. Acima de tudo, a ideia chave por detrás da identificação de tópicos é que os documentos mostram múltiplos tópicos, e portanto a questão-chave da identificação de tópicos é a descoberta de uma distribuição de tópicos sobre cada documento e uma distribuição de palavras sobre cada tópico. E isso é conseguido através do treino de um modelo.

No treino de um modelo, para realmente determinar os tópicos num corpus, a identificação de tópicos necessita de simular o processo generativo através do qual os documentos são criados, de modo a conseguir fazer *reverse engineering* desse processo. Os documentos constituintes de um corpus foram gerados por algum processo complexo subjacente, que não é conhecido. Modelamos o processo real através de um sintético, que se aproxima do processo real, e tentamos encontrar parâmetros deste processo sintético que se ajustem bem aos dados. Este processo sintético é referido como o processo generativo do algoritmo, que se revela diferente de acordo com o algoritmo em utilização.

O problema computacional central para a identificação de tópicos é de como utilizar os documentos em estudo para inferir a estrutura de tópicos ocultos. Esta tarefa pode ser pensada como uma 'inversão' do processo generativo. Estima-se a distribuição dos tópicos e a distribuição das palavras para cada tópico durante o treino. Dado um novo documento, é possível gerar a distribuição mais provável sobre os tópicos que geraram o documento.

No nosso projeto utilizamos 2 algoritmos diferentes para a identificação dos tópicos latentes no conjunto de tweets recolhidos e achamos por isso necessário descrever o seu funcionamento em maior detalhe nas secções seguintes.

2.5.3.1 Latent Dirichlet Allocation (LDA)

Latent Dirichlet Allocation (LDA) ou Alocação de Dirichlet latente é uma das abordagens mais populares utilizadas na identificação de tópicos numa variedade de aplicações. LDA foi desenvolvido em 2003 pelos investigadores David Blei, Andrew Ng e Michael Jordan [20] como uma generalização de uma abordagem anterior de nome Análise probabilística de semântica latente (pLSA), ou *Probabilistic latent semantic analysis* em inglês. Apesar dos métodos possuírem bastantes semelhanças, algumas das vantagens obtidas através desta nova abordagem incluem a produção de uma melhor desambiguação de palavras, uma atribuição mais precisa dos documentos

aos tópicos, diminuição da vulnerabilidade ao *overfitting* com o aumento do tamanho do corpus, etc.

O algoritmo LDA, de uma forma simplificada, utiliza um modelo probabilístico generativo e distribuições de dirichlet para conseguir descobrir tópicos que estão ocultos (latentes) num conjunto de documentos de texto, inferindo possíveis tópicos com base nas palavras contidas nos documentos. Para facilitar a compreensão, achamos relevante uma analogia entre a distribuição de dirichlet e a distribuição normal, uma vez que a distribuição normal é um conceito familiar a grande parte da população.

A distribuição normal é uma distribuição de probabilidade sobre todos os números reais. É descrita por uma média e uma variância. A média é o valor esperado desta distribuição, e a variância diz-nos o quanto podemos esperar que as amostras se desviem da média. A distribuição de dirichlet é também uma distribuição de probabilidade, mas não é uma amostragem a partir do espaço dos números reais. Em vez disso, é uma amostragem sobre um simplex de probabilidade. Um simplex de probabilidade é um espaço matemático onde cada ponto representa uma distribuição de probabilidade entre um número finito de eventos mutuamente exclusivos. Cada evento é muitas vezes chamado de categoria e normalmente a variável K é utilizada para denotar o número de categorias. Um ponto sobre um simplex de probabilidade pode ser representado por K números não negativos que somam 1. Por exemplo:

$$(0.7, 0.3)$$

$$(0.2, 0.1, 0.7)$$

$$(0.05, 0.1, 0.25, 0.1, 0.2, 0.3)$$

Estes números representam probabilidades sobre K categorias distintas. Nos exemplos acima, K assume valores de 2, 3, e 6 respetivamente. É por isso que também são chamadas distribuições categóricas. Quando estamos a lidar com distribuições categóricas e temos alguma incerteza sobre como é essa distribuição, a forma mais simples de representar essa incerteza como uma distribuição de probabilidade é a distribuição de dirichlet. Uma distribuição de dirichlet K-dimensional tem parâmetros K. Estes parâmetros podem ser qualquer número positivo. Por exemplo, um dirichlet de 4 dimensões pode ter este aspeto:

$$(24, 5, 33, 38)$$

Note-se que estes 4 parâmetros podem ser normalizados (divididos pela sua soma) para formar uma distribuição de probabilidade multiplicada por uma constante de normalização:

$$100 * (0.24, 0.05, 0.33, 0.38)$$

No caso da distribuição normal, a média e a variância dizem-nos que tipo de amostras devemos esperar. No caso da distribuição de dirichlet, as probabilidades que advêm da normalização anterior (24%, 5%, 33%, 38%) são o valor médio do dirichlet. Assim, todas as amostras dele serão centradas em torno desse simplex. A constante de normalização, 100 neste caso, não é a variância, mas está relacionada. Quanto maior for, mais próximas estarão as amostras da média. 100 é um peso bastante elevado, pelo que a maioria das amostras desta distribuição estarão próximas de (24%, 5%, 33%, 38%).

Como uma breve visão geral, podemos pensar na distribuição de dirichlet como uma 'distribuição sobre distribuições'. Na sua essência, responde à pergunta: 'tendo em conta este tipo de distribuição, quais são algumas distribuições de probabilidade reais que eu provavelmente irei encontrar?'

No LDA, processo generativo é definido por uma distribuição conjunta de variáveis ocultas e observadas. Se tivermos K tópicos que descrevem um conjunto de documentos, então a mistura de tópicos em cada documento pode ser representada por uma distribuição K-nomial, uma forma de distribuição multinomial.

Uma distribuição multinomial é uma generalização da mais familiar distribuição binomial (que tem 2 resultados possíveis, como por exemplo lançar uma moeda ao ar). Uma distribuição K-nomial tem K resultados possíveis (tal como num dado de K lados). No LDA, o dirichlet é uma distribuição de probabilidade sobre as distribuições K-nomiais das misturas de tópicos, mas há também outra distribuição de dirichlet usada no LDA, uma distribuição sobre as palavras em cada tópico. Assim, LDA utiliza duas distribuições de dirichlet no seu algoritmo. Ao utilizar um processo generativo e distribuições de dirichlet, LDA pode generalizar melhor para novos documentos depois de ter sido treinado num determinado conjunto de documentos. As distribuições de dirichlet permitem a amostragem de distribuições de probabilidade de um tipo específico.

O processo generativo do LDA funciona da seguinte forma: A partir de uma distribuição de dirichlet $Dir(\alpha)$, é retirada uma amostra aleatória que representa a distribuição dos tópicos, ou mistura de tópicos, de um determinado documento, representada por θ. A partir de θ, seleciona-se um determinado tópico Z com base nessa distribuição. A seguir, a partir de outra distribuição de dirichlet $Dir(\beta)$, seleciona-se uma amostra aleatória que representa a distribuição de palavras do tópico Z, representada por φ. A partir de φ, escolhemos uma palavra w. Formalmente, o processo para generativo pode para um corpus D com M documentos de comprimento N_i pode ser descrito da seguinte forma:

1. Escolher $\theta_i \sim$ Dir(α), onde i $\in$ {1,...,M}
2. Escolher $\varphi_k \sim$ Dir(β), onde k $\in$ {1,...,K}
3. Para cada uma das posições de palavras i, j, onde i $\in$ {1,...,M}, e j $\in$ {1,...,N_i}
    - Escolher um tópico $z_{i,j} \sim$ Multinomial(θ_i)

- Escolher uma palavra $w_{i,j} \sim \text{Multinomial}(\varphi_{z_{i,j}})$

Um aspeto importante a notar do LDA é a necessidade de definir o número de tópicos latentes antes da sua execução. Ao escolher o valor K estamos a afirmar que é possível descrever os documentos a analisar através de K tópicos.

Para identificar os tópicos constituintes de documentos, o LDA faz o oposto do processo generativo. O LDA retrocede desde o nível dos documentos para determinar os tópicos que provavelmente geraram o corpus. Supondo que possuímos um conjunto de documentos e que escolhemos um número fixo de tópicos K para descobrir, se quisermos utilizar o LDA para aprender a representação de tópicos de cada documento e as palavras associadas a cada tópico, utilizamos um processo de inferência estatística. No nosso caso, a implementação do LDA que utilizamos utiliza a amostragem de Gibbs. A amostragem de Gibbs é um algoritmo para a amostragem sucessiva de distribuições condicionais de variáveis, cuja distribuição sobre estados converge para a verdadeira distribuição a longo prazo.

A utilização da amostragem de Gibbs no LDA pode ser descrita da seguinte forma:

1. Atribuir aleatoriamente um dos K tópicos a cada palavra em cada documento.
2. Para cada documento d:
   - Para cada palavra w em d:
     - Assumir que todas as atribuições exceto a atual estão corretas.
     - Para cada tópico t em K, calcular 2 proporções:
       (a) p(tópico t | documento d) = a proporção de palavras no documento d que estão atualmente atribuídas ao tópico t.
       (b) p(palavra w | tópico t) = a proporção de atribuições ao tópico t sobre todos os documentos que provêm desta palavra w.
     - Multiplicar as duas proporções anteriores para cada tópico t: p(tópico t | documento d) * p(palavra w | tópico t).
     - Atribuir à palavra w o tópico com o maior valor do produto anterior.
3. Repetir o passo anterior durante um número predefinido de iterações.

Se observarmos os passos anteriores conseguimos ver que no passo 2 o algoritmo calcula a probabilidade condicional de dois componentes, um relacionado com a distribuição dos tópicos num documento e outro relacionado com a distribuição das palavras num tópico. Por conseguinte, cada atribuição de um tópico a uma palavra depende de ambas as probabilidades anteriores. De acordo com o modelo generativo, $p(tópico t|documento d) \times p(palavra w|tópico t)$ é essencialmente a probabilidade de um tópico t ter gerado a palavra w, por isso faz sentido que a nova atribuição de tópico a uma palavra dependa desta probabilidade (estamos a atribuir a uma palavra o tópico mais provável de a ter gerado). Por outras palavras, neste passo estamos a assumir que todas

as atribuições de tópicos exceto a atribuição da palavra a considerar estão corretas e estamos a atualizar a atribuição da palavra em questão com base no nosso modelo de como os documentos são gerados. Após repetir este passo um grande número de vezes, acaba-se por atingir um estado mais ou menos estável onde as atribuições podem ser consideradas como boas.

Sendo assim, podemos utilizar estas atribuições para estimar as misturas de tópicos de cada documento (contando a proporção de palavras atribuídas a cada tópico dentro desse documento) e as palavras associadas a cada tópico (contando a proporção de palavras atribuídas a cada tópico em geral).

Cada documento pode ser representado por:

$$D_i = peso_1 \times Tópico_1 + ... + peso_k \times Tópico_k \tag{2.26}$$

2.5.3.2 Non-Negative Matrix Factorization (NMF)

A *Non-Negative Matrix Factorization* (NMF) ou Fatorização Matricial Não Negativa é um modelo linear algébrico capaz de representar uma matriz de alta dimensão constituída apenas por coeficientes não-negativos através do produto de duas matrizes de baixas dimensões também não negativas. O modelo foi introduzido pela primeira vez por Paatero e Tapper em 1994 [66], e popularizado num artigo de Lee and Seung em 1999 [48]. Desde então, o número de publicações de referência à técnica tem crescido rapidamente. Suponhamos que fatorizamos uma matriz X em duas matrizes W e H para obter uma aproximação de baixo nível de tal forma que

$$X \approx WH \tag{2.27}$$

Vamos assumir que X é constituída por n observações cada uma com p dimensões/atributos e que cada coluna de X é uma observação, i.e. $X \in \mathbb{R}^{p \times n}$. Deste modo, podemos considerar o NMF como um método para extrair um novo conjunto de r atributos a partir dos dados originais. Tal como no LDA é necessário definir o grau de fatorização, r, antes da sua execução.

Figura 2.8: Exemplo da matriz original X a ser aproximada

Nós queremos reduzir as dimensões originais p para um número r. Sendo assim, vamos ter uma matriz $W \in R^{p \times r}$ e $H \in R^{r \times n}$.

A interpretação de W é que cada coluna é um elemento base. Por elemento base entendemos algum componente que se repete em todas as n observações originais. Estes são os blocos de construção fundamentais a partir dos quais podemos reconstruir as aproximações a todas as observações originais.

A interpretação de H é que cada coluna dá as 'coordenadas de uma observação' na base W. Por outras palavras, diz-nos como reconstruir uma aproximação ao ponto de dados original a partir de uma combinação linear dos blocos de construção em W.

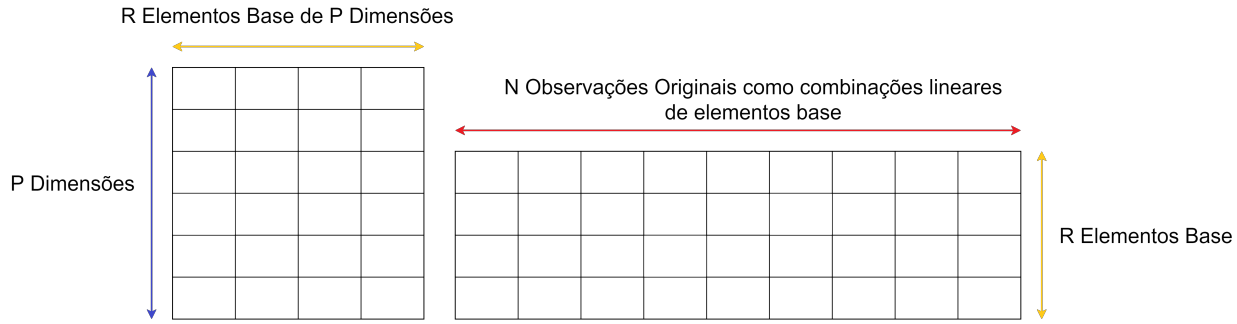

Figura 2.9: Exemplo das matrizes W e H, respetivamente

Para reconstruir uma aproximação de um valor original na matriz X, x_i, apenas precisamos de fazer uma soma pesada de elementos base, onde cada coluna de W é um elemento base e cada linha de H contém o peso associado a esse elemento base para a observação em questão:

$$x_i = \sum_{j=1}^{r} w_{ij} \times h_i \tag{2.28}$$

Por exemplo, dado um conjunto de imagens a preto e branco de um rosto contendo p pixéis, para cada imagem organizamos os valores de cada pixel num único vetor, de modo a que a entrada na i-ésima posição represente o valor do i-ésimo pixel. As linhas de uma matriz $X \in R^{p \times n}$ representam os p pixéis e as n colunas representam cada uma delas uma imagem.

O NMF aplicado a esta matriz vai produzir duas matrizes W e H. As colunas de W podem ser interpretadas como imagens (as imagens base), e H diz-nos como reconstruir uma aproximação a um determinado rosto original através das imagens base. Neste contexto, as imagens base podem ser interpretadas como características comuns aos rostos (olhos, narizes, bigodes, lábios, etc) enquanto as colunas de H indicam que característica está presente em que imagem.

A NMF é adequada para tarefas onde os fatores subjacentes podem ser interpretados como não-negativos. No caso do NMF aplicado à identificação de tópicos, considere a representação matricial de *bag-of-words* onde cada linha corresponde a uma palavra, cada coluna a um documento

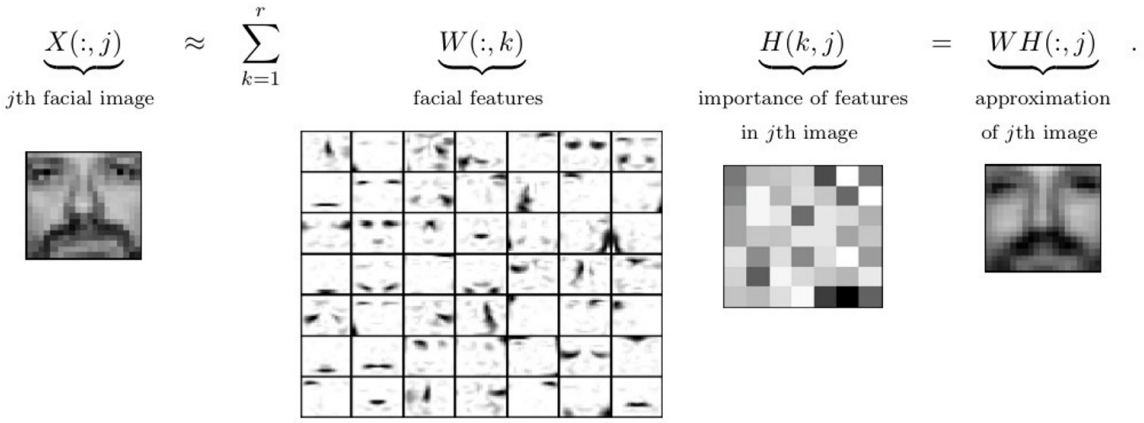

Figura 2.10: Exemplo do NFM aplicado a processamento de imagens [48]

e cada célula da matriz representa o peso de uma determinada palavra no respetivo documento (o peso pode ser a frequência absoluta, uma frequência ponderada por tf-idf ou algum outro esquema).

Quando decompusermos esta matriz vamos obter duas matrizes W e H. As colunas de W podem ser interpretadas como documentos base (*bag of words*). Estes documentos base são uma representação dos tópicos latentes no corpus, são essencialmente conjuntos de palavras encontradas simultaneamente em diferentes documentos. H diz-nos como somar as contribuições de diferentes tópicos para reconstruir a mistura de palavras de um dado documento original. Portanto, dado um conjunto de documentos, NMF identifica tópicos e simultaneamente classifica os documentos entre estes diferentes tópicos. Temos mais uma vez uma representação de cada documento como uma soma ponderada dos seus tópicos de composição.

Por exemplo, se um assumirmos que o corpus consiste num conjunto de tweets por parte de múltiplas HEI. A palavra 'professor' seria suscetível de aparecer em publicações relacionadas com a educação, e por isso vai coocorrer com palavras como 'estudante' e 'campus'. Portanto, estas palavras seriam provavelmente agrupadas num vetor de componentes 'educação', e cada publicação teria um certo valor/peso associado ao tópico 'educação'.

De modo a realizar o NMF formalizamos uma função objetiva e otimizamo-la iterativamente. O NMF é um problema *NP-hard* em geral, pelo que visamos encontrar um bom mínimo local. Para medir a qualidade da aproximação é necessário definir funções de custo, utilizando uma métrica de proximidade entre a matriz original, V, e a aproximação resultante, WH. Duas métricas comuns utilizadas são o a distância euclidiana, dada por

$$||X - WH||^2 = \sum_{ij}((X)_{ij} - (WH)_{ij})^2 \tag{2.29}$$

e a divergência (generalizada) de Kullback-Leibler, dada por

$$D(X||WH) = \sum_{ij} \left((X)_{ij} log \frac{(X)_{ij}}{(WH)_{ij}} - (X)_{ij} + (WH)_{ij} \right) \tag{2.30}$$

Ao minimizar estas métricas, sujeitas às restrições $W_{ij} \geq 0$ e $Hij \geq 0$, conseguimos obter uma boa aproximação da matriz original X.

2.5.3.3 Avaliação dos Modelos

Depois de aplicado com sucesso a identificação de tópicos a uma coleção de documentos, é necessário encontrar uma forma de medir o seu sucesso em identificar um conjunto útil de tópicos, ou seja, é necessário existirem parâmetros utilizados na avaliação do modelo resultante. Existem várias formas de fazer esta avaliação, sendo elas a avaliação humana, métricas intrínsecas de avaliação ou métricas extrínsecas de avaliação.

A avaliação humana consiste na interpretação dos tópicos, das palavras presentes nos tópicos e dos tópicos presentes nos documentos e identificar tópicos que 'não pertencem' a um documento ou palavras que 'não pertencem' a um tópico. A utilidade do modelo é determinada através do parecer de um ator humano com conhecimento do domínio.

As métricas extrínsecas de avaliação consistem na determinação do desempenho do modelo ao executar tarefas predefinidas, como a classificação. Ou seja, dado um modelo treinado num conjunto de dados previamente catalogados, avaliar o desempenho do modelo na atribuição correta de um tópico a um documento na fase de previsão.

Por fim, temos as métricas intrínsecas de avaliação que nos permitem justificar a seleção do modelo com base nas características do próprio. Vejamos a métrica intrínseca de avaliação: perplexidade. A perplexidade capta a 'surpresa' de um modelo quando fornecido com novos dados nunca antes observados. Pode-se pensar na métrica de perplexidade como a medida da probabilidade de alguns novos dados, dado o modelo que foi aprendido anteriormente. Ou seja, até que ponto o modelo representa ou reproduz as estatísticas de dados não observados. Contudo, estudos recentes demonstraram que a probabilidade preditiva (ou equivalentemente, perplexidade) e o juízo humano não estão muitas vezes correlacionados, e mesmo por vezes ligeiramente anti correlacionados [26]. Foi possível provar que, dado um tópico, as cinco palavras com maior frequência que lhe pertencem não são normalmente boas para descrever uma ideia coerente ou que pelo menos não são suficientemente boas para ser capazes de reconhecer uma palavra intrusa.

Esta limitação da medida da perplexidade serviu de motivação para mais trabalho na tentativa de modelar o parecer humano, e portanto a coerência de um tópico. A medida de coerência atribui uma pontuação a um único tópico (entre 0 e 1) ao medir o grau de semelhança semântica entre as palavras de pontuação elevada pertencentes ao mesmo. Esta medida ajuda a distinguir entre tópicos que são semanticamente interpretáveis e tópicos que são produto de inferência estatística.

No contexto no qual estamos inseridos, devido à inexistência de *labels* atribuídas com o

respetivo tópico às publicações da HEI recolhidas, empregamos a avaliação humana e as métricas intrínsecas de avaliação para determinar a utilidade e qualidade dos nossos modelos.

Capítulo 3

Revisão Bibliográfica

A revisão do estado da arte foi feita através do estudo de artigos que identificamos como relevantes através do nosso conhecimento prévio do domínio, assim como através da seleção de artigos complementares resultante de uma pesquisa sistemática utilizando os motores de busca académicos.

O tema da dissertação incide essencialmente sobre a utilização das redes sociais pelas instituições de ensino superior e a estratégia segundo a qual as utilizam. Como componentes acessórios do tema agregador temos o processamento de linguagem natural, reconhecimento de padrões, análise de sentimento e identificação de tópicos. Sendo assim, determinamos um conjunto de critérios segundo os quais realizamos a filtragem das fontes relevantes que se enquadrem nesses temas, encontradas pelos motores de busca.

Na seleção destas mesmas fontes, escolhemos apenas artigos que focam no uso de redes sociais por parte das instituições em si, e não pelo seu corpo docente ou os seus alunos matriculados, pois apesar de haver uma abundância de pesquisa nesse aspeto, esse tipo de artigos não auxiliam na determinação da estratégia de comunicação das instituições. Focamo-nos também na seleção de artigos que mencionam o papel das redes sociais por parte das HEI como instrumentos de marketing e comunicação, dispensando assim outros que abordam o uso das redes sociais como ferramenta de ensino. Por fim, procuramos dar preferência a artigos de avaliação comparativa das diversas técnicas/ferramentas que necessitaríamos de utilizar para cumprir com o nosso objetivo, nomeadamente, técnicas/ferramentas de pré-processamento, tratamento de linguagem natural, análise de sentimento e deteção automática de tópicos, de forma a conseguirmos, através dos mesmos, determinar o melhor plano de atuação, utilizando os métodos mais eficientes e apropriados. A obtenção destas fontes foi feita através das seguintes *queries* e/ou palavras-chave: 'social media comunication in higher education', 'social media in higher education', 'social media marketing', 'sentiment analysis', 'twitter pre-processing', 'twitter topic model' e 'NPL library comparison'.

No tópico em questão, existem já esforços feitos no sentido de aprofundar o conhecimento sobre as diferentes estratégias empregues pelas instituições. Nesta secção, descrevemos algumas

das abordagens anteriores mais relevantes, interpretamos os seus resultados e comparamos a sua metodologia com a implementada nesta dissertação.

Ashley e Tuten (2014)[18] realizaram um estudo das estratégias presentes no conteúdo partilhado nas redes sociais por parte das top 100 marcas do ranking da Interbrand - Melhores Marcas Globais. Os resultados obtidos foram sob a forma de orientações/melhores práticas genéricas como manter uma forte presença social, publicar conteúdo novo e implementar incentivos que promovam interação.

Peruta e Shields (2016)[68] analisaram as diferenças entre frequência de publicações no Facebook, tipos de publicações e interação/resposta de faculdade de diferentes tipos (artes liberais, públicas, privadas), utilizando uma amostra de 66 instituições norte americanas e concluíram que o tipo de universidade teria impacto na estratégia de comunicação e na interação resultante, sendo que as universidades de artes liberais eram alvo de maior interação do que as restantes. Concluíram também que padrões de publicação mais frequentes resultam em menos interação/resposta em cada publicação individual.

Oliveira e Figueira (2017)[64] introduzem métodos de análise das estratégias de publicação no Facebook de instituições de ensino superior, tendo como base a população das Instituições Politécnicas de Ensino Superior Português e utilizando *clusters* para a classificação de acordo com modelo de linhas editoriais pré-definidas (não obtidas a partir dos dados) e Indicadores Chave de Performance (KPI) para uma posterior avaliação de eficiência de cada categoria editorial. A atribuição das áreas editoriais foi feita através do um modelo *ensemble* constituído por 6 classificadores proeminentes (máquinas de vetores de suporte, random forests, logiBoost, k-Nearest Neighbours, perceptrons multicamadas e redes neuronais profundas) utilizando como dados de treino publicações classificadas manualmente como pertencentes a uma das 7 áreas pré-definidas.

Das 43 instituições em análise, foram detetadas 43 estratégias diferentes de conteúdo, porque a intensidade de a comunicação (número de publicações) em cada uma das sete áreas editoriais é diferente para cada instituição, levando assim a 43 conjuntos distintos de combinações. Para diferenciar estratégias de conteúdo sem ceder a uma análise não produtiva ou não generalizável, foi necessário encontrar padrões ou grupos de HEI com estratégias de conteúdo semelhantes entre eles. Para esse efeito, foi realizada uma análise de *clustering* utilizando o método *k-means*. Com base nesta análise, foi possível concluir que o público demonstra uma maior propensão para interagir com mensagens relacionadas com a identidade e imagem de uma HEI, sendo este elemento considerado um elemento-chave a incluir na conceção de uma estratégia de comunicação.

Omran e Treude (2017)[13] analisaram 1350 artigos de conferência na área de engenharia de software e descobriram que, dos 232 artigos que mencionavam *natural language*, apenas 33 referiam a biblioteca específica utilizada e que apenas dois desses artigos forneciam algum tipo de justificação rudimentar para a escolha da mesma. De seguida, procederam à comparação do output de 4 bibliotecas populares para o tratamento de limagem natural (SyntaxNet da Google, Stanford CoreNLP, NLTK e spaCy) numa amostra de 1116 *tokens* manualmente anotados com a

part-of-speech tag correta e concluíram que, a biblioteca com melhor performance (baseada na precisão) nas tarefas analisadas, *part-of-speech tagging* e tokenização, foi a spaCy, que não foi utilizada em nenhum dos artigos revistos.

Figueira (2018a) [34] propõe um modelo em 3 passos capaz de analisar e avaliar a estratégia de comunicação no Facebook de uma determinada HEI, com base numa amostra de 5 Instituições de Ensino Superior com melhor ranking mundial em 2017, assim como um modelo de previsão de interação, capaz de determinar quais estratégias que terão melhor resposta. Os 3 passos passam por 1) compreender a estratégia de comunicação das HEI através de métodos de *machine learning* para obter atributos relacionados com os padrões de publicação das mesmas, 2) a comparação das diferentes HEI através de métricas manufaturadas capazes de determinar a eficiência de cada instituição (uma razão entre o esforço de publicação e a resposta obtida) e, por fim, 3) a construção de 3 modelos para a previsão da: interação nos 3 dias seguintes à publicação, o sentimento médio das respostas nos 4 dias após a publicação e o enfraquecimento das respostas 3 dias após a publicação. Os classificadores obtidos demonstraram uma exatidão acima de 80% e um F1 *score* acima de 84

Figueira (2018b) [35] propõe um modelo automático para determinar o comportamento de publicação e a identificação da estratégia de publicação de cada HEI na rede social Facebook, utilizando uma amostra de 10 universidades de topo do ranking mundial, focando o estudo na frequência e intensidade de publicações e das respostas assim como a analise do conteúdo das mesmas e categorização da estratégia editorial conforme 4 categorias pré estabelecidas. Uma das principais conclusões obtidas é de que as HEI não publicam em função de um possível envolvimento futuro. O que sucede na maioria dos casos é que o Facebook é utilizado como um meio de comunicação social de notícias sobre a instituição e os seus agentes.

Oliveira e Figueira (2018)[63] apresentam um estudo longitudinal com um alcance de 4 anos sobre a população total das Instituições Politécnicas de Ensino Superior Português onde analisam a taxa de adesão às redes sociais (Facebook), o desempenho e eficiência de cada instituição abordada e propõem uma metodologia para a avaliação comparativa do desempenho das instituições no setor do ensino superior. Deste trabalho foi possível concluir que nos 4 anos estudados (com início em 2014) é possível observar um maior esforço por parte das HEI no desenvolvimento e manutenção de uma presença mais expressiva nas redes sociais, mas também que nem todas as organizações se demonstraram eficientes nessa alocação de esforço. Do mesmo modo, foi possível também identificar as tendências nas estratégias de comunicação das HEI e dos tipos de publicação que causam um maior envolvimento (link, foto, status, vídeo, etc).

Symeonidis, Effrosynidis e Arampatzis (2018)[84] realizaram uma comparação experimental de 16 técnicas de pré-processamento utilizando 4 algoritmos de *machine learning* populares(*Linear SVC*, *Bernoulli Naive Bayes*, *Logistic Regression* e *Convolutional Neural Networks*). As técnicas foram avaliadas de acordo com a sua precisão na classificação do sentimento para 2 conjuntos de dados diferentes (manualmente catalogados), recolhidos da rede social Twitter. As experiências realizadas demonstraram que, na análise de sentimento em dados com origem no Twitter, algumas técnicas de pré-processamento fornecem melhores resultados na classificação de ambos os

conjuntos de dados utilizados, enquanto outras diminuem a precisão. As técnicas recomendadas são *stemming*, a substituição de repetições de pontuação, e remoção de números. As não recomendadas incluem a remoção da pontuação, utilização de palavras capitalizadas, substituição de gíria/calão, substituição das negações por antónimos, e correção ortográfica.

Albalawi, Yeap e Benyoucef (2020) [14] investigaram o tema da identificação de tópicos e as suas áreas de aplicação, métodos e ferramentas comuns. Para além disso, examinaram e compararam cinco métodos de identificação de tópicos frequentemente utilizados, aplicados a dados sociais textuais curtos, para demonstrar os seus benefícios na deteção prática de tópicos importantes. Estes métodos são a Análise Semântica Latente (LSA), Alocação de Dirichlet latente (LDA), Fatorização Matricial Não Negativa (NMF), Projeção aleatória e Análise de Componentes Principais (PCA). Foram selecionados dois conjuntos de dados textuais para avaliar o desempenho dos métodos de identificação de tópicos incluídos com base na qualidade dos tópicos e algumas métricas estatísticas de avaliação padrão, como a sensibilidade, a precisão, o F-*score* e a coerência. Como resultado desta avaliação, é possível concluir que o LDA e NFM foram métodos que permitiram extrair tópicos mais significativos, de maior qualidade e maior coerência.

Com base estudos previamente mencionados, conseguimos adotar práticas comuns estabelecidas neste tipo de análise, nomeadamente as operações de NLP a realizar, as bibliotecas e algoritmos a utilizar, as técnicas a empregar na caracterização das estratégias de comunicação, assim como ferramentas eficazes na visualização das mesmas. Do mesmo modo, conseguimos identificar aspetos que gostaríamos de alterar nas abordagens anteriores de modo a cumprir com o nosso objetivo com um melhor desempenho. Destes aspetos, surgiu a nossa metodologia, que pode ser explicada de uma forma simples através das diferenças que executamos entre as abordagens anteriores e a nossa.

Com este projeto procuramos estudar a estratégia de comunicação das HEI no Twitter, sendo que os estudos [34, 35, 63, 64, 68] utilizam publicações recolhidas das páginas de Facebook das instituições a analisar. Esta diferença pode parecer trivial, mas a estratégia de comunicação de uma HEI pode variar de acordo com a rede social a qual está empregue, nem que não seja pelo simples facto da demográfica dos seus utilizadores ser significativamente diferente [25]. No nosso caso, o Twitter possui ainda mais diferenças, sendo umas das mais facilmente observáveis o tamanho máximo de publicação que esta plataforma possibilita aos seus utilizadores (280 caracteres), revelando-se bastante restritivo neste aspeto. Como tal, esperamos encontrar algumas diferenças das estratégias previamente identificadas.

Ao contrário da metodologia utilizada nos artigos [34, 35, 63, 64], onde a classificação das estratégias de comunicação das HEI era feita de acordo com um modelo editorial imposto pelos autores, e cuja a classificação recorria ao treino supervisionado de classificadores numa amostra das publicações a analisar, classificada manualmente, o nossa abordagem pretende não só utilizar esse tipo de modelo editorial estabelecido com base em conhecimento de domínio mas também utilizar técnicas de *machine learning* não supervisionadas de análise de conteúdo para obter os tópicos e áreas editoriais nas quais as HEI vão incidir a sua comunicação e comparar a qualidade

dos modelos resultantes, quer na interpretação dos tópicos agregadores, quer na sua capacidade preditiva da interação futura (à semelhança do que foi realizado na terceira fase do artigo [34]) ou do tópico a ser abordado numa publicação seguinte. Deste modo pretendemos conseguir uma melhor caracterização das estratégias de comunicação e mais fiel à realidade.

Nos artigos [63, 64] as HEI utilizadas como amostra para o estudo da estratégia de comunicação são Instituições Politécnicas de Ensino Superior Português. No nosso projeto, ao selecionar as instituições de ensino superior com base no ranking de 2019/2020 do CWUR pretendemos obter estratégias de comunicação mais bem definidas, sendo que as instituições de topo possuirão, em teoria, estratégias mais bem elaboradas e trabalhadas.

Pretendemos também fazer uma análise mais detalhada da que foi conseguida no artigo [68], onde apenas diferenciam as 66 instituições em 3 categorias e as estratégias são analisadas essencialmente na base de frequência de publicação e quantidade de interação, com pouco ênfase no conteúdo, ou no artigo [18], onde as estratégias identificadas são sob a forma melhores práticas generalizadas. Apesar de também apresentarmos caracterizações fiéis das instituições de acordo com a sua frequência de publicações e quantidade de interações, o nosso foco será essencialmente no conteúdo das publicações em análise, de que forma este conteúdo se insere nas estratégias de comunicação em questão e quais os tópicos/temas agregadores de público ou com melhor resposta.

Como mencionado anteriormente, utilizando os resultados do artigo [84] conseguimos selecionar as técnicas de processamento a utilizar na nossa amostra de tweets, nomeadamente a *lemmatization*, substituição de URLs, menções de utilizadores e contrações, remoção de números, entre outras. Do mesmo modo, com base nos resultados do artigo [13] conseguimos determinar a biblioteca de processamento de linguagem natural a utilizar, a SpaCy. Por fim, com base nos resultados do artigo [14] obtivemos também os métodos de identificação de tópicos capazes de gerar tópicos de melhor qualidade, LDA e NMF, utilizando dados recolhidos de fontes semelhantes à utilizada nesta dissertação (dados de curta extensão, provenientes de redes sociais/*microblogs*).

Capítulo 4

Metodologia

A nossa metodologia consiste nas quatro etapas seguintes: começa pela caracterização das estratégias de comunicação das HEI através de fatores externos às áreas editoriais presentes no conteúdo, de modo a obter uma boa representação das práticas e padrões de publicação das instituições nesse contexto. De seguida, avaliamos e interpretamos os modelos resultantes de diferentes abordagens de identificação de tópicos que atribuem um padrão editorial às estratégias de comunicação das HEI de acordo com o conteúdo dos seus respetivos tweets. Em terceiro lugar, realizamos uma breve análise do sentimento predominante das publicações, determinamos a sua evolução ao longo do período estudado e de que forma a distribuição de sentimento difere em cada área editorial. Finalmente, utilizando os modelos obtidos anteriormente, fazemos uma atribuição de uma área editorial a cada publicação no nosso conjunto de dados e através de aprendizagem supervisionada, avaliamos a capacidade preditiva dos nossos modelos ao tentar obter previsões para o envolvimento futuro de um tweet assim como para o próximo tópico a ser abordado por uma determinada instituição. Esta metodologia é explicada em pormenor ao longo deste capítulo.

4.1 Análise Exploratória da Frequência de Publicações

Nesta primeira etapa, os dados recolhidos em formato JSON através da API do Twitter durante o período selecionado, correspondente a um ano letivo, são convertidos para uma *dataframe* para facilitar a manipulação. De seguida, os dados são agrupados por instituições e é realizada uma análise comparativa dos padrões de publicação das instituições envolvidas. Isto passa por analisar, para cada instituição: o volume de publicações no período de tempo observado, a distribuição das métricas que quantificam o envolvimento (número de retweets e de favoritos), a relação entre o número de subscritores da instituição e o envolvimento observado para a mesma, o período de atividade, a evolução mensal do número de publicações, número de retweets e número de favoritos e os horários preferenciais de publicação.

4.2 Identificação de Tópicos

Nesta etapa, o campo de texto dos dados recolhidos é sujeito a um processo de limpeza, onde são removidos ou alterados diversos aspetos através das bibliotecas SpaCy[1] e Emoji[2]: são removidos URLs, repetições de pontuação, o texto 'RT' a sinalizar um retweet, menções de utilizadores, é feita a conversão para letra minúscula, tokenização, remoção de *stop-words*, *lemmatization*, conversão de emojis para texto corrente e, por fim, são criados bigramas e trigramas de acordo com os *tokens* obtidos e é feita a *POS tagging* (mantendo apenas nomes, verbos, adjetivos e advérbios). De seguida, realizamos 2 experiências diferentes para a obtenção do melhor modelo capaz de caracterizar as áreas editoriais predominantes no conteúdo.

Primeiramente, na experiência de aprendizagem não-supervisionada, utilizamos um processo iterativo através do qual criamos um conjunto de modelos LDA e NMF com diferentes valores para o número de tópicos e comparamos a sua qualidade de modo a selecionar o melhor modelo resultante. De seguida, utilizamos um modelo editorial definido com base na metodologia utilizada pelo Centro para Ranking Mundial de Universidades (CWUR) [5], servindo-nos dos 4 indicadores robustos que esta metodologia refere como 4 novas áreas editoriais distintas ('educação', 'emprego', 'corpo docente', 'investigação'), acrescentando apenas 2 áreas editoriais com base no nosso conhecimento do domínio e na situação pandémica atual ('saúde' e 'sociedade'). Para cada experiência, selecionamos o melhor modelo resultante das respetivas operações e fazemos a atribuição de um tópico dominante a cada publicação no nosso conjunto de dados. Com base nessas atribuições, torna-se possível fazer uma análise detalhada da relação entre as métricas de envolvimento e as diferentes áreas editoriais, da estratégia editorial característica de cada instituição e da evolução temporal dessa estratégia.

4.3 Análise de Sentimento

Após o passo anterior, para cada experiência, segue uma análise do sentimento na qual, através da biblioteca TextBlob, obtemos os valores de polaridade para cada publicação e, através de um processo de discretização, conseguimos obter também o sentimento predominantemente presente no seu conteúdo (Negativo, Neutro, Positivo). Isto torna possível a análise da evolução temporal do sentimento, da sua relação com as métricas de envolvimento, da sua distribuição por instituição e da sua distribuição por tópico/área editorial.

4.4 Análise Preditiva

Por fim, testamos a capacidade preditiva dos modelos resultantes de cada uma das experiências. Primeiramente, treinamos nos dados já catalogados, de acordo com o respetivo modelo, 10

[1] https://spacy.io/
[2] https://pypi.org/project/emoji/

algoritmos diferentes de aprendizagem supervisionada apresentados na secção 2. A tarefa trata de prever o número de retweets de uma determinada publicação com base no seu tópico dominante, horário de publicação, número de seguidores da HEI, polaridade, o seu 'retweet status' e o seu 'quote status' (se a publicação é um retweet ou uma quote de uma outra publicação anterior). No final, comparamos o desempenho de cada modelo de acordo com as métricas de avaliação MSE, RMSE, MAE e R2.

Do mesmo modo, utilizando 8 algoritmos diferentes de aprendizagem supervisionada também mencionados na secção 2 conseguimos treinar 8 diferentes modelos na tarefa de prever a área editorial da próxima publicação por parte de uma HEI com base numa janela temporal de publicações anteriores (os tópicos atribuídos aos 7 dias anteriores), o horário da publicação, média do número de retweets das últimas publicações e média de favoritos das últimas publicações. No final, comparamos mais uma vez o desempenho de cada modelo de acordo com as métricas de avaliação exatidão, precisão, sensibilidade e F1-*score*, e comentamos a qualidade dos resultados obtidos.

Capítulo 5

Desenvolvimento e Implementação

Neste capítulo é explicado o desenvolvimento da dissertação com base na metodologia apresentada na secção 4. São apresentadas em detalhe as diferentes fases do desenvolvimento e modo como foram implementadas.

5.1 Conjunto de Dados

As HEI selecionadas para a nossa análise são: Caltech, Cambridge University, ETH Zurich, Harvard University, Imperial College, Johns Hopkins University, Massachusetts Institute of Technology (MIT), Stanford University, The University of Chicago, UC Berkeley, UCL e University of Oxford. Os dados foram recolhidos utilizando a API do Twitter, sendo constituídos por um conjunto de objetos Tweet representativos de todos os tweets das instituições selecionadas durante a janela temporal de 1 de Setembro de 2019 a 31 de Outubro de 2020.

Esta janela não foi selecionada de forma arbitrária. Embora nem todas as HEI comecem e terminem o seu ano letivo exatamente nestas datas, acreditamos que este período é o mais adequado para representar um ano letivo para a maioria das instituições envolvidas. Um aspeto interessante deste período, e o qual vamos explorar extensivamente na interpretação dos nossos resultados, é que nos permite fazer uma análise das estratégias de comunicação das HEI no contexto atual pandémico. Tendo em conta a data em que a Organização Mundial de Saúde (OMS) declarou a Covid-19 como Pandemia, 11 de Março de 2020, o período selecionado permite-nos também obter uma divisão algo equilibrada do conjunto de dados em períodos pré/pós Covid-19, com uma janela temporal próxima dos 6 meses para cada um. Isto torna possível obter alterações comportamentais das HEI conforme o avanço da pandemia, caracterizar diferentes padrões editoriais em períodos antes e durante a pandemia e comparar o impacto da mesma nas estratégias de comunicação das HEI selecionadas.

Os objetos Tweet presentes no conjunto de dados possuem uma longa lista de atributos de raiz, incluindo atributos fundamentais tais como 'id', 'created_at', e 'text'. O objeto Tweet é também o objeto 'pai' de vários objetos filho. Quando os tweets são apresentados em formato

JSON, eles são uma mistura de atributos de raiz e atributos filho (representados com através da notação {} no exemplo seguinte):

```
{
 "created_at": "Wed Oct 10 20:19:24 +0000 2018",
 "id": 1050118621198921728,
 "id_str": "1050118621198921728",
 "text": "Sample Text",
 "user": {},
 "entities": {}
}
```

Bloco de Código 5.1: Exemplo de tweeet JSON

Objetos filho como 'user' e 'entities' possuem os seus próprios atributos com informação relativa à sua função. De todos os objetos filho presentes nos atributos, o mais relevante na nossa análise foi o 'user', sendo que os restantes revelaram pouca ou nenhuma utilidade. O objeto 'user' contém metadados da conta de utilizador do Twitter que descrevem o utilizador do Twitter referenciado:

```
"user": {
    "id": 6253282,
    "id_str": "6253282",
    "name": "Twitter API",
    "screen_name": "TwitterAPI",
    "location": "San Francisco, CA",
    "url": "https://developer.twitter.com",
    "description": "The Real Twitter API."
    "verified": true,
    "followers_count": 6129794,
    "friends_count": 12,
    "listed_count": 12899,
    ...
    }
```

Bloco de Código 5.2: Exemplo do atributo user [6]

Através deste campo, torna-se possível a identificação da instituição responsável pelo Tweet juntamente com o número de seguidores da HEI identificada, duas informações essenciais para a análise comparativa.

Na tabela seguinte, apresentamos cada um dos atributos de um tweet, o seu tipo e a sua interpretação.

Tabela 5.1: Atributos do objeto Tweet

Atributo	Tipo	Descrição
created_at	String	Tempo UTC do momento de publicação do Tweet.
id	Int64	Representação em número inteiro do identificador único de um Tweet.
id_str	String	Representação em String do identificador único de um Tweet.
text	String	O texto UTF-8 contido na publicação.
source	String	Utilitário utilizado para publicar o Tweet, como uma string formatada em HTML.
truncated	Boolean	Indica se o valor do parâmetro de texto foi truncado.
in_reply_to_status_id	Int64	Nullable. Se o Tweet for uma resposta, este campo contém o ID do Tweet original (número inteiro).
in_reply_to_status_id_str	String	Nullable. Se o Tweet for uma resposta, este campo contém o ID do Tweet original (string).
in_reply_to_user_id	Int64	Nullable. Se o Tweet for uma resposta, este campo contém o ID do utilizador do Tweet original (número inteiro).
in_reply_to_user_id_str	String	Nullable. Se o Tweet for uma resposta, este campo contém o ID do utilizador do Tweet original (string).
in_reply_to_screen_name	String	Nullable. Se o Tweet for uma resposta, este campo contém o nome do utilizador do Tweet original (string).
user	User object	Objeto do utilizador que publicou o Tweet.
coordinates	Coordinates	Nullable. Representa a localização geográfica do Tweet, tal como relatado pelo utilizador ou pela aplicação cliente. A matriz de coordenadas internas é formatada como geoJSON (longitude, latitude).
places	Places	Nullable. Quando presente, Indica que o tweet está associado (mas não necessariamente originário de) um lugar.
quoted_status_id	Int64	Este campo só aparece quando o Tweet é uma citação. Contém o ID do Tweet citado (número inteiro).
quoted_status_id_str	String	Este campo só aparece quando o Tweet é uma citação. Contém o ID do Tweet citado (string).
is_quote_status	Boolean	Indica se o Tweet é uma citação

No total, o nosso conjunto de dados consiste em 18727 Tweets com os atributos anteriormente

quoted_status	Tweet	Este campo só aparece quando o Tweet é uma citação. Contém o Tweet original que foi citado.
retweeted_status	Tweet	Este campo só aparece quando o Tweet é um retweet. Contém uma representação do Tweet original que foi retweetado.
retweet_count	Int	Número de retweets do Tweet.
favorite_count	Int	Número de favoritos do Tweet.
entities	Entities	Entidades identificadas a partir do texto do Tweet.
favorited	Boolean	Nullable. Indica se o Tweet foi gostado pelo utilizador que o publicou.
retweeted	Boolean	Nullable. Indica se o Tweet foi retweetado pelo utilizador que o publicou.
possibly_sensitive	Boolean	Nullable. Este campo só aparece quando o Tweet contém um link. Indica se o URL mencionado contém conteúdo identificado como sensível.
lang	String	Nullable. Indica um identificador de idioma BCP 47 correspondente ao idioma detetado no texto do Tweet, ou 'und' se nenhum idioma for detetado.

apresentados, apesar de apenas alguns deles possuírem informação relevante para a nossa análise.

5.2 Análise Exploratória

Após a conversão dos dados em formato JSON para uma *dataframe* da biblioteca Pandas[1], começamos por obter informação sobre as suas características. Com uma análise inicial da frequência de publicação das diferentes instituições, obtivemos o seguinte gráfico de barras:

Ao observar a frequência de publicação dentro do período especificado (figura 5.1), pudemos determinar que o Imperial College foi a HEI que mais publicou, enquanto que a Universidade de Stanford foi a que menos publicou. Na verdade, os números são bastante diferentes para cada organização. Se os dividirmos por frequência de publicação, temos: Stanford, UCL e Caltech no mesmo grupo, com menos de 1k de publicações; Imperial College, Johns Hopkins e UC Berkeley com mais de 2k e, por último, as restantes instituições com valores intermédios. Estes valores resultam de médias entre 2 e 8 publicações por dia, dependendo da HEI em questão.

É importante também identificar as diferenças de dimensões entre as instituições envolvidas, visto que estas foram escolhidas de acordo com o ranking da CWUR, que em nenhuma altura quantifica a sua presença nas redes sociais (nomeadamente no Twitter).

[1] https://pandas.pydata.org

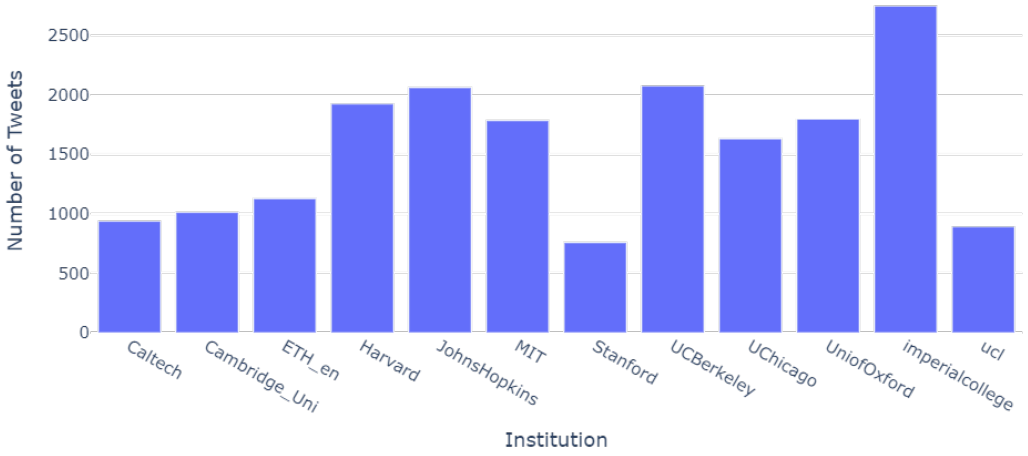

Figura 5.1: Volume de publicações das HEI

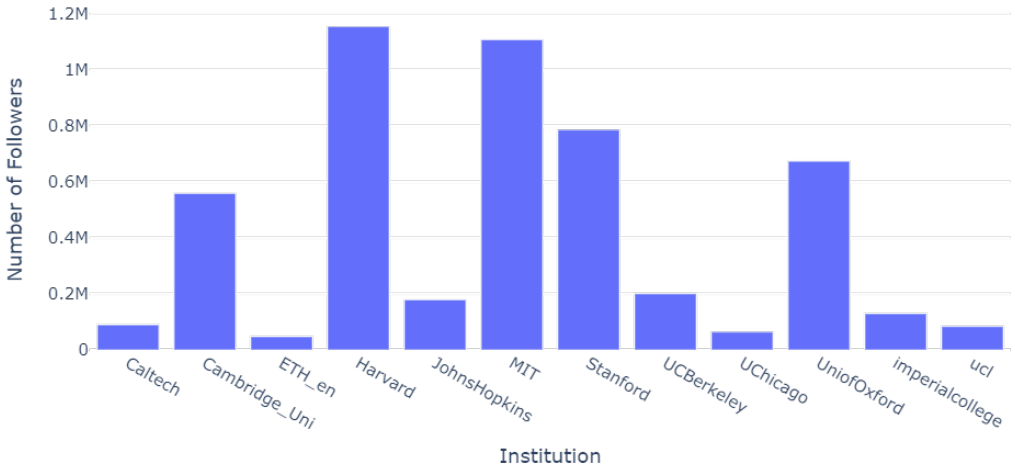

Figura 5.2: Número de seguidores por HEI

Através da figura 5.2 conseguimos perceber que existem diferenças significativas de dimensão entre as instituições. O número máximo de seguidores pertence a Harvard (1.15M) enquanto que o menor número de seguidores pertence a ETH Zurich (46.97k). Ao organizá-las por ordem de grandeza teríamos primeiro Harvard e MIT, com mais de 1M de seguidores, após estas teríamos Cambridge, Stanford e Oxford, com seguidores entre os 500k e 800k, de seguida Johns Hopkins, Imperial College e UCBerkeley, com um alcance de 100k a 200k e, por fim, Caltech, ETH Zurich, UChicago e UCL, cada uma com menos de 100k subscritores. É por isso importante ter em conta estas diferenças de dimensão aquando da análise do envolvimento das instituições.

Achamos também relevante observar a distribuição dos atributos representativos do envolvimento, o 'retweet_count' e 'favorite_count', numa tentativa inicial de identificar as instituições com maior sucesso/alcance nas suas publicações. O número médio de retweets e favoritos é também significativamente diferente de instituição para instituição. No que diz respeito aos retweets, varia de 10 (ETH Zurich) a 119 (Stanford) e o número médio de favoritos por tweet

varia de 13 (UCL) a 157 (Harvard). Os desvios-padrão para estas duas métricas em cada HEI são extremamente elevados: isto é o resultado de tweets 'virais' outliers com altas contagens de retweets e favoritos que inflacionam a distribuição. Apesar desses fenómenos serem também dignos de análise, focamos a nossa atenção na obtenção de representações fidedignas das métricas de envolvimento comuns para cada HEI, e isto envolve a remoção de outliers, identificados como os valores superiores a Q3 + 1.5 × IQR, onde Q3 representa a mediana dos pontos de dados mais altos e IQR representa o alcance entre Q3 e Q1 (a mediana dos dados). Isto proporcionou os seguintes resultados:

	Antes			Depois		
Instituição	média	σ	max	média	σ	max
Caltech	21.52	43.60	906	17.85	19.30	77
Cambridge	69.89	262.71	5855	36.84	36.22	167
ETH	13.94	26.13	298	9.45	10.80	45
Harvard	157.12	391.51	7790	92.79	52.51	260
JohnsHopkins	26.43	104.21	2156	5.47	10.02	42
MIT	71.14	155.54	3353	45.38	44.79	204
Stanford	103.36	232.45	5066	62.89	47.04	210
UCBerkeley	16.19	53.86	1558	5.42	9.63	37
UChicago	14.39	50.56	1473	5.62	8.01	32
UniofOxford	109.93	1663.60	61929	18.46	28.09	116
ImperialCollege	22.90	61.50	2226	11.74	12.32	54
UCL	12.59	38.68	672	5.08	6.78	27

Tabela 5.2: Estatísticas dos favoritos

	Antes			Depois		
Instituição	média	σ	max	média	σ	max
Caltech	100.20	678.32	12180	7.35	7.22	37
Cambridge	61.40	631.91	16237	19.11	13.14	63
ETH	9.66	42.56	1289	5.10	4.09	19
Harvard	41.23	107.18	2323	24.12	14.93	76
JohnsHopkins	23.39	59.89	1084	8.82	9.91	48
MIT	29.63	107.64	2895	17.57	10.60	52
Stanford	119.47	2205.66	60531	18.90	12.68	62
UCBerkeley	49.36	983.15	39247	6.45	4.95	25
UChicago	12.60	196.73	7919	4.32	3.49	17
UniofOxford	41.62	707.19	28650	6.89	10.36	45
ImperialCollege	14.87	123.36	5514	4.99	5.05	23
UCL	11.39	64.24	1805	4.03	4.83	22

Tabela 5.3: Estatísticas dos retweets

Com a remoção de outliers observamos valores bastante mais sensatos de ambas as métricas de envolvimento, onde o desvio padrão não assume valores extremos. Os valores médios passam para entre 4.03 (UCL) e 24.12 (Harvard) no que diz respeito ao número de retweets e para entre 5.08 (UCL) e 92.79 (Harvard) no que diz respeito ao número de favoritos. Com base nos novos valores, podemos representar a distribuição destes valores por instituição, sob a forma de diagramas de caixa:

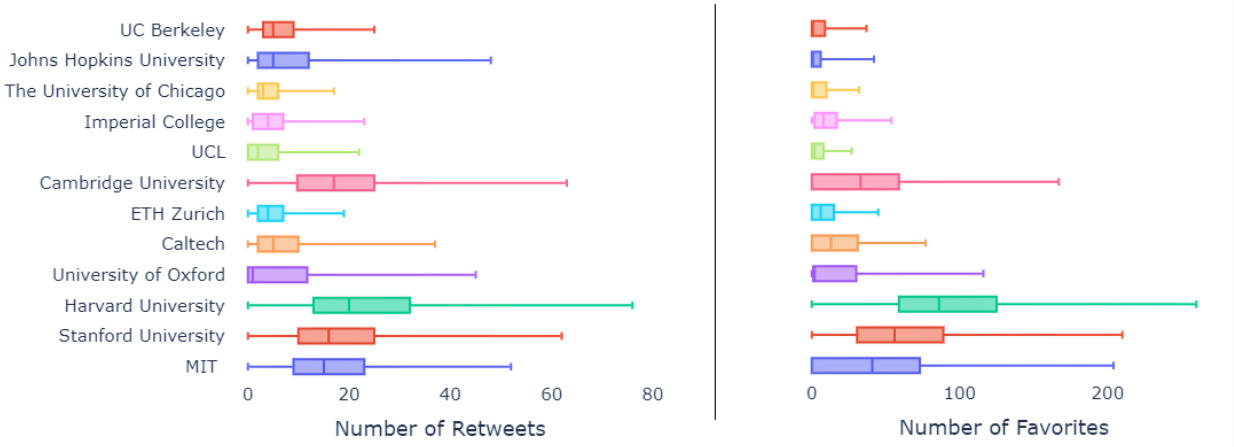

Figura 5.3: Distribuições de retweets e favoritos por HEI

Ao analisar a figura resultante, é possível observar diferenças significativas entre as instituições neste aspeto. Instituições como Cambridge, Harvard, Stanford e MIT, as instituições com maior número de seguidores, possuem medianas superiores às restantes HEI, num alcance [15,20] em número de retweets e [33,86] em número de favoritos. Estas universidades apresentam também os maiores intervalos interquartis em ambas as métricas, ou seja, os valores estão mais dispersos em torno da medida de centralidade. Conseguimos também observar que os valores máximos em ambas as métricas são superiores nas universidades mencionadas.

A universidade de Oxford, apesar de também pertencer ao grupo de universidades de maiores dimensões, acaba por fugir ao padrão identificado anteriormente, possuindo por vezes valores menos dispersos do que instituições mais pequenas (por exemplo, Caltech em número de retweets) assim como medianas de valor mais baixo (por exemplo, Imperial College em número de favoritos) e valores máximos menores (por exemplo, Johns Hopkins em número de retweets).

Apesar dessas exceções, uma interpretação possível desta informação seria através da existência de uma correlação entre o número de seguidores de uma HEI e o número de favoritos e retweets das publicações de sua autoria. De modo a verificar esta correlação, organizamos os diagramas de dispersão presentes na figura 5.4. Graças a ela conseguimos verificar uma característica dos dados que não foi aparente inicialmente, e que se tornou numa barreira na tentativa de verificação da correlação anterior. O campo 'followers_count' do atributo 'user' associado a um tweet possui a informação sobre o número de seguidores do utilizador responsável pela publicação.

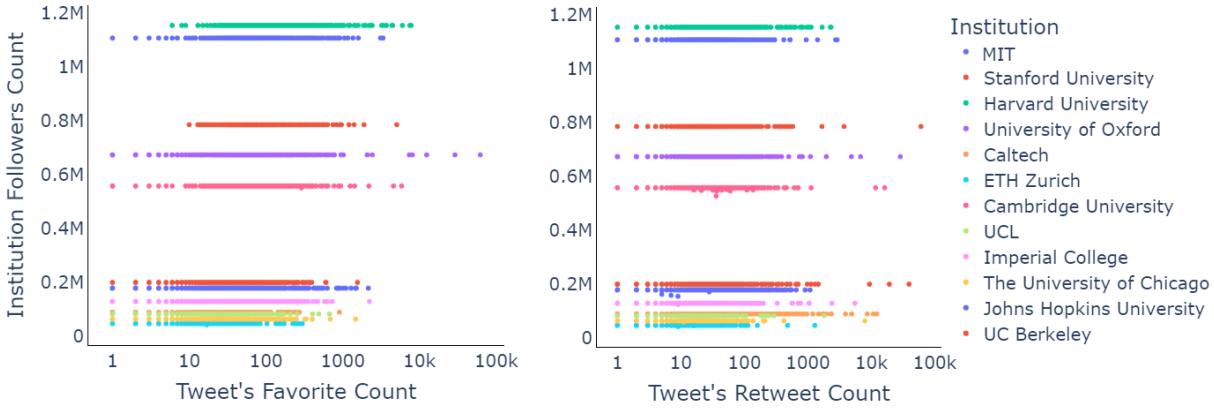

Figura 5.4: Relação de seguidores com favoritos/retweets

No entanto, a API do Twitter retorna o número de seguidores do utilizador no momento do pedido, e não na altura da publicação, e é por isso que o número de seguidores das diferentes HEI aparece quase estático, os tweets foram recolhidos simultaneamente e num curto espaço de tempo. Sendo assim, através dos diagramas anteriores, não é possível confirmar ou negar a existência de uma correlação entre número de seguidores e as métricas de envolvimento.

Passamos de seguida por analisar a distribuição temporal das publicações e os seus respetivos números de retweets e favoritos, na tentativa de identificar similaridades evidentes entre HEI que pudessem indicar a mesma tendência/evento. Para tal, agregamos os tweets por mês de acordo com as respetivas instituições e produzimos os seguintes gráficos de linhas:

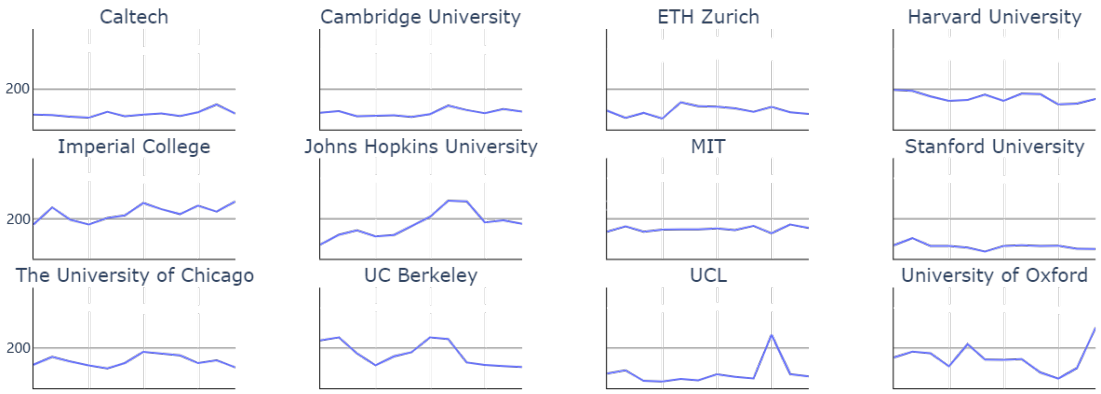

Figura 5.5: Evolução do número de publicações mensais por HEI

Para facilitar a comparação, utilizámos a mesma escala para todos os gráficos. Com base na figura anterior conseguimos concluir que grande parte das instituições possuem hábitos de publicação mensais relativamente estáveis, sendo Oxford, Johns Hopkins, UC Berkeley e UCL as exceções. A par de isso, pouca mais informação pode ser obtida através deste gráfico em relação a eventos ou tendências comuns. Avançamos por isso para os gráficos com a informação sobre o número de favoritos e retweets:

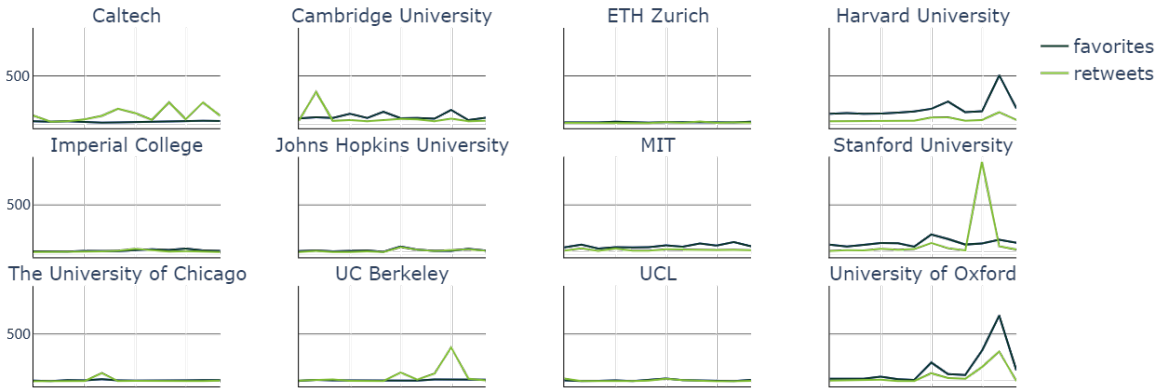

Figura 5.6: Evolução do número de retweets/favoritos mensais por HEI

Analisando as séries cronológicas representadas, podemos novamente encontrar inúmeras diferenças entre as 12 HEI. Contudo, existem algumas semelhanças no que diz respeito à sazonalidade e tendências que não podemos descartar. O primeiro aspeto evidente a salientar é que o número médio de favoritos e de tweets apresenta geralmente o mesmo comportamento devido à alta correlação entre os dois, ou seja, espera-se que os tweets com um elevado número de favoritos tenham também um elevado número de retweets (e vice-versa), embora, em Caltech, Cambridge, Stanford e Berkeley nem sempre seja esse o caso. Outro aspeto notável é que na maioria das HEI o pico do número médio de retweets e favoritos ocorre na segunda metade do ano letivo, enquanto que Cambridge e Chicago figuram como exceções. A semelhança interessante final é notada em Harvard, Stanford, UC Berkeley e Oxford, em que tendo em conta o número de retweets, podemos afirmar que estas HEI experienciaram eventos muito semelhantes, caracterizados por dois cumes e dois vales, embora não necessariamente com a mesma intensidade e nem sempre nos mesmos exatos períodos.

Após obter esta caracterização do comportamento mensal de publicação e das respetivas métricas de envolvimento, procuramos também obter os padrões comportamentais relacionados com os horários preferenciais de publicação. Isto passa por obter os *heatmaps* seguintes para cada instituição onde diferentes unidades temporais (horas, dias da semana e meses) estão representados nos seus eixos. O *heatmap* da distribuição de publicações por dia da semana em relação à hora do dia, representado na figura 5.7 revelou-se dos mais úteis na recolha de informação sobre as HEI, já que muitas delas partilham os mesmos padrões.

Na figura conseguimos observar horários de publicação mais rígidos (como é o caso de MIT ou Johns Hopkins) assim como horários mais flexíveis (como Harvard e Stanford). Apesar dessas diferenças, existe uma característica comum a todas as instituições, a existência de um período prolongado do dia sem nenhum tipo de publicações, independentemente do dia da semana. A altura e duração desse período depende da instituição em questão. Esses períodos ocorrem normalmente entre o fim do dia e a madrugada, existindo universidades com períodos tão extensos

como da meia-noite ao meio-dia (MIT) e universidades com períodos tão curtos como das 4 às 10 da manhã (Harvard). Uma outra característica comum a grande parte das instituições (com a Universidade de Harvard como exceção) é a diminuição significativa de publicações aos fins de semana, com algumas delas quase cessando por completo a atividade (como é o caso de Zurich e Caltech), e outras limitando o grosso da sua atividade a um intervalo de tempo reduzido (MIT e Imperial College).

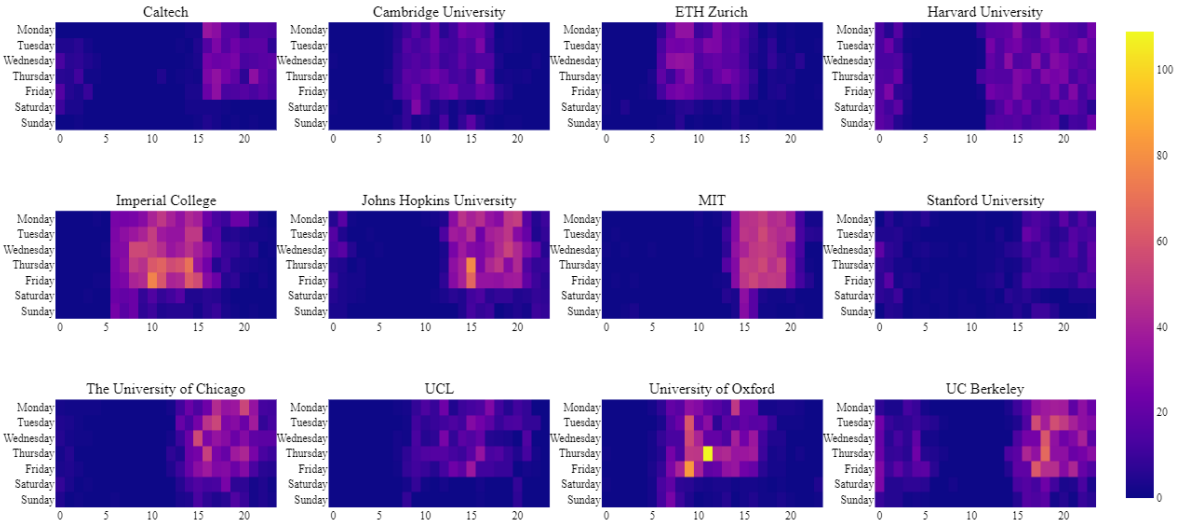

Figura 5.7: Comportamentos de publicação de acordo com o dia da semana e a hora do dia

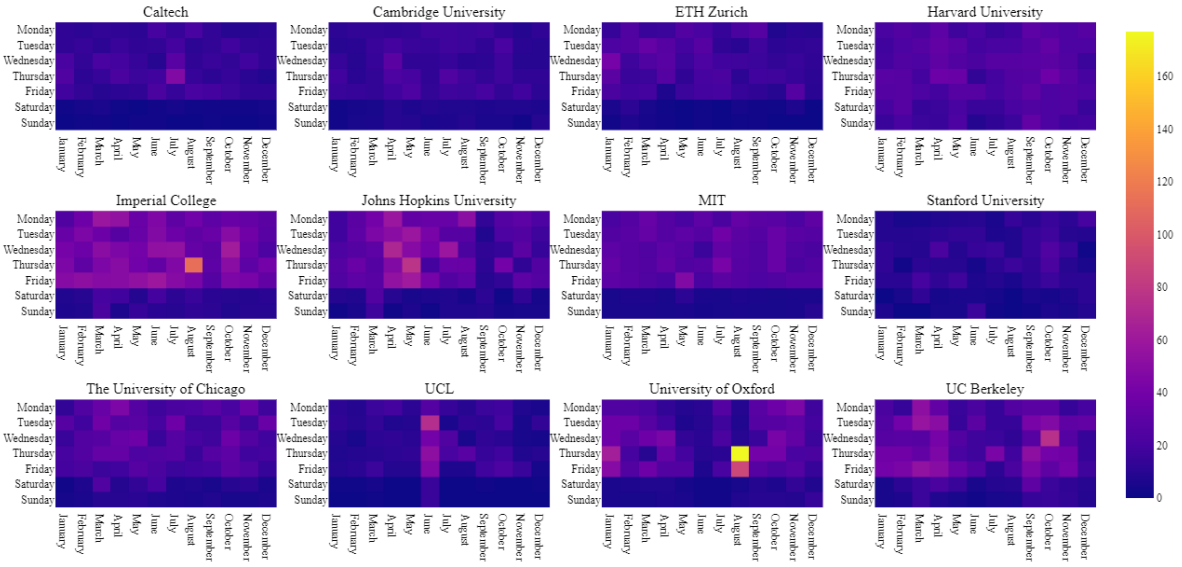

Figura 5.8: Comportamentos de publicação de acordo com o dia da semana e o mês do ano

A interpretação do *heatmap* da distribuição de publicações por dia da semana em relação ao mês do ano, figura 5.8, não introduz informação relevante nova, para além de reafirmar a existência de um período de publicação diminuída ao fim de semana, como se pode observar novamente na imagem. De um modo geral, não parece existir uma preferência do dia da semana a publicar, de acordo com o mês do ano, tirando exceções pontuais como Caltech e UCL em Julho e Imperial College e Oxford em Agosto.

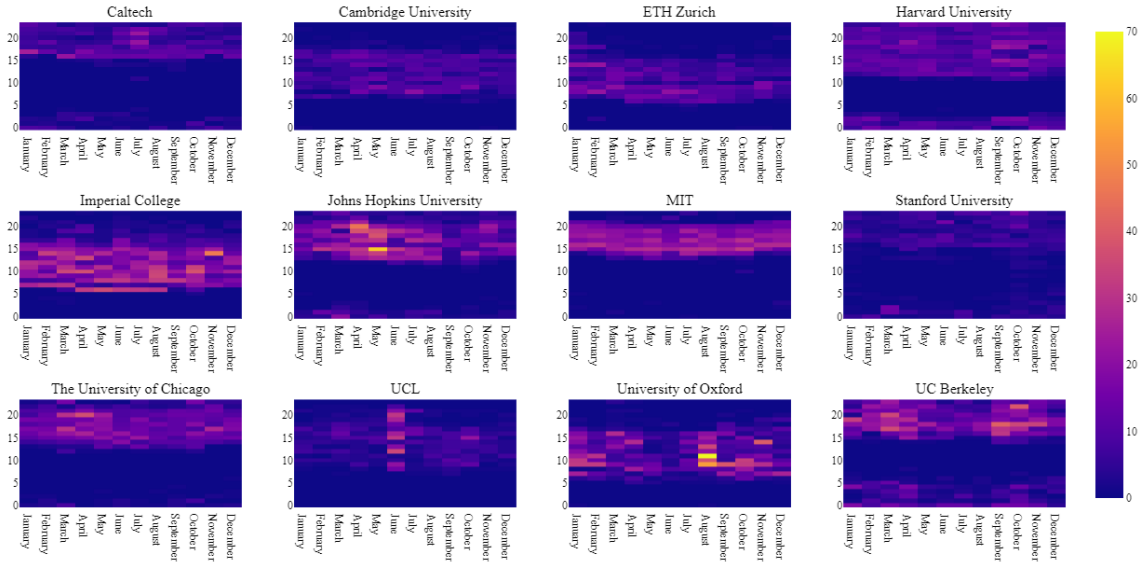

Figura 5.9: Comportamentos de publicação de acordo com a hora do dia e o mês do ano

Por fim, ao observar os *heatmaps* da distribuição de publicações por hora do dia em relação ao mês do ano da figura 5.9 conseguimos observar mais um padrão, apesar de este não ser comum a todas as instituições analisadas. Algumas instituições (Caltech, Zurich, Harvard, Imperial College, UChicago e MIT) apresentam uma ligeira curva de atividade, onde nos meses nas extremidades do ano, as instituições começam o seu período de atividade mais tarde e/ou terminam-no mais tarde. O exemplo mais claro desta realidade é o do MIT, onde em Janeiro, Fevereiro, Novembro e Dezembro, o horário de publicação atrasa cerca de 1 hora (das 15h às 21h) em comparação com os restantes meses do ano (das 14h às 20h).

5.3 Identificação de Tópicos

Como explicamos no capítulo 4, realizamos duas experiências diferentes na criação dos modelos dos tópicos latentes, na primeira utilizamos técnicas de *machine learning* não supervisionadas e não impusemos nenhum modelo editorial pré-definido enquanto que na segunda, utilizamos conhecimento de domínio para determinar as áreas editoriais. Nas secções seguintes vamos explicar de que modo essas experiências foram realizadas e explorar a sua implementação.

5.3.1 Experiência 1

Nesta experiência tentamos obter o melhor modelo a partir de um conjunto de modelos editoriais gerados utilizando os dois algoritmos com melhor desempenho e mais adequados para o nosso tipo de dados (de acordo com [14]): LDA e NMF.

Inicialmente, com o auxílio da biblioteca SpaCy, os campos de textos dos tweets recolhidos são sujeitos aos procedimentos de pré-processamento descritos na secção 'Identificação de Tópicos' do capítulo 4. De seguida, utilizando o *script* seguinte foi possível realizar a otimização de hiperparâmetros dos respetivos algoritmos, nomeadamente o número de tópicos a utilizar no processo de identificação. Isto é conseguido através do seu funcionamento iterativo, onde para cada valor de número de tópicos k é treinado um modelo com esse exato número de tópicos e é avaliada a sua coerência de acordo com a pipeline definida em [76]. Nesta fase, utilizamos a biblioteca Gensim[2], uma das bibliotecas especializadas na aprendizagem de modelos de tópicos mais populares, com mais de 2600 citações académicas [7].

```
import gensim.models.nmf as nmf

"""Create Dictionaries"""
id2word = corpora.Dictionary(data_lemmatized)

"""Create Corpus"""
texts = data_lemmatized

"""Term Document Frequency"""
corpus = [id2word.doc2bow(text) for text in texts]

def compute_coherence_values(dictionary, corpus, texts, limit, start=2, step=1):
    """
    Compute c_v coherence for various number of topics

    Parameters:
    ----------
    dictionary : Gensim dictionary
    corpus : Gensim corpus
    texts : List of input texts
    limit : Max num of topics

    Returns:
    -------
    model_list : List of topic models
    coherence_values : Coherence values corresponding to the model with
        respective number of topics
    """
    coherence_values = []
    model_list = []
    for num_topics in range(start, limit, step):
```

[2] https://radimrehurek.com/gensim

```
        model = nmf.Nmf(corpus=corpus, num_topics=num_topics,
            id2word=dictionary, w_max_iter=3000, random_state=seed)
        model_list.append(model)
        coherencemodel = CoherenceModel(model=model, texts=texts,
            dictionary=dictionary, coherence='c_v')
        coherence_values.append(coherencemodel.get_coherence())

    return model_list, coherence_values
```

Bloco de Código 5.3: Treino de modelos com diferentes valores de k (NMF)

Após a sua execução, obtemos um conjunto de modelos com diferentes números de tópicos e os seus respetivos *scores* de coerência. Para selecionar o melhor dos modelos resultantes, utilizamos os valores de coerência obtidos assim como a interpretabilidade dos tópicos gerados. No que diz respeito à coerência, para facilitar a interpretação, criamos um gráfico de linhas com os *scores* para os modelos gerados pelos respetivos algoritmos:

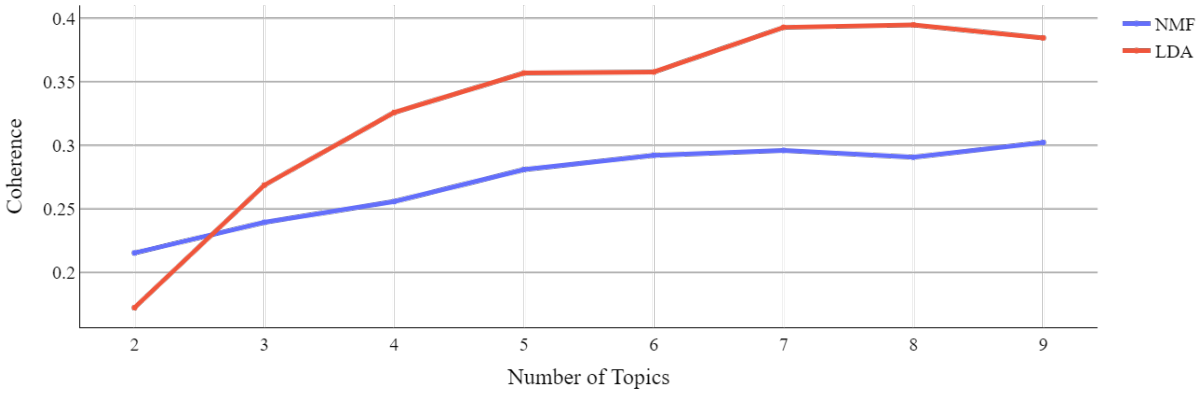

Figura 5.10: Relação da coerência de acordo com o número de tópicos

Como o valor de coerência tende a aumentar quanto maior o número de tópicos utilizados (o que pode levar a *overfitting*), limitamos o número de tópicos a 10 e consideramos, para o número ideal de tópicos, o valor mais elevado antes do crescimento começar a aplanar, ou refletir uma diminuição significativa. De acordo com estes critérios, na interpretação da figura anterior, podemos concluir que os modelos de 5 tópicos são os mais adequados para a nossa análise, tanto para o algoritmo NMF como para o LDA, possuindo valores de coerência de 0.28 e 0.36 respetivamente. Partindo desse princípio, no que diz respeito à interpretabilidade, tentamos observar os termos mais relevantes em cada um dos 5 tópicos gerados para ambos os modelos selecionados anteriormente.

Assumindo que ϕ_{kw} representa a probabilidade de um termo w $\in$ 1, ..., V para um tópico k $\in$ 1, ..., K, onde V denota o número de termos no vocabulário, a relevância de um termo w no tópico k é dada por

$$relevância(w, k) = log(\phi_{kw}) \tag{5.1}$$

que resulta na classificação dos termos por ordem decrescente da sua probabilidade específica para o tópico [81]. Sendo assim, obtivemos os 10 termos mais relevantes para cada tópico sob a forma dos seguintes *word clouds* e avaliamos a sua interpretabilidade:

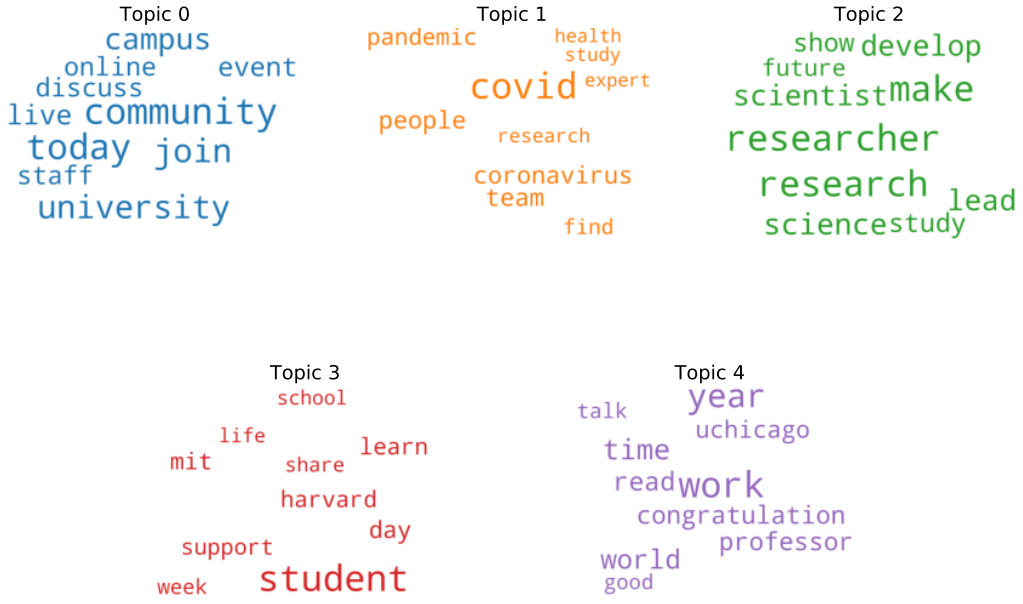

Figura 5.11: *Word clouds* dos tópicos LDA

Figura 5.12: *Word clouds* dos tópicos NMF

Como é possível de observar, o modelo de LDA, para além de possuir um maior valor de coerência como vimos anteriormente, parece ser também mais fácil e de interpretar relativamente ao modelo NMF. Isto parece dever-se em parte à distância entre tópicos dos dois modelos. Podemos ver que é bastante mais comum nos tópicos do modelo NMF a partilha de palavras relevantes entre os diferentes tópicos (como é o caso de 'research', 'comunity', 'researcher', 'help', etc). Isto diminui a individualidade de cada tópico e torna difícil sua a interpretação. Os tópicos do NMF não aparentam possuir temas agregadores identificáveis e não coincidentes.

Em contrapartida, analisando as *word clouds* dos tópicos gerados pelo modelo LDA, conseguimos, de um modo geral, interpretar os tópicos de acordo com o contexto em que estão inseridos. O tópico 0 parece incidir sobre eventos universitários: palavras como 'university', 'join', 'online', 'live', 'today', 'discuss' e 'event' enfatizam essa ideia. Os tópicos 1 e 2 estão claramente relacionados com a saúde pública/Covid-19 e investigação, respetivamente. O tópico 3 aparenta relacionar-se com a educação, evidenciado por palavras como: 'student', 'learn', 'school', 'support'. Finalmente, o Tópico 4, parece ser o mais difícil de categorizar, mas observando não só as 10 palavras mais relevantes, mas sim as 30 palavras mais relevantes, podemos obter uma imagem mais clara. Acreditamos que este tópico diz respeito à imagem pública devido a palavras como 'congratulation', 'work', 'good', 'great', 'woman', 'celebrate', 'hope', 'world' e 'history'.

Sendo assim, decidimos selecionar o modelo LDA de 5 tópicos como o modelo de melhor qualidade gerado pelas técnicas de aprendizagem não-supervisionada. Tendo os tópicos 0 a 4 sido devidamente identificados, podemo-nos referir a eles daqui em diante como Eventos, Saúde, Investigação, Educação e Imagem, respetivamente.

Após a obtenção do modelo editorial a utilizar, prosseguimos para a classificação de cada uma das publicações de acordo com o seu tópico dominante. Este processo de associação de um tópico a uma publicação é muito simples: visto que o modelo LDA representa os documentos sob a forma de uma soma pesada dos tópicos e as suas respetivas probabilidades, apenas necessitamos de recolher o tópico de maior probabilidade e esse passará a ser referido como o tópico dominante da publicação em questão:

```
def format_topics_sentences(ldamodel, corpus, texts, df):
    """
    Assing dominant topic to corresponding tweet

    Parameters:
    ----------
    ldamodel : Gensim LDA model
    corpus : Gensim corpus
    texts : List of input texts
    df: Pandas Dataframe with tweet data

    Returns:
    -------
    topics_df : Pandas Dataframe with the dominant topic assignment for every
        document in the given corpus
```

```
    """

    sent_topics_df = pd.DataFrame()
    """ Get main topic in each document """
    for i, row in enumerate(ldamodel[corpus]):
        row = sorted(row, key=lambda x: (x[1]), reverse=True)
        dominant = -1
        dominant_perc = -1
        for j, (topic_num, prop_topic) in enumerate(row):
            if j == 0:  """dominant topic """
                dominant = topic_num
                dominant_perc = prop_topic
                break

        topics_df = topics_df.append(pd.Series([int(dominant),
            round(dominant_perc,4)]), ignore_index=True)

    topics_df.columns = ['topic','topic_perc']

    """Append original text to the end of the output"""
    contents = pd.Series(texts)
    topics_df = pd.concat([topics_df, contents], axis=1)
    topics_df = pd.merge(df, topics_df, left_index=True, right_index=True)
    return(topics_df)
```

Bloco de Código 5.4: Atribuição do tópico dominante (experiência 1)

Por fim, funcionando como uma amostra da atribuição anterior, recolhemos os tweets mais representativos de cada tópico para obter uma noção, ainda que vaga, da qualidade deste processo:

Tabela 5.4: Tweets mais representativos de cada tópico (experiência 1)

Tópico	Tweet
Eventos	police_car_light Due to the continued power outage on campus, the following Rec Sports facilities & programs will be closed and canceled on 10/11: - RSF - Stadium Fitness Center - Intramural Sports - Sports Clubs - Spieker Pool, Hearst Pool, and Strawberry Canyon Pool
Saúde	Cheap, daily, DIY tests can be as effective as a vaccine at interrupting coronavirus transmission — and may the only viable option for a quick return to normal life, says epidemiologist and disease testing expert Michal Mina.
Investigação	The origin of strong magnetic fields in massive stars has been solved sparkles Astrophysicists from @OxfordPhysics @UniHeidelberg @maxplanckpress and @HITstudies have shown how strong magnetic fields can be formed in stellar mergers by developing a model with large computer simulations.
Educação	Our @osp_cp offers two free college access programs for high school students: Upward Bound and STEM Initiative. Students receive tutoring, take enrichment classes, and receive college admissions & financial aid application assistance. Apply by November 1: http://bit.ly/2N38hmq.
Imagem	Shout out to our 2020 alumni #Oscar nominees: best original song composer Joshuah Brian Campbell AB'16 for "Stand Up"from "Harriet"and best picture producer David Heyman AB'83 for "Marriage Story"and "Once Upon a Time...in Hollywood"!

5.3.2 Experiência 2

Nesta experiência estabelecemos um modelo editorial com base em conhecimento de domínio, segundo o qual iremos classificar as publicações recolhidas. Como mencionamos no capítulo 4, baseamo-nos nos indicadores segundo os quais o CWUR classifica as melhores universidades a nível mundial. A CWUR utiliza sete indicadores objetivos e robustos agrupados em quatro áreas para classificar as universidades do mundo:

1. Qualidade da Educação, medida pelo número de ex-alunos de uma universidade que ganharam grandes distinções académicas em relação à dimensão da universidade (25%)
2. Empregabilidade de ex-alunos, medido pelo número de ex-alunos de uma universidade que ocuparam cargos executivos de topo nas maiores empresas do mundo em relação à dimensão da universidade (25%)
3. Qualidade da Universidade, medida pelo número de membros do corpo docente que ganharam as principais distinções académicas (10%)
4. Desempenho a nível de investigação:
    - Resultados da investigação, medidos pelo número total de trabalhos de investigação (10%)
    - Publicações de Alta Qualidade, medida pelo número de artigos de investigação que aparecem em revistas conceituadas (10%)
    - Influência, medida pelo número de artigos de investigação que aparecem em revistas altamente influentes (10%)
    - Citações, medidas pelo número de trabalhos de investigação com um elevado número de citações (10%)

Sendo estes os critérios de avaliação utilizados, seria natural que as publicações no Twitter das instituições no topo do ranking selecionadas incidissem nesses temas. Instituições que conseguem atingir o topo do ranking possuem boa qualidade de educação, empregabilidade de ex-alunos, qualidade institucional/académica e investigação. Como tal, é de esperar as suas interações nas redes sociais com o seus públicos-alvo sejam dominadas pelos tópicos que destacam as suas mais valias.

Para além destes 4 indicadores, através duma análise manual do conteúdo das publicações das instituições selecionadas, e tendo também em conta o contexto atual pandémico no qual estamos inseridos, conseguimos identificar mais dois tópicos com presença significativa nas publicações das instituições, sendo eles o tópico da saúde (maioritariamente publicações relacionadas com saúde pública e a pandemia) e o tópico de sociedade/comunidade (publicações de cariz social e relacionadas direitos humanos e com as comunidades locais).

Com estes 6 tópicos obtidos, definimos como os seus centroides as palavras que acreditamos que melhor os representam, sendo elas: 'education', 'employment', 'faculty', 'research', 'health'

e 'society'. De seguida, vamo-nos servir de modelos de *embedding* de palavras pré treinados para as representar, assim como todas as restantes palavras presentes no conjunto dos tweets. Inicialmente, na determinação no modelo de *embeddings* a utilizar, o modelo Glove de 200 dimensões, pré-treinado em 2B de tweets e com 1.2M de dimensão de vocabulário [8] foi a nossa primeira opção. Contudo, este modelo não coloca restrições no idioma das palavras utilizadas para treino, sendo que grande parte do vocabulário aprendido está em outros idiomas e não possui nenhuma utilidade no nosso contexto. Talvez graças a este aspeto, observamos que bastantes termos constituintes do nosso corpus não fazem parte do vocabulário desse modelo, ou seja, não possuem o seu respetivo vetor, o que significa que teríamos de os remover da nossa análise e possivelmente perder informação valiosa.

Sendo assim, decidimos antes utilizar um modelo Word2Vec de 300 dimensões, pré-treinado em dados de notícias do GoogleNews (cerca de 100B de palavras) e com 3M de dimensão de vocabulário completamente em inglês [9], porque para além de possuir um número bastante superior de palavras representadas, possui também *embeddings* de maiores dimensões. O número de dimensões das *embeddings* determinam o grau de compressão intencional da informação lexical, sendo que uma maior dimensionalidade permite ao modelo distinguir um maior número detalhes lexicais dos dados, se estes os possuírem.

Com o modelo de *embeddings* de palavras determinado, começamos por fazer os mesmos processos de limpeza de texto e pré-processamento mencionados na experiência 1, assim como a remoção dos termos utilizados pelas instituições nas suas publicações que não tenham uma representação vetorial (*embedding*) no modelo word2vec pré-treinado. De seguida, para cada palavra, calculamos a sua similaridade a cada um dos centroides previamente definidos, através da semelhança cosseno implementada por um módulo da biblioteca Gensim, de fórmula

$$similaridade(A, B) = \frac{A \cdot B}{||A|| \times ||B||} \tag{5.2}$$

onde A e B são as representações vetoriais (*embeddings*) das palavras envolvidas

De seguida, obtivemos os valores TF-IDF para cada palavra em cada documento com o auxílio da biblioteca scikit-learn[3]. O valor de TF-IDF (*term frequency-inverse document frequency*) é uma medida estatística capaz de avaliar a relevância de uma palavra para um documento constituinte de uma coleção de documentos. O objetivo de utilizar TF-IDF em vez das frequências brutas de ocorrência de um termo num dado documento é reduzir o impacto dos termos que ocorrem muito frequentemente num dado corpus e que são, portanto, empiricamente menos informativos do que os termos que ocorrem numa pequena fração do corpus. O valor é calculado pela biblioteca utilizada, multiplicando duas métricas: quantas vezes uma palavra aparece num documento, e a frequência inversa da palavra em todo um conjunto de documentos. O valor TF-IDF para um termo t presente num documento d, contido num conjunto de documentos D é calculado da seguinte forma:

[3]https://scikit-learn.org/

$$tfidf(t,d,D) = tf(t,d) \times idf(t,D) \tag{5.3}$$

onde

$$tf(t,d) = f_{t,d} \tag{5.4}$$

$$idf(t,D) = log\left(\frac{(1+|D|)}{(1+df(t))}\right) + 1 \tag{5.5}$$

sendo que $f_{t,d}$ representa o número de vezes que o termo t ocorre no documento d e $df(t)$ representa a frequência de documentos de t, ou seja, o número de documentos do conjunto D que contêm o termo t. Tendo determinado a similaridade de cada termo a cada centroide dos tópicos identificados, assim como os valores de TF-IDF dos termos por documento, fazemos a atribuição de um tópico dominante a cada publicação de acordo com o seguinte código:

```
topic_perc = []

def assign_topic(row):
    """
    Assing dominant topic to corresponding tweet

    Parameters:
    ----------
    row : Pandas Dataframe row

    Returns:
    -------
    topic: The dominant topic discovered for the current row
    """

    """get the non-zero tf-idf values for current row"""
    document = tf_idf_vector[row.name]
    df_temp = pd.DataFrame(document.T.todense(), index=cv.get_feature_names(),
        columns=['tfidf'])
    df_temp = df_temp.sort_values(by=["tfidf"],ascending=False)
    df_temp = df_temp[df_temp.tfidf != 0]
    df_temp.reset_index(level=0, inplace=True)
    values = df_temp.values.tolist()
    dic = {}
    for pair in values:
        key, value = pair[0], pair[1]
        dic[key] = value

    """calculate the internal score for each topic"""
    scores = [0,0,0,0,0,0]
    for word in row['text']:
        for i in range(len(centroids)):
```

```
            scores[i] = scores[i] + dic[word]*similarity[word][i]
    topic = scores.index(max(scores))
    topic_perc.append(max(scores))

    """assign the topic with the highest score to the current row"""
    return topic

df['topic'] = df.apply(assign_topic, axis=1)
df['topic_perc'] = topic_perc
```

Bloco de Código 5.5: Atribuição do tópico dominante (experiência 2)

Ao contrário da primeira experiência, não possuímos a representação de um documento como um conjunto de tópicos associados a uma probabilidade, mas sim um conjunto de tópicos associados a um *score*. Assumindo como exemplo a frase 'Sample phrase', a palavra 'Sample' possui 6 *scores* diferentes, cada um relativo à sua pertença aos 6 tópicos diferentes. De igual modo, a palavra 'phrase' possui também 6 *scores* para cada tópico. No final, o *score* de cada um dos 6 tópicos resulta da soma dos *scores* de cada palavra presente no texto desse documento (neste caso, 'sample' e 'phrase'), para o tópico em questão.

Um *score* de termo t, presente num documento d, contido num conjunto de documentos D, para um determinado tópico k é dado pela seguinte formula:

$$Score(t, d, D, k) = tfidf(t, d, D) * similaridade(t, centroide_k) \tag{5.6}$$

Sendo assim, o *score* de um tópico é obtido através de:

$$Score(k, d) = \sum_{i=1}^{i=|d|} Score(t_i, d, D, k) \tag{5.7}$$

Após o cálculo dos respetivos *scores* para cada tópico num determinado documento, o tópico com o maior *score* é atribuído a esse documento como o seu tópico dominante, ou seja:

$$TópicoDominante(d) = \max(Score(k_i, d)), k \in [0, 5] \tag{5.8}$$

Por fim, mais uma vez como uma amostra da atribuição anterior, recolhemos os tweets mais representativos de cada tópico para obter uma vaga noção da qualidade deste processo:

Tabela 5.5: Tweets mais representativos de cada tópico (experiência 2)

Tópico	Tweet
Educação	Dr Sandra Leaton-Gray works in UCL's Institute of Education (IOE) (@ioe_london). She teaches Sociology of Education and runs international education research projects. #UCLDisruptYourThinking –(2/8)
Emprego	We are committed to actualizing opportunities for all. We're proud to launch the Reimagining Pathways to Employment in the US Challenge with @ThinkMFF & @newprofit to accelerate future employment, especially for underserved communities. More:
Faculdade	Congratulations to five @UCL academics who have been made Fellows of the British Academy for humanities and social sciences: Prof Meg Russell, Prof Essi Viding, Prof Elaine Unterhalter, Prof Christopher Pinney and Prof Rick Rawlings party_pooper
Investigação	When it comes to microbiome research, pile_of_poo matters.Research at Caltech explores how the propensity of rodents (often used in microbiome research) to engage in coprophilia (aka, poop eating) may impact research outcomes.
Saúde	Health care systems rather look at diseases than the causes of health. The Life Science Zurich Business Nework conference THE CAUSE OF HEALTH showed what the future of personalized health could look like, and @ETH_en reported on this event:
Sociedade	The Departmental Society for Geology, the De La Beche Society, will be changing its name in line with the Society's values of equality and inclusivity

5.4 Análise de Sentimento

Inicialmente, para realizar a análise de sentimento, o processo de limpeza de texto necessitou de ser refeito com uma ligeira alteração. Como mencionámos no capítulo 3, e de acordo com o artigo [84], obtivemos as técnicas de pré-processamento recomendadas e também as não recomendadas para a análise de sentimento. Na limpeza de texto realizada durante a fase de identificação de tópicos, apenas consideramos as *POS Tags* de nomes, verbos, adjetivos e advérbios, fazendo com que os *tokens* de pontuação fossem removidos. Para a formulação de modelos de tópicos, onde apenas os termos que constituem o texto são relevantes, este processamento sem pontuação é ideal. No entanto, no que toca a análise de sentimento, a pontuação consegue também transmitir informação sobre o sentimento do autor da publicação em relação ao tópico a ser abordado e, sendo assim, a remoção de pontuação é uma a das técnicas não recomendadas pelo artigo mencionado que aplicamos na limpeza de texto anterior e tivemos de alterar para este novo contexto, acrescentando a *POS Tag* para a pontuação ao conjunto de *tags* permitidas.

Com o pré-processamento executado, passamos à atribuição do sentimento e polaridade, através de seguinte código:

```
from textblob import TextBlob

def getPolarity(row):
    """
    Assign polarity to corresponding tweet

    Parameters:
    ----------
```

```
    row : Pandas Dataframe row

    Returns:
    --------
    Float polarity value between -1 and 1
    """

    try:
        return TextBlob(row['text']).sentiment.polarity
    except:
      return 0

def getSentiment(row):
    """
    Assign Sentiment to corresponding tweet

    Parameters:
    -----------
    row : Pandas Dataframe row

    Returns:
    --------
    sentiment: Sentiment String
    """

    if row['polarity'] > 0.33:
        sentiment = 'Positive'
    elif row['polarity'] < -0.33:
        sentiment = 'Negative'
    else:
        sentiment = 'Neutral'
    return sentiment
```

Bloco de Código 5.6: Atribuição do sentimento e polaridade

Como podemos observar no bloco de código anterior, primeiramente obtemos a polaridade utilizando a biblioteca TextBlob, introduzida no capítulo 2. De seguida, com base nesses valores de polaridade, e através de valores limite, categorizamos esses valores em 3 categorias diferentes, 'Positive', 'Neutral' e 'Negative', correspondentes ao sentimento positivo, neutro e negativo, cada uma delas abrangendo os valores de]0.33,1], [0.33, -0.33] e]-0.33, -1] respetivamente. Estes limites foram escolhidos desta forma pois não só permitem intervalos iguais para as diferentes categorias, como garantem que termos com sentimento de baixa intensidade (menores que 1/3 de um sentimento completamente positivo ou negativo) são considerados como neutros, de modo tornar a análise mais fidedigna.

De forma semelhante ao que foi feito após as atribuições de tópicos dominantes na secção anterior, recolhemos também o tweet mais positivo, mais negativo e mais neutro:

Tabela 5.6: Tweets mais representativos de cada sentimento

Sentimento	Polaridade	Tweet
Positivo	1	Happy birthday, MIT! birthday_cake MIT's charter was signed April 10, 1861. http://mitsha.re/WsNx50zaW61
Neutro	0	The IEOR Department at UC Berkeley is now hiring one co-instructor for the Data-X spring semester course, as well as multiple graduate student instructors. Apply now: https://buff.ly/2Mto3YV
Negativo	-1	To avoid the worst impacts of climate change, the world's electric energy systems must stop producing carbon by 2050. "My work has shown me that we do have the means to tackle the problem, and we can start now," says NSE PhD student Nestor Sepulveda. http://mitsha.re/PZRn50xVYZn

5.5 Análise Preditiva

Para finalizar o desenvolvimento, testamos a capacidade preditiva dos modelos obtidos em duas tarefas diferentes, uma de classificação e outra de regressão. Para isso, realizamos os seguintes passos em duas cópias idênticas da nossa coleção de tweets das HEI, onde apenas o modelo editorial utilizado para a atribuição do tópico dominante difere, sendo que uma das atribuições resulta da experiência 1 enquanto que a outra da experiência 2. A tarefa de regressão consiste no treino supervisionado de um modelo de *machine learning* capaz de prever o número de retweets de uma publicação, visto que este valor serve como um bom indicador do sucesso obtido no alcance dos seus públicos-alvo. A tarefa de classificação consiste também no treino supervisionado de um modelo, capaz de prever o tópico dominante de uma publicação por parte de uma HEI, com base nos tópicos das publicações anteriores, numa janela temporal de 7 dias.

5.5.1 Previsão do Número de Retweets

Na tarefa de regressão, começamos por definir os atributos que poderiam possuir utilidade para o treino deste modelo, sendo que também consideramos atributos resultantes de *feature engeneering*. Como o objeto Tweet e o objeto User possuem bastantes atributos de metadados sem utilidade para este tipo de análise, utilizamos apenas aqueles através dos quais conseguimos retirar alguma informação considerada por nós relevante.

Sendo assim, obtivemos a partir do atributo 'created_at' os atributos 'weekday', 'hour', 'month' (que correspondem ao dia da semana, hora e mês da publicação, respetivamente) e a partir do atributo 'retweeted_status.text', o atributo 'is_retweet_status' (que indica se a publicação é um retweet). No que diz respeito ao conteúdo, obtivemos para cada publicação a probabilidade associada a cada uma das áreas editoriais/tópicos que o constituem. Dependendo da experiência utilizada na obtenção do modelo editorial, a quantidade destes novos atributos será diferente, uma vez que o modelo editorial da experiência 1 irá resultar em 5 novos atributos (topic_0, topic_1, ... , topic_4) enquanto que o modelo da experiência 2 resulta na criação de 6 novos atributos (topic_0, topic_1, ... , topic_5). Dos atributos originais do dataset, assim como

daqueles criados num passo anterior de desenvolvimento, selecionamos o 'user.followers_count', 'polarity', 'is_quote_status', 'dominant_topic' e 'topic_perc'.

De seguida, com o auxílio da biblioteca scikit- learn, dividimos os conjuntos de dados em conjuntos de treino e conjuntos de teste, com um *split* de 80%/20% e normalizamos os atributos anteriores. Após a sua execução, todos os atributos serão ajustados ao intervalo [0,1], o que significa que o valor mínimo e máximo de um atributo será 0 e 1, respetivamente.

$$x_{ajustado} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{5.9}$$

A ideia principal por detrás da normalização/padronização é bastante simples. As variáveis que são medidas em escalas diferentes não contribuem igualmente para a adaptação do modelo e podem acabar por criar *biases*. Assim, para lidar com este potencial problema, a normalização MinMax é normalmente utilizada antes do treino do modelo.

Com os conjuntos de dados de treino criados, realizamos *univariate feature selection* para reduzir a dimensionalidade dos dados e obter apenas os atributos que promovem melhores resultados nesta tarefa. A *univariate feature selection* funciona através da seleção dos melhores atributos com base em testes estatísticos univariados. Pode ser visto como uma etapa de pré-processamento para um estimador.

Para tal, é necessária uma função de pontuação, como a informação mútua, que dada uma grande quantidade de amostras, é capaz de capturar qualquer tipo de dependência estatística entre duas variáveis. A informação mútua é calculada entre duas variáveis e mede a redução da incerteza para uma variável dado um valor conhecido da outra variável. Com base na implementação de informação mútua para *feature selection* da biblioteca scikit-learn, obtivemos os 10 atributos mais importantes, com *scores* de informação mútua de acordo com os gráficos de barras presentes nas figuras seguinte.

Como podemos verificar através das figuras, para ambas as experiências, o número de seguidores de uma instituição foi considerado o atributo de maior importância, ou seja, o atributo com a maior quantidade de informação mútua em relação à variável alvo 'retweet_count', o que vai de acordo com o que foi observado durante a análise exploratória: que parece existir algum tipo de relação proporcional entre o número de seguidores de uma instituição e o número de retweets das suas publicações.

Para além disso, conseguimos também verificar que os atributos criados com base nas distribuições de tópicos (topic_0 a topic_n) foram todos eles considerados dos mais importantes na determinação do número de retweets. Observando os *scores* de cada um dos atributos é possível verificar também que, excetuando o número de seguidores do utilizador, os restantes atributos possuem muito menos informação mútua do que inicialmente espectávamos, podendo este aspeto comprometer a qualidade dos resultados. Sendo assim levanta-se a questão de se a informação fornecida em relação ao conteúdo é realmente suficiente para obter uma previsão de qualidade do número de retweets de uma publicação.

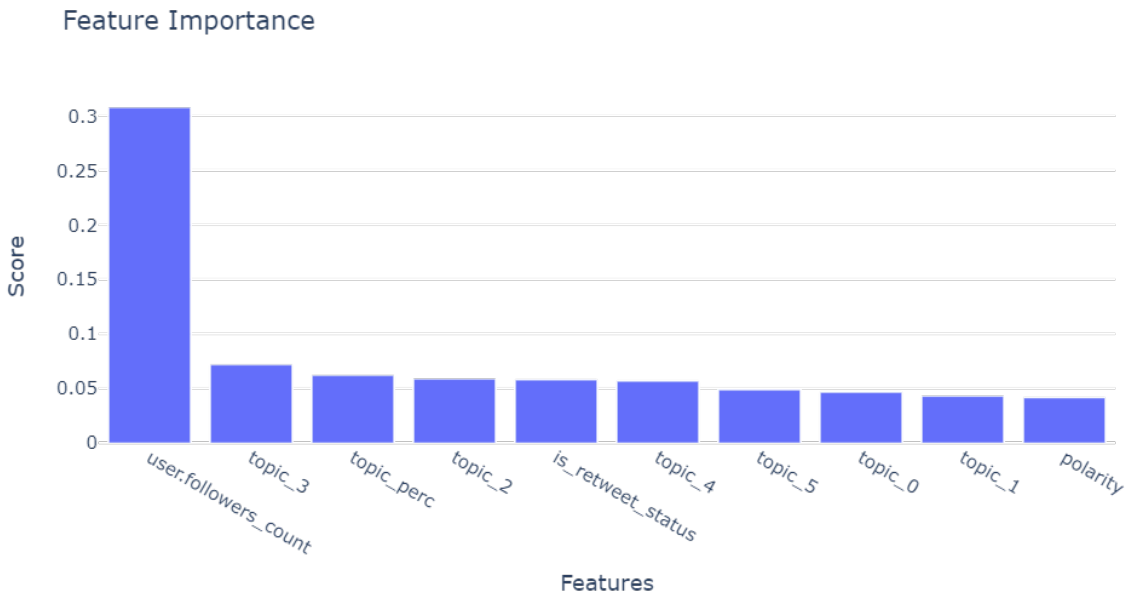

Figura 5.13: Importância de atributos (experiência 1)

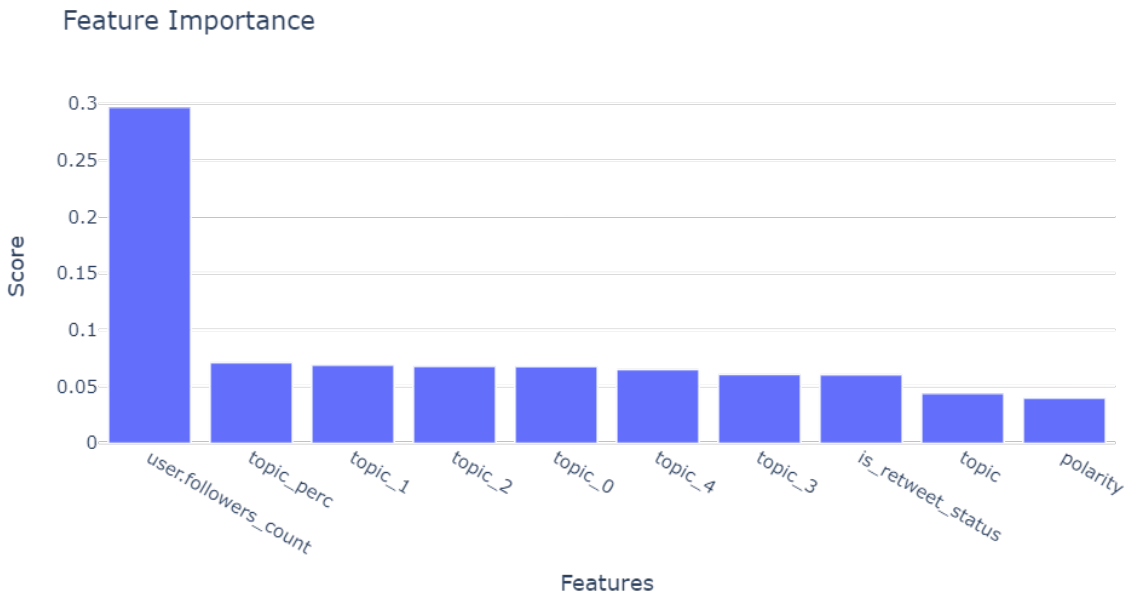

Figura 5.14: Importância de atributos (experiência 2)

Após os passos anteriores, prosseguimos para o treino de 10 diferentes modelos nos dados de treino especificados e avaliamos o seu desempenho na tarefa em questão. Os métodos utilizados na obtenção dos modelos foram 3 variantes de regressão linear (linear, ridge e lasso), árvores de decisão, 3 métodos de combinação de árvores de decisão (bagging, boosting, random forest), k-nearest neighbors, máquinas de vetores de suporte, e perceptrons multicamadas.

Numa grande parte dos projetos de *machine learning*, os diferentes modelos são treinados no

conjunto de dados e é selecionado o modelo que apresenta o melhor desempenho. No entanto, há sempre espaço para melhorias, pois não podemos dizer com certeza absoluta que um modelo em particular seja o melhor para o problema em questão. Por conseguinte, o nosso objetivo é melhorar os modelos de qualquer forma possível. Um fator importante no desempenho destes modelos são os seus hiperparâmetros, uma vez estabelecidos os valores apropriados para estes hiperparâmetros, o desempenho de um modelo pode melhorar significativamente. De modo a encontrar valores ótimos ou perto de ótimos para os hiperparâmetros dos modelos, utilizamos uma *grid search* ou pesquisa/busca de grelha.

A *grid search* consiste no processo de efetuar a otimização de hiperparâmetros a fim de determinar os seus melhores valores para um determinado modelo, treinado num determinado conjunto de dados. Como mencionado anteriormente, o desempenho de um modelo depende significativamente do valor dos hiperparâmetros. Sendo que não há forma de conhecer antecipadamente os melhores valores para hiperparâmetros, idealmente, precisamos de testar o espaço de todos os valores possíveis para conhecer os valores ótimos. Fazer isto manualmente poderia levar um tempo e recursos consideráveis, pelo que utilizamos a GridSearchCV para automatizar este processo.

A GridSearchCV é uma função inserida no pacote de *model_selection* do scikit-learn. Esta função ajuda a percorrer hiperparâmetros pré-definidos e a treinar o estimador (modelo) no seu conjunto de dados. Assim, no final, podemos selecionar os melhores parâmetros a partir dos hiperparâmetros listados. O bloco de código seguinte demonstra um exemplo da execução da GridSearchCV:

```
from sklearn.neighbors import KNeighborsRegressor

"""Hyperparameter values"""
criterion = ['mse', 'friedman_mse', 'mae', 'poisson']
splitter= ['best', 'random']
max_depth = [int(x) for x in np.linspace(5, 50, 10)]
max_depth.append(None)
min_samples_split = [int(x) for x in np.linspace(2, 30, 15)]
max_features = ['auto', 'sqrt', 'log2']
min_impurity_decrease = [0.0, 0.05, 0.1]
ccp_alpha = [x/10 for x in  np.linspace(0, 9, 9)]

"""Create the grid"""
grid = {
    'criterion': criterion,
    'splitter': splitter,
    'max_depth': max_depth,
    'min_samples_split': min_samples_split,
    'max_features': max_features,
    'min_impurity_decrease' : min_impurity_decrease,
    'ccp_alpha' : ccp_alpha
}
```

```
"""Define the base learner"""
base = DecisionTreeRegressor()

"""Create the grid search Decision Tree"""
results = GridSearchCV(
    estimator = base,
    param_grid = grid,
    cv = KFold(n_splits=5, random_state=1, shuffle=True),
    scoring = 'neg_mean_squared_error',
    verbose = 1,
    """Parallel Running"""
    n_jobs = -1
)

"""Fit the grid search model"""
results.fit(x_train, y_train)

"""View the best parameters from the grid search"""
print('Config: %s' % results.best_params_)
```

Bloco de Código 5.7: Execução da GridSearchCV para uma Árvore de Decisão

Os valores pré-definidos para hiperparâmetros são fornecidos à função GridSearchCV através da definição de um dicionário no qual mencionamos um hiperparâmetro particular juntamente com os valores que este pode assumir. A GridSearchCV tenta todas as combinações dos valores passados no dicionário e avalia o modelo para cada combinação usando o método de *cross-validation*. Assim, após a utilização desta função, obtemos o ganho/perda para cada combinação de hiperparâmetros e podemos escolher a que apresentar o melhor desempenho. A métrica utilizada na avaliação dos modelos é também um dado necessário a fornecer à função, sendo que no nosso caso utilizamos o RMSE.

Este tipo de busca exaustiva, mesmo quando automatizada, pode na mesma resultar em elevados custos temporais em casos de conjuntos de dados com um elevado número de dimensões e, principalmente, em casos de modelos com um elevado número de hiperparâmetros. É possível mitigar o problema de longos tempos de execução através da execução concorrente. O parâmetro 'n_jobs' da GridSearchCV permite especificar o número de Cores do CPU a utilizar durante o *loop* de *cross-validation* da pesquisa. A atribuição do valor - 1 a este parâmetro permite a utilização de todos os processadores. No nosso caso, servimo-nos de máquinas virtuais com recursos alocados pela Google [4], de modo a tirar o melhor proveito desta medida de mitigação. Conseguimos efetuar o processo de *grid search* utilizando 80 *workers* concorrentes e, deste modo, reduzir o custo temporal associado a este processo a um valor tolerável, sendo que o modelo mais dispendioso de otimizar apenas demorou 205.2 minutos a finalizar (MLP).

[4]https://colab.research.google.com/notebooks/intro.ipynb

5.5.2 Previsão do Tópico Dominante de uma Publicação

Na tarefa de classificação, começamos também por *feature engineering*. Num contexto real, para obter o tópico dominante de uma publicação, pouca da informação que possuímos se revela útil, sendo que grande parte de dados não conseguimos obter em tempo real. Imaginemos o exemplo seguinte: Uma HEI pretende determinar o tópico dominante das próximas publicações de uma outra HEI adversária, numa tentativa de identificar a sua estratégia editorial e adotá-la, visto que esta tem revelado bons resultados. A informação que está disponível a esta HEI não é a mesma que à qual nós conseguimos ter acesso no nosso dataset de publicações passadas. Campos como o conteúdo da publicação, a percentagem do tópico dominante, a polaridade, o número de retweets e o número de favoritos podem parecer excelentes atributos para utilizar nessa previsão, mas são impossíveis de obter, já que a publicação ainda não aconteceu. Como tal, a melhor fonte de informação disponível a esta HEI é o passado, as publicações que já ocorreram, cujos atributos que demonstramos interesse em utilizar anteriormente já se encontram disponíveis. Ao obter informação sobre métricas de publicações anteriores a HEI torna possível a caracterização da publicação a acontecer com base nelas.

Sendo assim, definimos como objetivo a previsão do tópico dominante de uma publicação, dada informação da mesma instituição nos 7 dias anteriores. Uma das informações mais importantes a obter dessa janela temporal são os tópicos dominantes, com a possibilidade de encontrar alguma regularidade na sua ocorrência. Como tal, necessitamos encontrar uma boa representação para caracterizar o tópico dominante de cada um desses dias.

Por exemplo, assumindo que no dia 2 de Setembro de 2019 a universidade Caltech realizou apenas 1 publicação dominada pela área editorial de educação. Neste caso, a caracterização do tópico dominante nesse dia, para a universidade de Caltech é simples: o tópico dominante desse dia corresponde ao tópico dominante da sua única publicação. Assumindo agora 3 publicações diferentes, duas delas dominadas pela área da investigação e a última pela área da educação. Necessitamos de uma forma de agregar os tópicos dominantes desse dia num só valor, visto que existem três publicações e dois tópicos dominantes diferentes.

O método de agregação selecionado consiste na moda de todos os valores de tópico dominante existentes no mesmo dia, para a mesma instituição. A sua implementação é descrita no bloco de código seguinte.

```
"""Group databy calendar day and aggregate via list"""
temp = df.set_index('created_at').groupby([pd.Grouper(freq='D'),'user.name'])
['topic'].apply(list).reset_index()

def compute_mode(numbers):
    """
    Calculate the mode given a list of numbers

    Parameters:
    ----------
    numbers : A list of integer numbers

    Returns:
    -------
    lista : A list consisting in the most ocorring values in the input
    """
    lista = []
    counts = {}
    maxcount = 0
    for number in numbers:
        if number not in counts:
            counts[number] = 0
        counts[number] += 1
        if counts[number] > maxcount:
            maxcount = counts[number]

    for number, count in counts.items():
        if count == maxcount:
            lista.append(str(number))

    return lista


def assign_mode(row):
    """
    Assing the mode to a publication

    Parameters:
    ----------
    row : Pandas Dataframe row

    Returns:
    -------
    An integer value representing the assigned mode
    """
    lista = compute_mode(row['topic'])
    if len(lista) != 1:
        return -1
    else:
        return lista[0]
```

```
"""Assign mode to grouped dataframe"""
temp['topic'] = temp.apply(assign_mode, axis=1)

"""Colect average values of publication metrics for grouped dataframe"""
temp2 = df.set_index('created_at').groupby([pd.Grouper(freq='D'),'user.name'])
['topic_perc','favorite_count','retweet_count','polarity'].mean().reset_index()

"""Merge the two dataframes"""
df = pd.merge(temp, temp2, on=['created_at','user.name'], how='outer')

"""Remove rows with more than one mode"""
df = df_new[df_new.topic != -1]
```

Bloco de Código 5.8: Agregação das publicações por Instituição/Dia

Como é possível observar no código anterior, começamos por agregar as publicações numa nova *dataframe* estruturada por dia e instituição. Com esta operação, o valor do atributo 'topic' passou a uma lista dos tópicos dominantes observados no dia correspondente. De seguida, através das funções definidas, 'assign_mode' e 'compute_mode', atribuímos a cada uma das observações da *dataframe* anterior a sua respetiva moda.

Se notarmos a implementação da função 'assign_mode' podemos verificar que, quando a observação em questão possui mais do que um valor mais frequente, optamos por atribuir-lhe o valor -1 de moda, sinalizando aquela observação para ser eliminada num passo posterior.

Inicialmente, ponderamos uma implementação diferente para este tipo de problema, onde dado um conjunto de valores, todos eles representando a moda do tópico dominante para o dia e instituição em questão, faríamos uma atribuição aleatória de um dos valores como o tópico dominante desse dia. Contudo, decidimos descartar essa abordagem, com receio de resultar numa caracterização pouco fidedigna da realidade. Demos preferência à remoção do dia por completo à instituição por duas razões distintas: primeiramente, o número de casos onde esta situação ocorre é muito reduzido, sendo que não se perde muita informação com essa remoção e, em segundo lugar, acreditamos que esta abordagem resulta num *dataframe* mais fiel à realidade. Se num determinado dia, para uma determinada HEI, os tópicos dominantes mais frequentes forem mais do que um, então esse dia não foi verdadeiramente dominado por um único tópico, e, como tal, acreditamos estar mais correto classificá-lo como um dia 'neutro' e remover a observação do *dataframe* do que classificá-lo aleatoriamente a um dos tópicos mais frequentes e manufaturar dados que não existem.

Após a atribuição da moda, recolhemos também as médias do número de retweets, favoritos, percentagem do tópico dominante e da polaridade, agregadas por Dia e Instituição, e procedemos à combinação destes atributos com o *dataframe* obtido anteriormente. Este processo descrito resulta num *dataframe* temporário representado na tabela seguinte.

Tabela 5.7: Cinco primeiras linhas do *dataframe* temporário obtido

created_at	user.name	topic	topic_perc	favorite_count	retweet_count	polarity
2019-09-01	Cambridge University	2	0.50	118.00	54.00	0.25
2019-09-01	ETH Zurich	0	0.58	41.00	12.00	0.63
2019-09-01	Harvard University	3	0.44	243.57	40.29	0.06
2019-09-01	Imperial College	4	0.34	28.00	5.00	0.16
2019-09-01	MIT	3	0.23	84.00	27.00	0.00

Com base neste *dataframe* ainda não conseguimos realizar previsões. A etapa seguinte passa por reestruturar os dados desse *dataframe* em atributos que possam ser utilizados como dados de input para o treino de modelos num passo posterior. Neste momento, em cada observação, continuamos apenas com informação sobre ela, enquanto que necessitamos de mais informação sobre as observações que a precedem. Queremos portanto obter um *dataframe* com os seguintes atributos, para cada observação: o dia da publicação, os tópicos dominantes dos 7 dias anteriores, o número médio de retweets e favoritos durante esses dias, o dia da semana, mês do ano, a instituição responsável pela publicação e, por fim, o tópico alvo a determinar. Sendo assim, o processo de criação do *dataframe* descrito foi realizado através do seguinte código:

```
df['month'] = pd.DatetimeIndex(df['created_at']).month
df['year'] = pd.DatetimeIndex(df['created_at']).year
df[]'weekday'] = df.apply(lambda row: row['created_at'].weekday(), axis=1)

lista = []

"""limit the observations to on HEI at a time"""
for string in sorted(df['user.name'].unique()):
  temp = df[df['user.name'] == string].reset_index()
  temp['index'] = temp.index
  """create a new empty dataframe"""
  new_df = pd.DataFrame()
  i = 0
  """for each set of 7 days, determine the desired values"""
  while i <= len(temp)-8:
    seven_tweets = temp[(temp['index'] >= i) & (temp['index'] <= i+7)]
    day1 = seven_tweets['topic'].to_numpy()[0]
    day2 = seven_tweets['topic'].to_numpy()[1]
    day3 = seven_tweets['topic'].to_numpy()[2]
    day4 = seven_tweets['topic'].to_numpy()[3]
    day5 = seven_tweets['topic'].to_numpy()[4]
    day6 = seven_tweets['topic'].to_numpy()[5]
    day7 = seven_tweets['topic'].to_numpy()[6]
    year = seven_tweets['year'].to_numpy()[-1]
    month =  seven_tweets['month'].to_numpy()[-1]
```

```
    weekday = seven_tweets['weekday'].to_numpy()[-1]
    avg_retweets = seven_tweets['retweet_count'][:-1].mean()
    avg_favorites = seven_tweets['favorite_count'][:-1].mean()
    target_topic = seven_tweets['topic'].to_numpy()[-1]
    avg_polarity = seven_tweets['polarity'][:-1].mean()
    institution =  string
    """append the determined values to the dataframe"""
    new_df.append(pd.Series([day1,day2,day3,day4,day5,day6,day7,weekday,month,
        year,avg_retweets,avg_favorites,avg_polarity,institution,target_topic])
        ,ignore_index=True)
    i += 1

  new_df.columns =
      ['Day1','Day2','Day3','Day4','Day5','Day6','Day7','Year','Weekday','Month'
  ,'Avg Retweets','Avg Favorites','Avg Polarity','Institution','Target Topic']
  lista.append(new_df)

df = pd.concat(lista)
df = df.reset_index(drop=True)
```

Bloco de Código 5.9: Criação do dataframe a utilizar no treino de modelos

Através deste processo iterativo, para cada instituição, conseguimos associar a uma publicação os tópicos dominantes dos 7 dias anteriores à mesma, assim como outras métricas relativas a esse período (número médio de retweets, número médio de favoritos e valor médio de polaridade) e outros metadados (ano, mês, dia da semana).

Como podemos observar através do código, os atributos resultantes desta execução são maioritariamente categóricos, sendo necessária a sua conversão para numéricos de modo a realizar *feature selection* e utilizá-los no treino de diversos modelos. Para isso, realizamos a conversão desses atributos através de *one-hot encoding*, onde cada valor possível de um atributo é convertido numa coluna e é lhe atribuída um valor binário 0 ou 1. Os atributos afetados foram: 'Institution', 'Year', 'Month', 'Weekday', 'Day1', 'Day2', 'Day3', 'Day4', 'Day5', 'Day6' e 'Day7'. No caso específico dos atributos relativos à janela temporal de publicações anteriores ('Day1', ..., 'Day7') as colunas geradas dependem do modelo editorial utilizado, ou seja, da experiência a ser testada. Isto acontece porque o modelo da experiência 1 possui 5 possíveis áreas editoriais, enquanto que o modelo da experiência 2 possui 6. Visto que os valores dos atributos 'Day_n' são os tópicos dominantes relativos ao dia em questão e de acordo com o modelo utilizado na atribuição, quando o modelo da experiência 1 é utilizado, para cada um dos sete dias, são geradas apenas 5 colunas novas, enquanto que para o modelo da experiência 2 são geradas 6 colunas novas. Após esta conversão, fizemos a divisão dos conjuntos de dados de treino e de teste de acordo com um *split* de 80%/20%.

Uma das desvantagens mais aparentes é a grande dimensionalidade que este método acrescenta, sendo do conhecimento geral que normalmente uma menor quantidade de dimensões resulta em melhores resultados. No nosso caso o processo de conversão resultou numa *dataframe* com 71 dimensões, representando um aumento de 57 dimensões. Sendo assim, à semelhança da tarefa

de previsão anteriormente descrita, iniciamos um processo de *univariate feature selection* para diminuir a dimensionalidade e obter apenas os melhores atributos. Como a informação presente nos atributos originais ficou mais dispersa por uma grande quantidade de novos atributos, desta vez, decidimos selecionar os 20 atributos com maiores *scores* de informação mútua em relação à variável alvo 'Target Topic'. Estes foram os gráficos obtidos:

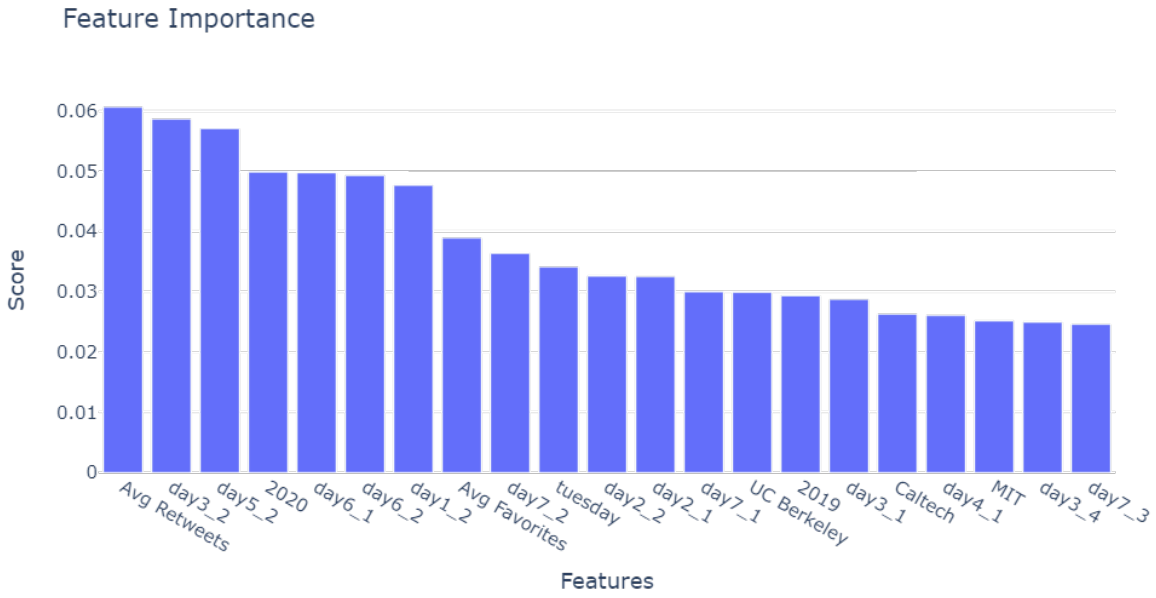

Figura 5.15: Importância de atributos (experiência 1)

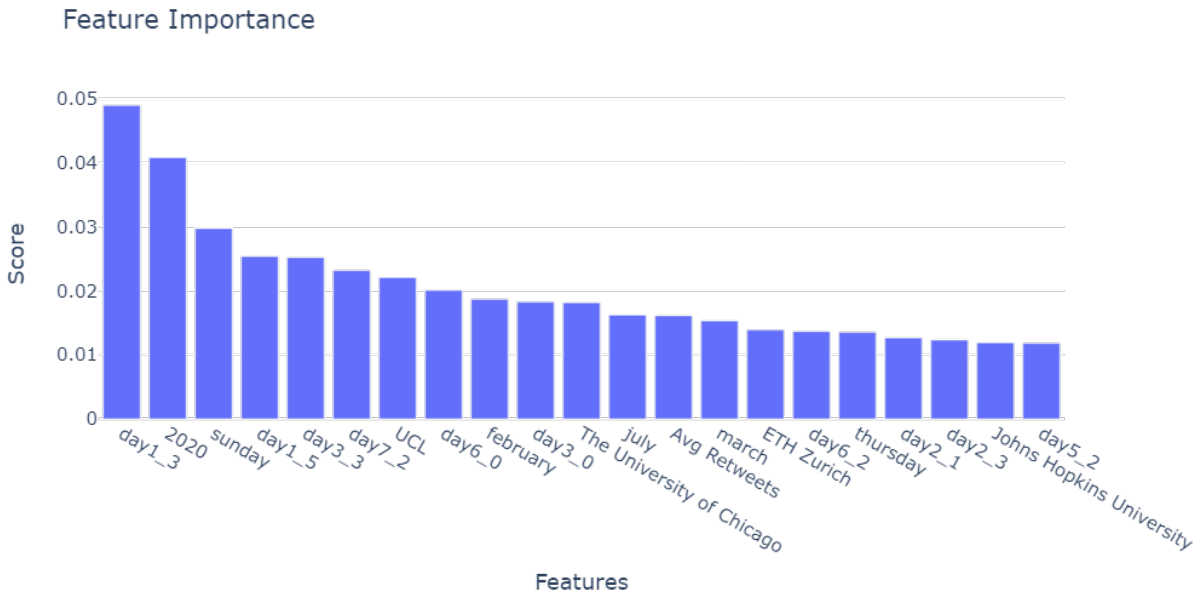

Figura 5.16: Importância de atributos (experiência 2)

Limitando os nossos conjuntos de dados a estes atributos, prosseguimos para o treino e avaliação de modelos utilizando 8 métodos diferentes, onde em cada um deles servimo-nos

novamente de uma *grid search*, com exatidão como métrica de avaliação, para fazer a otimização dos seus hiperparâmetros. Os métodos utilizados foram a regressão logística, 3 métodos de combinação de árvores de decisão (bagging, boosting, randomForest), árvores de decisão, k-nearest neighbors, máquinas de vetores de suporte, e perceptrons multicamadas.

Capítulo 6

Resultados e Discussão

Neste capítulo apresentamos os resultados obtidos nas diferentes vertentes de análise para cada uma das experiências realizadas e discutimos as diferenças observadas assim como as suas possíveis causas.

6.1 Análise Editorial

Após a determinação dos modelos editoriais a serem utilizados pelas HEI e a atribuição dos tópicos dominantes a cada publicação, é possível analisar várias vertentes das estratégias de comunicação obtidas em cada uma das experiências.

6.1.1 Experiência 1

Na experiência 1, onde o modelo editorial foi obtido através de estratégias não supervisionadas, e resultou em 5 tópicos diferentes, sendo eles 'Eventos', 'Saúde', 'Investigação', 'Educação' e 'Imagem, começamos por verificar as diferenças entre o número de publicações associadas a cada tópico (figura 6.1).

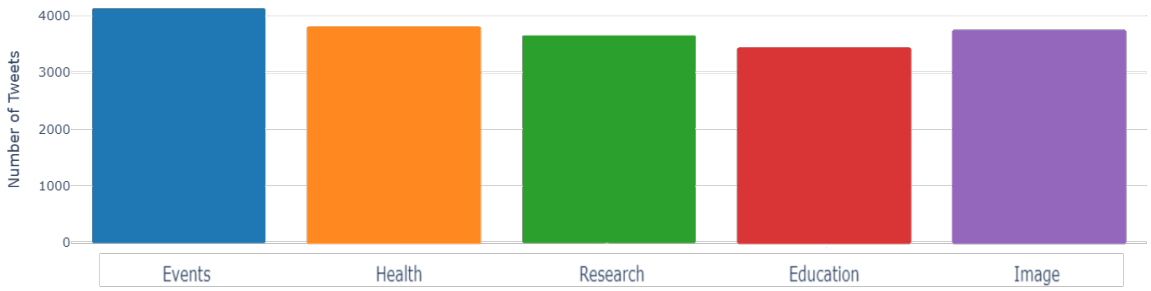

Figura 6.1: Distribuição de publicações por tópico (experiência 1)

Como é possível de observar, segundo este modelo, as áreas editoriais possuem uma presença relativamente equilibrada. De seguida exploramos a dimensão do corpo de texto das publicações para cada um dos tópicos, ou seja, o número de palavras para cada tópico obtido (figura 6.2):

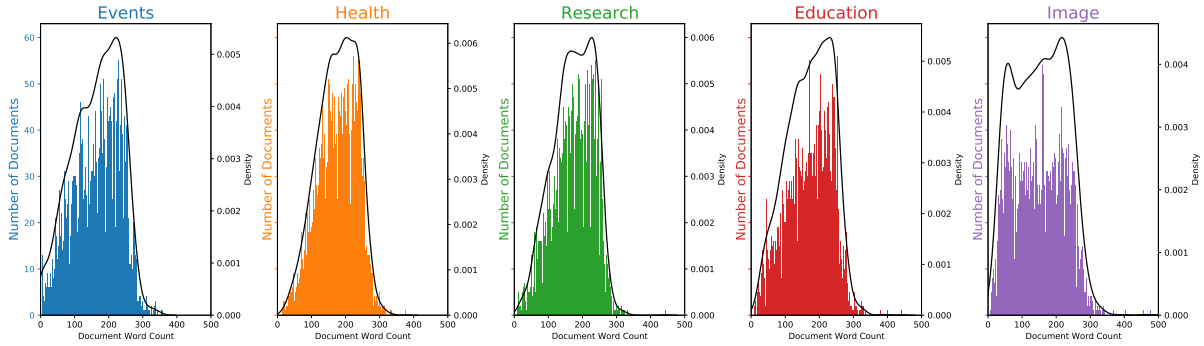

Figura 6.2: Distribuição do número de palavras por tópico (experiência 1)

Não aparentam existir diferenças significativas entre os primeiros 4 tópicos observados. As suas distribuições, apesar de possuírem algumas diferenças mínimas pontuais, apresentam formas semelhantes. A distribuição do número de palavras para o tópico de Imagem é que se revela mais diferente, com uma maior densidade de publicações menos extensas, seguida por uma brusca diminuição de densidade a partir das 100 palavras e, por fim, um crescimento quase constante até ao valor de 200 palavras, onde a distribuição assume novamente valores semelhantes às distribuições de outros tópicos.

Obtivemos também uma caracterização das estratégias de comunicação das HEI, onde cada instituição é representada pela quantidade total de publicações que realizou em cada área editorial (figura 6.3):

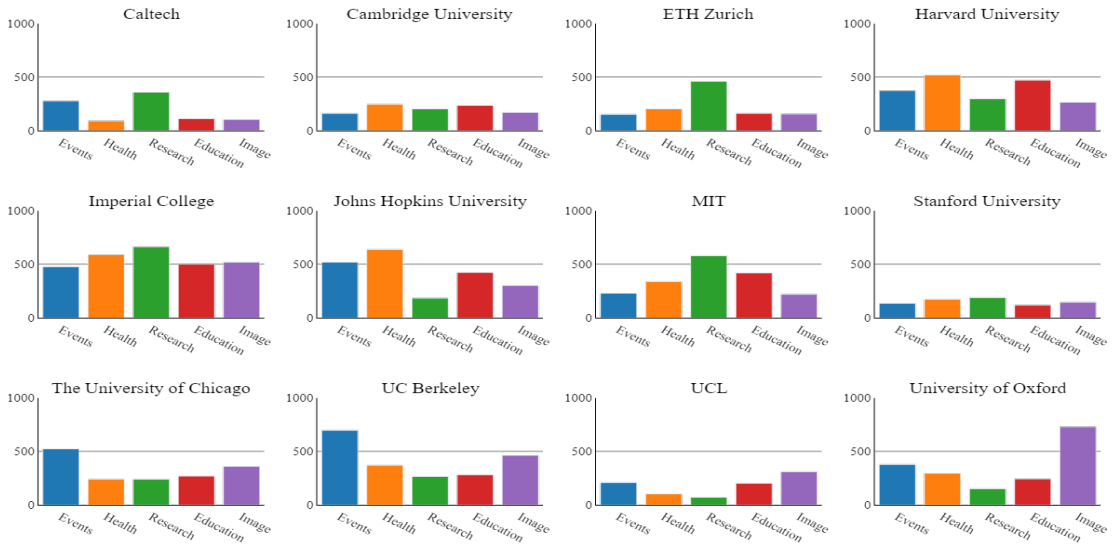

Figura 6.3: Distribuição do número total de publicações por HEI (experiência 1)

Através da figura anterior conseguimos observar que, para além de exceções como Stanford e Cambridge, que possuem uma distribuição bastante equilibrada, a maior parte das instituições não publica com a mesma frequência sobre todas as áreas editoriais disponíveis, o que resulta normalmente em uma ou duas áreas facilmente identificáveis como as áreas de maior interesse para cada uma delas. Podemos então associar a cada uma dessas universidades a sua área de maior interesse, sendo que para Caltech, Imperial College, ETH Zurich e MIT é a área de Investigação, para a University of Chicago e UC Berkeley é a área de Eventos, para a Johns Hopkins University e Harvard é a Saúde e por fim, para apenas a University of Oxford, é a Imagem. Segundo este esquema, a área da educação, apesar de às vezes perto, não se revelou a área de maior interesse em nenhuma das instituições estudadas, a respeito do número total de publicações.

De seguida, tentamos obter uma caracterização menos estática das estratégias de comunicação, onde para cada HEI, representamos a evolução temporal da distribuição das suas publicações pelos diferentes tópicos (figura 6.4):

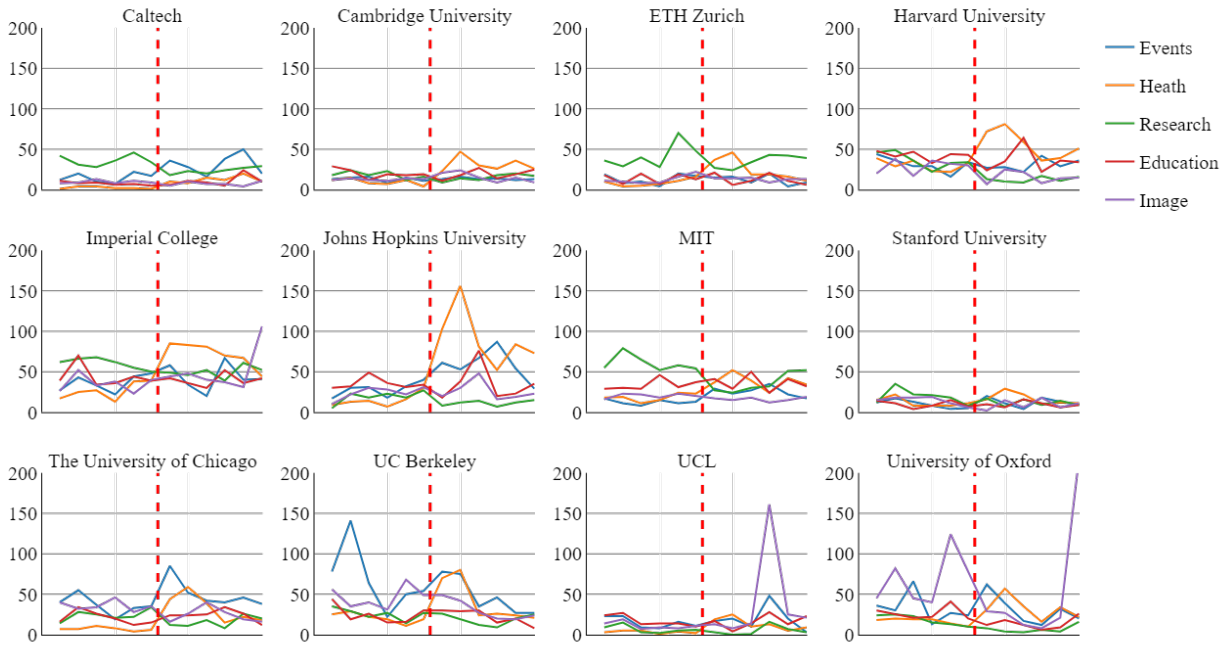

Figura 6.4: Distribuição de publicações mensais por HEI (experiência 1)

Nos gráficos da figura 6.4 o eixo temporal é partilhado para todas as HEI observadas. A reta vertical vermelha representa a data de declaração da Covid-19 como Pandemia, 11 de Março de 2020.

Um dos aspetos mais evidentes, que é facilmente observável em todas as HEI, é o aumento súbito e massivo do número de publicações de Saúde após a declaração da Covid-19 como uma pandemia por parte da OMS. Possivelmente correlacionado, podemos também observar uma

tendência decrescente partilhada do tema Investigação ao longo do ano académico, com o seu maior vale a coincidir, em muitos casos, com o aumento do tema Saúde. Isto pode ser devido às semelhanças semânticas entre os tópicos de Saúde e Investigação, uma vez que quando observada a distância entre tópicos do modelo, a distância deste par é mais baixa do que qualquer outro par possível. Este aspeto pode também ser explicado pelo facto dos esforços de investigação, ao longo do período estudado, estarem gradualmente a ser mais direcionados para a indústria da saúde, tendo em conta o agravamento da situação pandémica. Podemos dizer que o Imperial College, a UCL e a Universidade de Oxford possivelmente partilharam o mesmo evento, uma vez que todos exibem picos semelhantes no número de publicações de Imagem entre Junho de 2020 e Agosto de 2020. No caso da Universidade de Oxford, este pico coincide com o seu máximo absoluto em número médio de retweets e favoritos. Da mesma forma, Stanford e MIT também apresentam picos semelhantes em publicações de Investigação em Outubro de 2019, bem como Caltech e ETH Zurich, mais tarde nesse ano académico, em Janeiro de 2020.

As mudanças na frequência de publicações mensais por parte de várias HEI após a declaração pandémica merecem melhor análise. Se dividirmos o dataset com base nessa data, conseguimos obter caracterizações das estratégias em períodos antes e durante o Covid-19, no que diz respeito à distribuição dos tópicos dominantes (figuras 6.5 e 6.6):

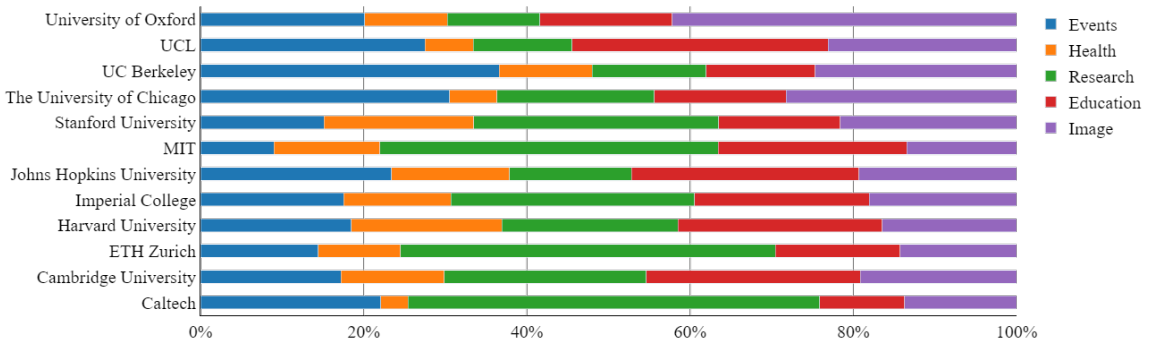

Figura 6.5: Frequências do número de publicações por tópico Pré Covid-19 (experiência 1)

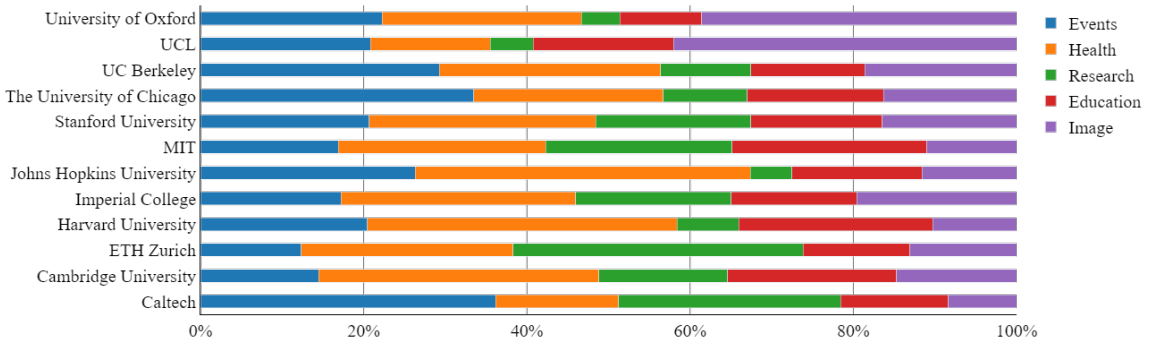

Figura 6.6: Frequências do número de publicações por tópico Pós Covid-19 (experiência 1)

Antes da pandemia, podemos ver que organizações como Stanford (30%), MIT (41%), Imperial College (30%), ETH Zurich (46%) e Caltech (50%) tinham a Investigação como principal tema de publicações. Apenas a UC Berkeley (37%) e a University of Chicago (31%) tiveram Eventos como tópico primário, embora esta última também possuísse uma preferência quase igual pela Imagem (28%). Centrando-nos em tweets orientados para a Educação, temos UCL (31%), Johns Hopkins (28%), Harvard (25%) e Cambridge (26%). No entanto, o grau de domínio deste tópico não é tão elevado como os anteriores. Todas as HEI com este tópico primário têm uma distribuição muito mais equilibrada e um segundo tópico proeminente com quase tanto foco como o primeiro. Tanto a UCL como Johns Hopkins dão prioridade ao tópico de Eventos no seu modelo editorial, enquanto que Harvard e Cambridge dão prioridade à Investigação. Finalmente, a Oxford tem 42% dos tweets no tópico Imagem, a única instituição com preferência por este tópico. Curiosamente, não existem HEI com uma estratégia que favoreça as publicações sobre o tema Saúde.

Após o início da Pandemia da Covid-19, podemos ver claramente mudanças significativas na estratégia de cada instituição. Como esperado, o tópico Saúde, que anteriormente não era o primeiro tópico de escolha em qualquer HEI, torna-se subitamente o foco principal da UC Berkeley (27%), Stanford (28%), MIT (25%), Johns Hopkins (41%), Imperial College (29%), Harvard (38%) e Cambridge (34%). UC Berkeley, Imperial College, Harvard e Cambridge mantêm o seu tópico dominante anterior como o segundo mais prevalecente, enquanto que nas restantes HEI isso não se observa. A University of Chicago, Oxford e a ETH Zurich mantiveram uma estratégia semelhante, mantendo o mesmo tópico principal de publicações. Contudo, o seu domínio diminuiu ligeiramente devido ao aumento do tópico Saúde presente em cada uma das estratégias pós Covid-19. UCL e Caltech tiveram as mudanças mais drásticas, sendo as únicas com uma mudança completa do tema dominante, mas não uma mudança para o domínio da Saúde. UCL junta-se à Universidade de Oxford com 42% de conteúdo relacionado com a imagem e a Caltech muda de Investigação para Eventos com 36%.

Para comparar o efeito da pandemia nas próprias áreas editoriais, obtivemos os gráficos de barras seguintes (figuras 6.7,6.8 e 6.9), que representam o número total de publicações, o número de retweets e o número de favoritos agrupados por tópico nas duas janelas temporais diferentes.

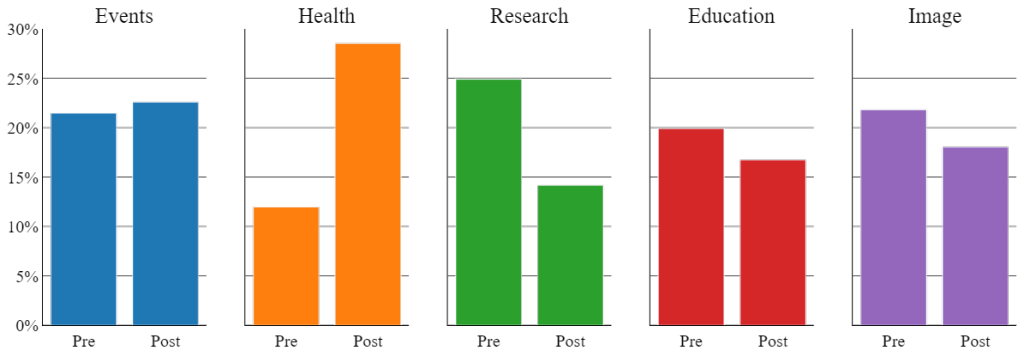

Figura 6.7: Percentagem de publicações por tópico (experiência 1)

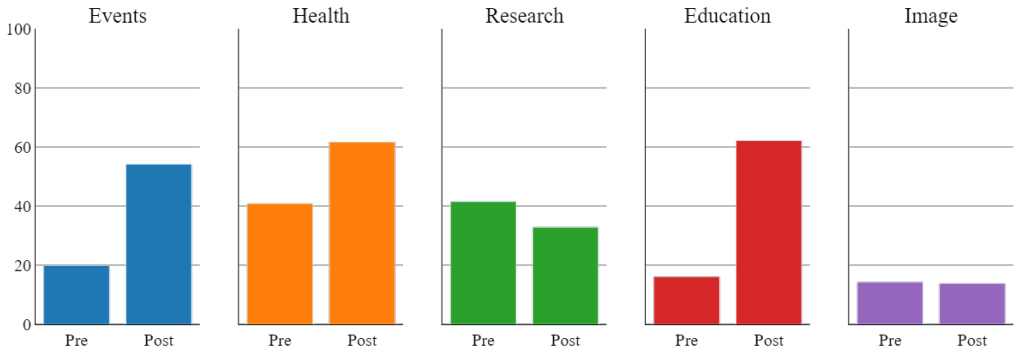

Figura 6.8: Número de retweets por tópico (experiência 1)

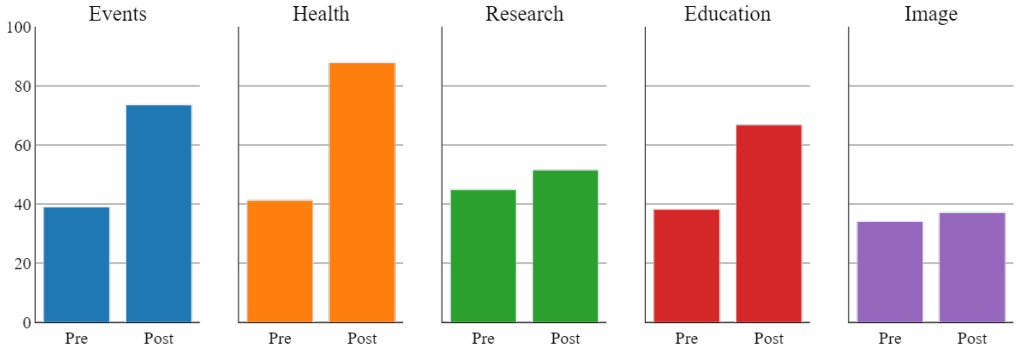

Figura 6.9: Número de favoritos por tópico (experiência 1)

Podemos ver que antes do Covid, a Investigação era o tema de publicação mais comum, representado por 25% dos Tweets. Após a pandemia, tornou-se o menos comum, sofrendo um decréscimo de cerca de 11%. Educação e Imagem também sofreram um decréscimo de 3% cada, fazendo de Eventos e Saúde os únicos tópicos cuja quota de publicação aumentou após a pandemia. A Saúde passou do menos comum para o mais comum, com um aumento de 17%.

Em relação ao número de retweets e favoritos, Eventos, Saúde e Educação tiveram resultados semelhantes, cada um aumentando o seu número de interações durante a pandemia. A Investigação e Imagem teve uma ligeira diminuição do número médio de retweets mas ainda assim observaram um aumento na contagem média dos favoritos, marcando um aumento global do número médio dos favoritos em todos os tópicos durante a pandemia.

6.1.2 Experiência 2

Na experiência 2, onde o modelo editorial foi obtido através de conhecimento de domínio, e resultou em 6 tópicos diferentes, sendo eles 'Educação', 'Emprego', 'Corpo Docente', 'Investigação' , 'Saúde' e 'Sociedade', procedemos a uma análise análoga à da experiência 1, começando por

verificar as diferenças entre o número de publicações associadas a cada tópico (figura 6.10).

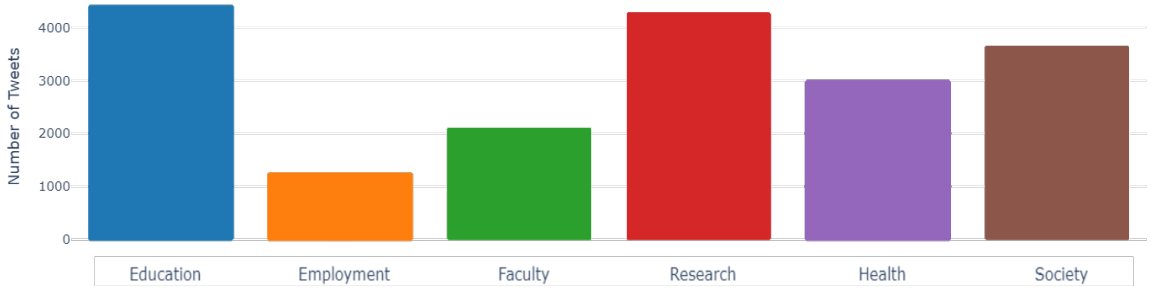

Figura 6.10: Distribuição de publicações por tópico (experiência 2)

Ao contrário da experiência anterior, as áreas editoriais segundo este modelo possuem presenças desequilibradas. A área com maior número de publicações é a Educação, seguida pela Investigação, num segundo lugar bastante próximo. Depois temos a Sociedade e Saúde em níveis semelhantes e, por fim, Corpo Docente e Emprego, sendo que esta última possui menos de 1/3 das publicações da área mais publicada. De seguida exploramos o número de palavras para cada tópico obtido (figura 6.11):

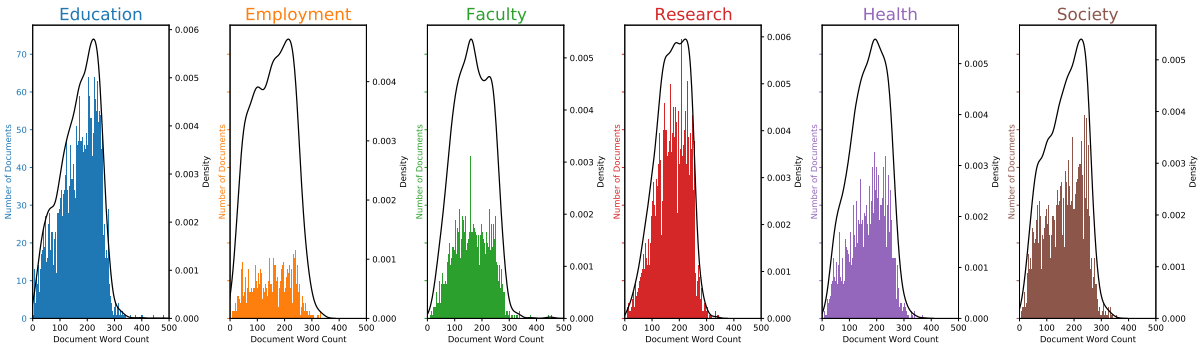

Figura 6.11: Distribuição do número de palavras por tópico (experiência 2)

Neste caso, as distribuições apresentam menores diferenças entre elas e não aparenta existir uma distribuição substancialmente diferente das suas homólogas, como era o caso do tópico Imagem na experiência anterior. A única diferença mais saliente e fácil de apontar é observada no tópico do Corpo Docente, onde a tendência decrescente do número de documentos com o aumento do número de termos começa ligeiramente mais cedo, por volta dos 150 termos, onde os restantes tópicos apenas registam esse aspeto a decididamente a partir dos 200 tópicos.

Mais uma vez, obtivemos as estratégias de comunicação das HEI através da figura seguinte (figura 6.12), onde cada instituição é representada pela quantidade total de publicações que realizou em cada área editorial.

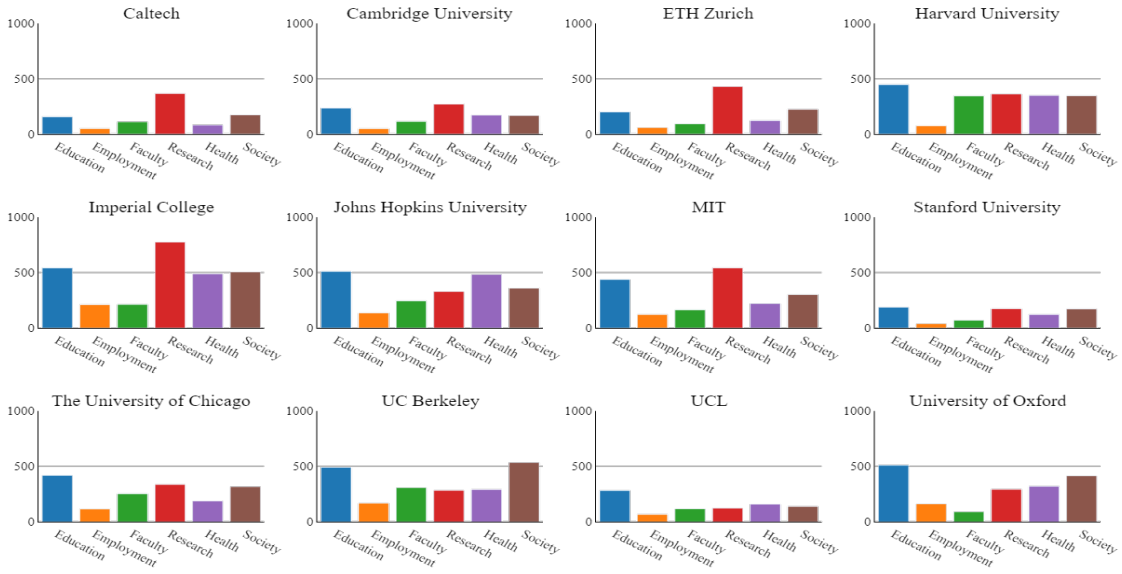

Figura 6.12: Distribuição do número total de publicações por HEI (experiência 2)

De seguida, analisamos a evolução temporal da distribuição das suas publicações pelos diferentes tópicos (figura 6.13):

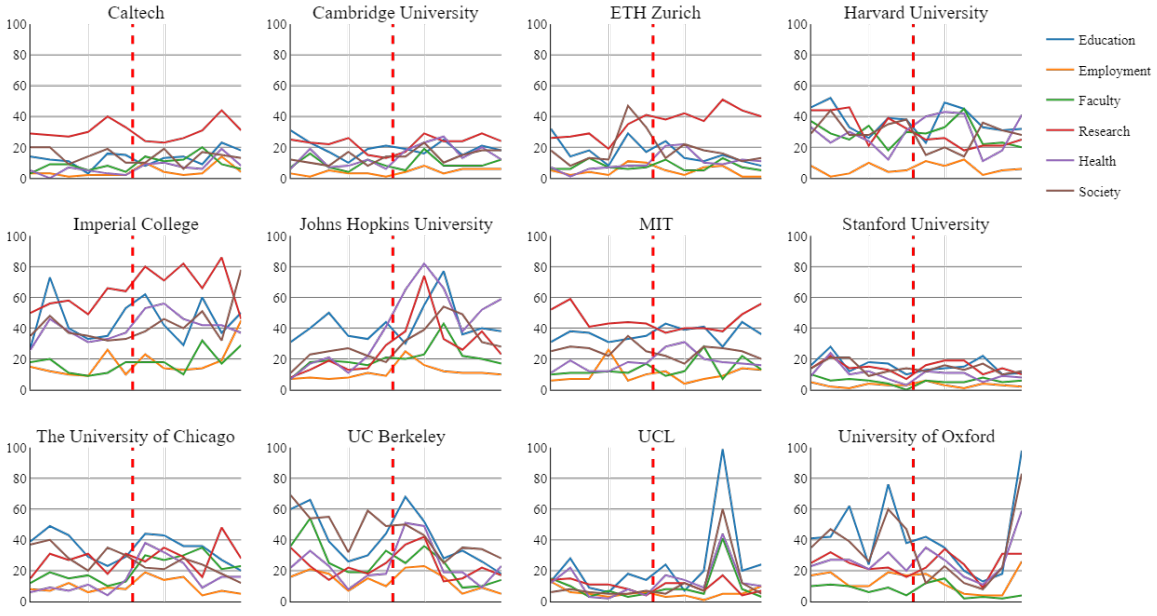

Figura 6.13: Distribuição de publicações mensais por HEI (experiência 2)

A respeito da figura 6.12, à semelhança da experiência anterior, a maior parte das instituições não publica com a mesma frequência sobre todas as áreas editoriais disponíveis. No entanto, a

área da Educação, que não se revelou de maior interesse para nenhuma HEI no modelo anterior, passa agora à área de maior interesse, com um maior número de publicações por parte de Harvard, Johns Hopkins, Stanford, University of Chicago, UCL e Oxford. Em segundo classificado aparece a área de investigação, também com uma grande prevalência, sendo a área de maior interesse por parte de Caltech, Cambridge, ETH Zurich, Imperial College e MIT. Por fim, temos a UC Berkeley com preferência pela área da Sociedade, sendo que Emprego e Corpo Docente não possuem o maior número de publicações em nenhuma das instituições estudadas.

Em relação à evolução temporal representada na figura 6.13, os resultados são menos homogéneos entre as diferentes instituições, em comparação com a experiência 1. Algo que esta experiência possui em comum com a anterior é novamente a subida do número de publicações da área da saúde no período imediatamente após a declaração da pandemia por parte da OMS. No entanto, parecem não existir mais tendências globais em outras áreas para todas as instituições estudadas, como acontecia com a área da Investigação na análise anterior. Em vez de isso, as variações do número de publicações mensais parecem mais específicas a cada instituição, e a grande parte dos cumes e vales que encontramos no número de publicações de um determinado tópico estão normalmente associados a mais do que um tópico, ou seja, os cumes são normalmente indicadores de uma alteração geral de atividade e não de uma mudança estratégica.

Submetendo este conjunto de dados à mesma análise pré/pós Covid-19, definida anteriormente, obtivemos os resultados seguintes (figuras 6.14 e 6.15):

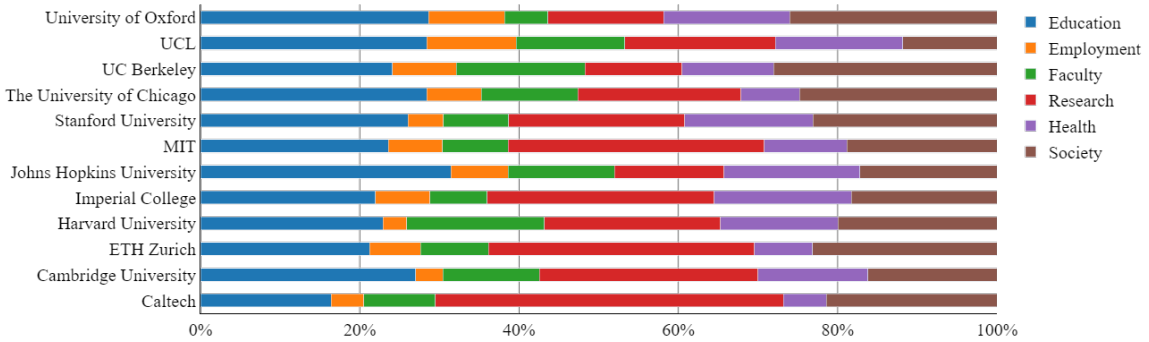

Figura 6.14: Frequências do número de publicações por tópico Pré Covid-19 (experiência 2)

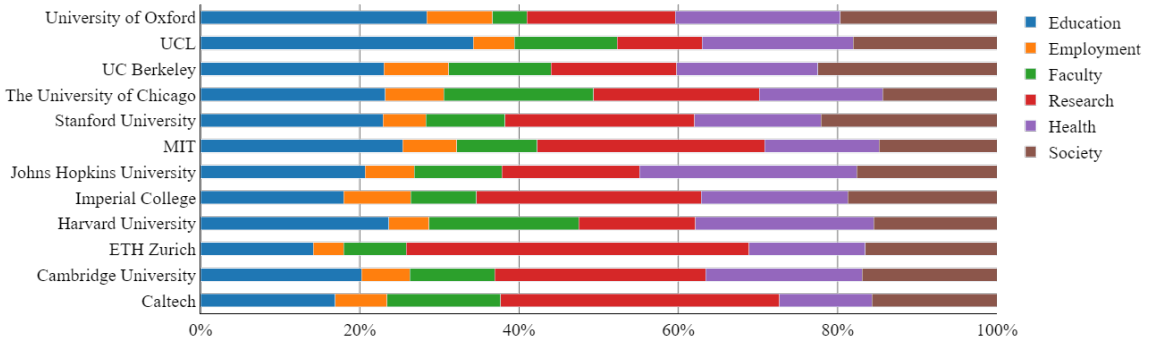

Figura 6.15: Frequências do número de publicações por tópico Pós Covid-19 (experiência 2)

No período pré Covid-19, podemos verificar a seguinte distribuição das instituições por tópico dominante mais frequente: Oxford (29%), UCL (28%), University of Chicago (28%), Stanford (26%), Johns Hopkins (31%) e Harvard (22%) tinham a área da Educação como o principal tema de publicações. Já MIT (32%), Imperial College (29%), ETH Zurich (33%) e Caltech (44%) possuíam mais publicações na área de Investigação. A Cambridge University (27%), divide a sua área de maior incidência de forma igual para estes dois tópicos. Por fim, temos a UC Berkeley, sendo a única instituição com maior incidência na área da Sociedade. Podemos observar que as áreas de Emprego, Corpo Docente, e Saúde não são as áreas com maior frequência de tweets em nenhuma instituição neste período. Um aspeto também interessante de observar é o equilíbrio das distribuições. As instituições com maior frequência no tópico da Investigação têm por norma uma percentagem considerável de diferença do segundo tópico com maior frequência (sendo Cambridge a única exceção, por possuir áreas de maior incidência partilhadas). No entanto, é mais comum encontrar estratégias com um segundo tópico predominante em HEI onde a Educação assume o tema mais frequente de publicações. Nestes casos, o tema acessório possui quase tanto foco como o principal, sendo que na University of Chicago, Oxford e Stanford esse tópico secundário é a Sociedade, em Harvard é a Educação e UCL e Johns Hopkins figuram como exceções. A diferença entre estas áreas nunca ultrapassa os 3 pontos percentuais.

Após o início da pandemia conseguimos observar através da figura 6.16 algumas mudanças na estratégia de cada instituição, apesar de estas aparentarem ser menos drásticas do que as obtidas na experiência anterior. Como esperado, à semelhança dos resultados anteriores, temos o aumento significativo do número de publicações no tópico da Saúde (apesar de mais reduzido), já que todas as HEI mantiveram ou subiram a sua quota de publicações nessa área. Esta subida resultou apenas numa única mudança do tópico dominante para esta área, por parte da Johns Hopkins University (27%), em contraste com as 7 mudanças para este tópico verificadas na experiência anterior. Outras instituições que mudaram de tópico principal de publicações foram apenas UC Berkeley (23%) para Educação (23%) e Stanford (24%) e Cambridge (26%) para a Investigação, sendo que as restantes mantiveram o mesmo tópico. Com a exceção de Stanford, onde o tópico principal e secundário diferem em apenas 1 ponto percentual, a dominância do tópico principal na área da Investigação continua a estar presente no período pós-pandemia.

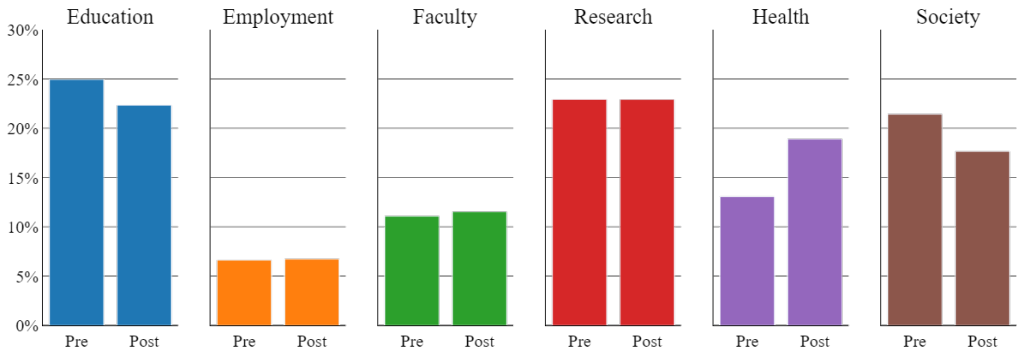

Figura 6.16: Percentagem de publicações por tópico (experiência 2)

Concluímos mais uma vez com a análise dos efeitos da pandemia nas próprias áreas editoriais, analisados através dos gráficos de barras representados nas figuras 6.16,6.17 e 6.18

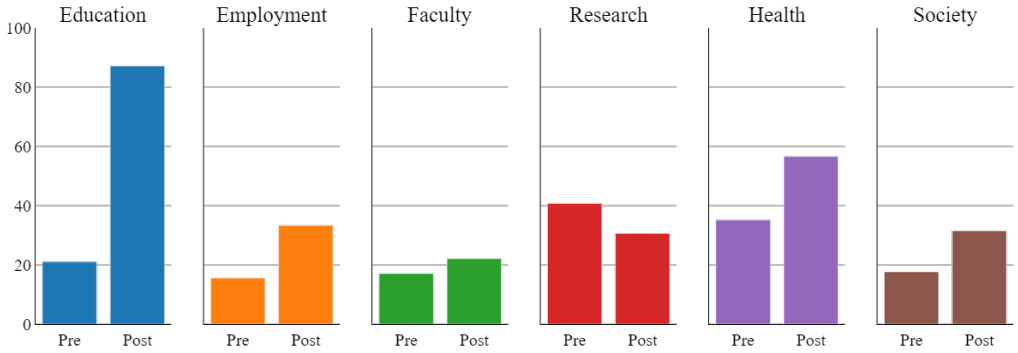

Figura 6.17: Número de retweets por tópico (experiência 2)

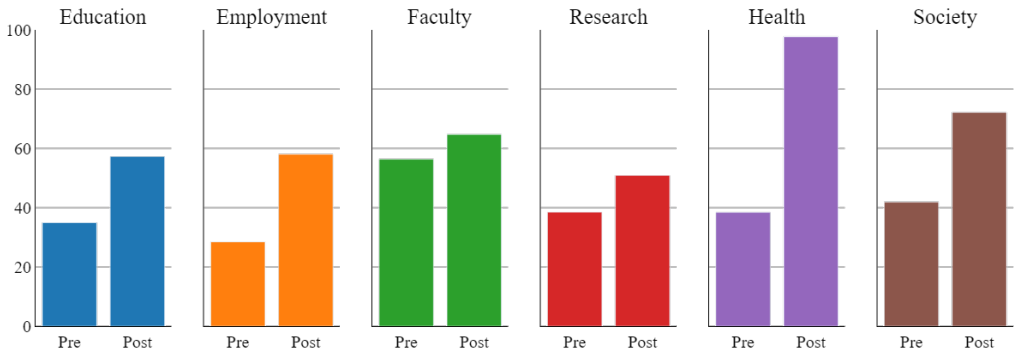

Figura 6.18: Número de favoritos por tópico (experiência 2)

Podemos ver que antes do Covid, a Educação era o tema de publicação mais comum, seguido pela Investigação, Sociedade, Saúde, Corpo Docente e Emprego. Após a pandemia, Educação e Sociedade diminuíram a sua quota de publicações em 3% e 4% respetivamente, enquanto que Saúde aumentou em 6% e os restantes temas mantiveram as mesmas dimensões. Isto permitiu a margem suficiente para a Investigação assumir a posição de área de publicação mais comum, por cerca de 1%. Em comparação com a experiência anterior, as alterações observadas são menos acentuadas, o que revela maior estabilidade entre os dois períodos, que vai de acordo com o observado nos gráficos anteriores.

Em relação ao número de retweets e favoritos conseguimos observar uma mudança tremenda no número médio de retweets, com um aumento de mais de 300% das publicações na área da Educação. As restantes áreas possuíram também crescimentos no número de retweets, embora mais ligeiros, exceto a Investigação, que diminui o seu número de retweets médios em cerca de 25%. No entanto, o número de favoritos aumentou para as publicações de todas as áreas, sendo que a mudança mais significativa ocorreu na área da saúde, onde este valor aumentou para mais do dobro do período pré Covid-19.

6.2 Análise de Sentimento

Nesta secção abordamos os resultados obtidos a respeito da análise do sentimento presente no conteúdo das publicações selecionadas. Inicialmente, testamos uma possível correlação entre a polaridade de uma a publicação e o seu número de retweets e favoritos.

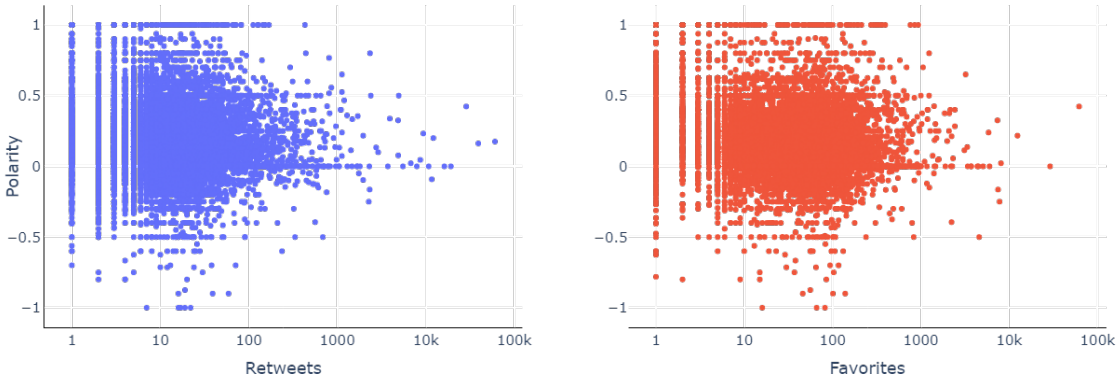

Figura 6.19: Relação da polaridade com retweets e favoritos

A partir da figura 6.19 conseguimos observar que não parece existir uma relação de proporcionalidade direta entre a polaridade e as métricas de envolvimento. No entanto, conseguimos ainda assim obter informação relevante a partir destes gráficos. Observando o extremo superior, com os valores mais elevados de retweets e favoritos, conseguimos perceber que as publicações mais neutras estão mais representadas. A partir de cerca de 1000 retweets ou favoritos são de longe mais frequentes as publicações de caráter mais neutro.

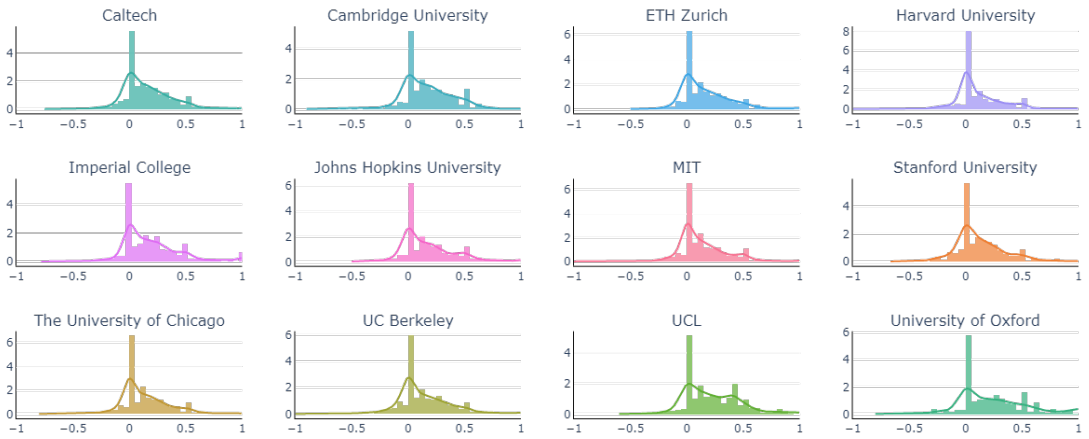

Figura 6.20: Distribuição de polaridade por HEI

Tentamos também observar as distribuições de polaridade específicas às instituições, 6.20, na tentativa de encontrar semelhanças e/ou diferenças através das quais pudéssemos caracterizar a sua estratégia

As distribuições de polaridade, e consequentemente, as distribuições de sentimento das diferentes HEI são praticamente idênticas. Podem ser todas geralmente caracterizadas por uma alta densidade de publicações neutras, com um número cada vez mais reduzido de publicações à medida que nos aproximamos dos extremos, quase à semelhança de uma distribuição normal. Ao contrário de uma distribuição normal, esta distribuição não possui simetria bilateral, sendo este o principal aspeto que as diferenciam. O número de publicações negativas é significativamente menor do que o número de publicações positivas, sendo que estas encontram-se maioritariamente entre o intervalo]0,0.5] de polaridade.

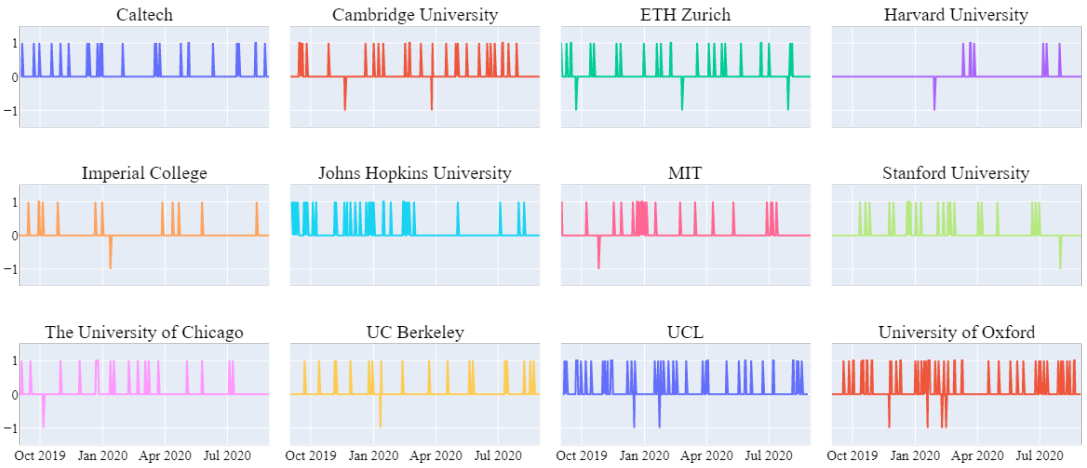

Figura 6.21: Evolução pontual do sentimento por HEI

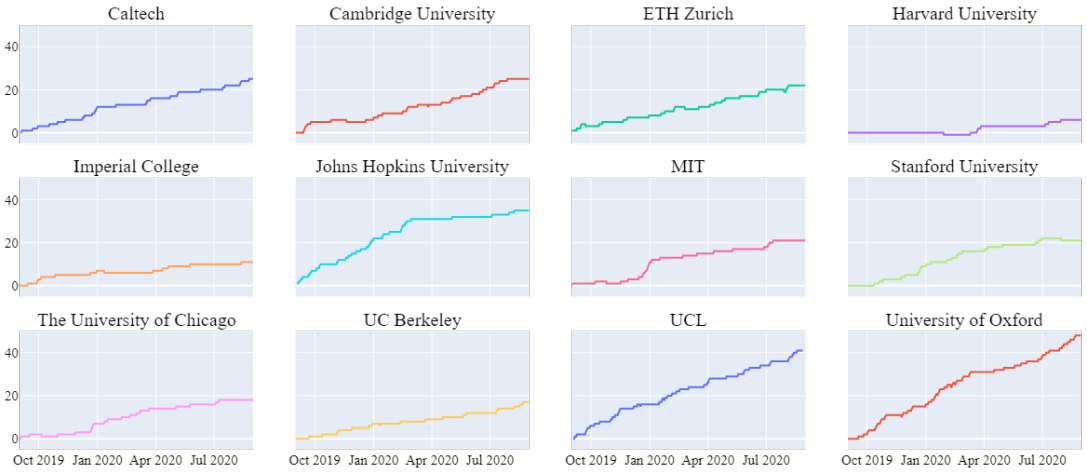

Figura 6.22: Evolução acumulada do sentimento por HEI

De seguida, decidimos observar a sazonalidade e as tendências do sentimento de cada HEI ao longo do tempo. Para tal, obtivemos gráficos para a evolução pontual e acumulada do sentimento para cada HEI. Para cada dia do período estudado foi calculada a média da polaridade dos seus tweets e arredondado esse valor para obter o sentimento médio desse dia. Nos gráficos acumulados, os valores dos dias seguintes são sempre somados aos dias anteriores. Os resultados estão representados nas duas figuras anteriores.

Com base nos gráficos de linhas da figura 6.21, descobrimos que o sentimento diário vai de encontro às distribuições observadas anteriormente, apesar de não conseguirmos identificar nenhum padrão óbvio partilhado por mais do que uma HEI. Grande parte das instituições possuem predominantemente períodos de duração considerável do sentimento neutro (sendo Harvard o melhor exemplo dessa realidade). A interromper estes períodos observamos mais frequentemente dias positivos do que negativos, como era espectável.

Para observar mudanças comportamentais, a figura 6.22 revelou-se mais útil, porque apenas é necessário determinar períodos com decréscimos, estagnações ou alterações de crescimento. A instituição que mais sobressaí nesta análise é a Johns Hopkins University, que a partir do início do mês de Março de 2020, mês onde foi declarado a Covid-19 como pandemia, demonstra uma diminuição significativa da taxa de crescimento, que perdura até ao final do ano letivo, sugerindo assim uma mudança estratégica a partir desse período. Esta característica está de acordo com a estratégia editorial pré e pós Covid-19 identificada pelo modelo obtido na experiência 2 (figuras 6.14 e 6.15). Segundo esse modelo, a Johns Hopkins University é a única instituição a alterar o seu tópico dominante para a área da Saúde no período pós pandémico, sendo assim natural que esta fosse a única HEI a demonstrar este tipo de comportamento na análise atual.

Ao observarmos as diferenças de sentimento pré e pós Covid, para cada instituição, podemos ganhar uma melhor perceção do que aconteceu no caso anterior, assim como no caso de outras instituições, cujas alterações não sejam percetíveis no gráfico anterior:

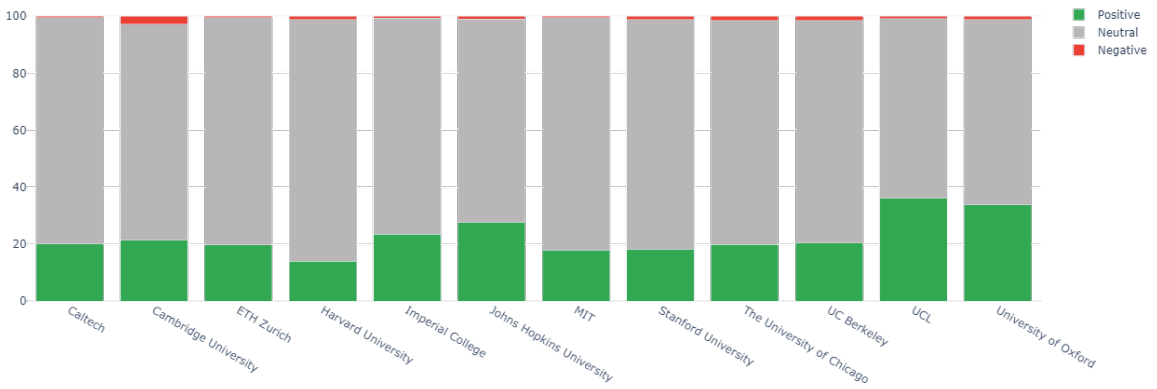

Figura 6.23: Distribuição do sentimento por HEI (pré Covid-19)

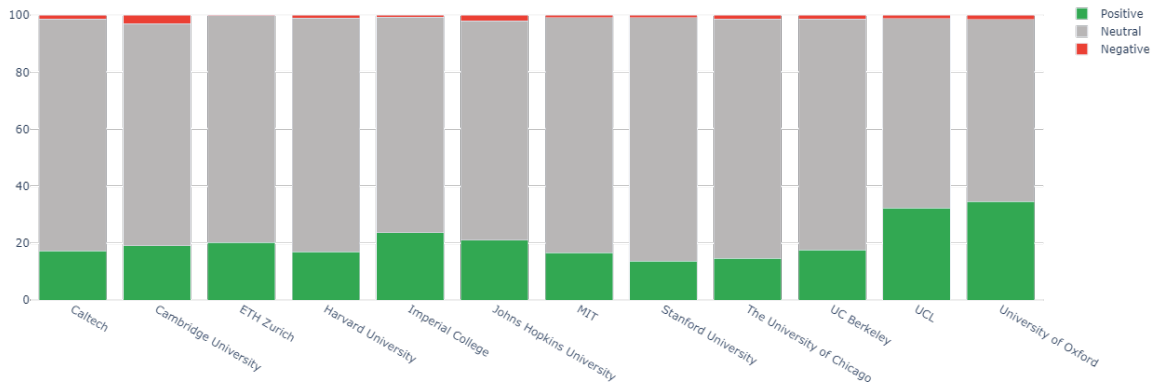

Figura 6.24: Distribuição do sentimento por HEI (pós Covid-19)

De acordo com os gráficos anteriores (figuras 6.23 e 6.24), mais uma vez, apenas Johns Hopkins demonstra diferenças evidentes na distribuição do sentimento entre os dois períodos estudados, sendo que as restantes instituições apresentam um caráter de sentimento mais consistente. Esta diferença é essencialmente associada ao número de positivos, que diminuiu significativamente (7%), enquanto que o número de tweets neutros subiu consideravelmente (6%) e o número de tweets negativos tiveram uma ligeira subida (1%). Estes valores estão em conformidade com o gráfico do sentimento acumulado presente na figura 6.22 para a instituição em questão, ou seja, a diminuição do número de tweets positivos e aumento de tweets negativos e neutros são o motivo da alteração no crescimento observada.

Por fim, para esclarecer estes resultados, procuramos obter a distribuição do sentimento por tópico, nos períodos pré e pós Covid, para cada experiência (figuras 6.25 e 6.26).

6.2.1 Experiência 1

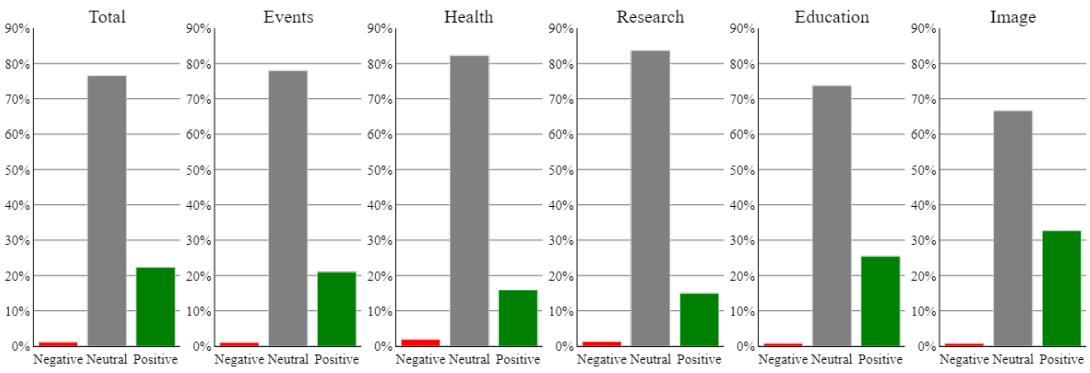

Figura 6.25: Distribuição do sentimento por tópico Pré Covid-19 (experiência 1)

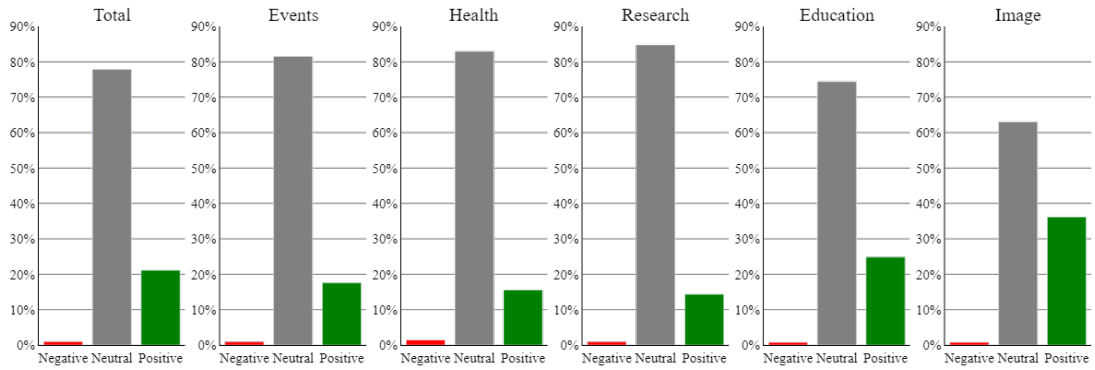

Figura 6.26: Distribuição do sentimento por tópico Pós Covid-19 (experiência 1)

Como os resultados para esta experiência não têm implicações diretas nos resultados das análises de sentimento anteriores (ou seja, mudanças estratégicas que ocorreram segundo o modelo da experiência 1 não foram possíveis de observar na evolução temporal do sentimento), apenas fazemos uma análise mais geral dos dois períodos, sem elaborar muito nas especificidades para cada tópico nem contrastar com as estratégias editoriais determinadas.

No período anterior à pandemia, a distribuição total tinha a seguinte estrutura: 77% das publicações de caráter neutro, 22% de caráter positivo e apenas 1% de caráter negativo. Neste período podemos salientar que o tópico com maior percentagem de tweets neutros é a Investigação, enquanto que a maior percentagem de tweets positivos pertence à Imagem e a maior percentagem de tweets negativos pertence à Saúde.

Após o início da pandemia, a distribuição total manteve a mesma percentagem de tweets negativos e subiu 1% na percentagem de tweets neutros (a percentagem de tweets positivos, consequentemente, desceu 1%). O tópico Imagem conseguiu uma subida de 3% do sentimento positivo, mantendo-se assim como o tópico com maior percentagem de sentimento positivo por uma larga margem. Os restantes tópicos mantiveram ou diminuíram a sua percentagem de tweets positivos (num decréscimo máximo de três pontos percentuais). O tópico com maior percentagem de tweets neutros continua a ser a Saúde, mas a maior percentagem de tweets negativos passou a ser um contestada por todos os tópicos, visto que todos eles partilham 1% de tweets negativos neste período.

6.2.2 Experiência 2

Para a experiência 2, vamos colocar ênfase na observação dos tópicos dominantes da Johns Hopkins University, visto que os resultados obtidos desta análise valorizaram o modelo editorial da experiência em questão. Segundo esse modelo editorial, esta instituição passou do tópico dominante da Educação antes da pandemia, para o tópico dominante da Saúde após a pandemia. Nas figuras 6.23 e 6.24 conseguimos confirmar que a alteração de crescimento do sentimento

acumulado para esta instituição se deveu a uma diminuição de 7% de tweets positivos e um aumento de 1% de tweets negativos. Sendo assim, se tópico da Saúde se revelar menos positivo após a pandemia do que o tópico da Educação (com maior número de tweets negativos e menor número de tweets positivos), possuímos uma boa base para concluir que o modelo da experiência 2 foi capaz de caracterizar corretamente a estratégia de comunicação dessa HEI, e possivelmente de outras (figuras 6.27 e 6.28).

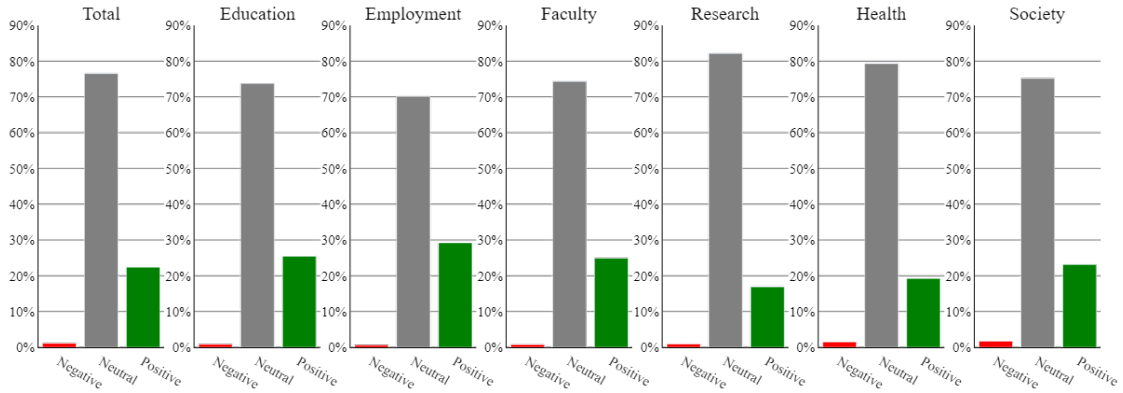

Figura 6.27: Distribuição do sentimento por tópico Pré Covid-19 (experiência 2)

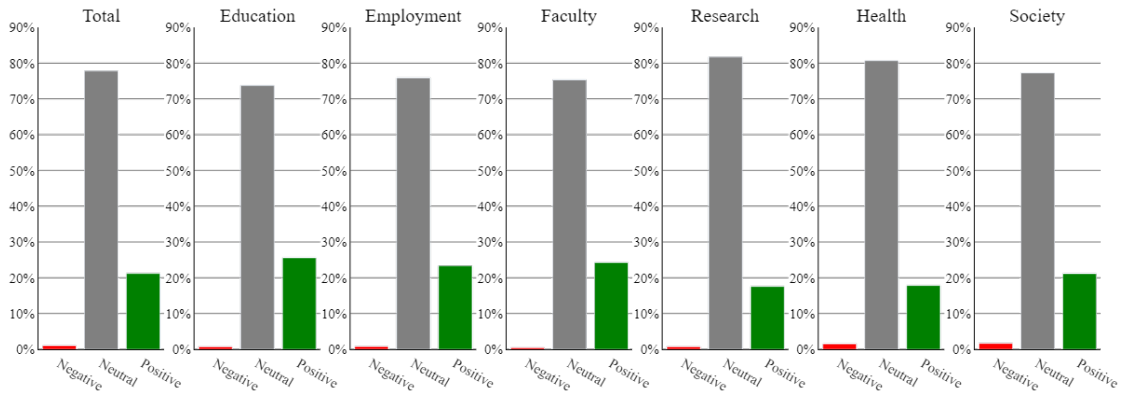

Figura 6.28: Distribuição do sentimento por tópico Pós Covid-19 (experiência 2)

No período pré Covid-19, conseguimos verificar que o tópico da Educação possui a seguinte distribuição: 74% de tweets neutros, 25% de tweets positivos e 1% de tweets negativos. Isto faz do tópico Educação o segundo tópico com maior número de tweets positivos (empatado com Corpo Docente), apenas atrás do tópico Emprego por 4 pontos percentuais. É também um dos 4 tópicos com menor número de tweets negativos (Educação, Emprego, Corpo Docente e Investigação).

Do mesmo modo, no período pós Covid-19, conseguimos verificar que o tópico Saúde possui a

seguinte distribuição: 81% de tweets neutros, 18% de tweets positivos e 2% de tweets negativos. Sendo assim, o tópico Saúde revela-se o tópico com menor número de tweets positivos (empatado com a Investigação) assim como, juntamente com o tópico Sociedade, o tópico com maior número de casos negativos. A alteração de tópico dominante por parte da Johns Hopkins University levaria a uma diminuição média de 7% do número de casos positivos (de 25% para 18%) e um aumento de 1% do número de casos negativos (de 1% para 2%). Estes valores, ao serem exatamente iguais aos valores obtidos em 6.23 e 6.24, não só confirmam que esta mudança estratégica é uma explicação credível para a evolução do sentimento observada, mas confirmam também que, no caso desta HEI, apenas a alteração de tópico dominante é refletida no sentimento, sendo que as alterações das percentagens nas restantes áreas ou não tiveram impacto no sentimento ou os seus impactos individuais foram anulados entre si.

6.3 Análise Preditiva

Nesta secção, apresentamos os resultados dos modelos treinados nas tarefas: previsão do número de retweets de uma publicação e previsão do tópico dominante de uma publicação.

6.3.1 Experiência 1

Na tarefa de previsão do número de retweets, obtivemos os seguintes resultados para a experiência 1:

Tabela 6.1: Desempenho dos diferentes modelos para a tarefa de regressão (experiência 1)

Modelo	MSE	RMSE	MAE	R2
Regressão Linear	91.1129	9.5453	6.8918	0.2998
Regressão de Ridge	91.1036	9.5448	6.8932	0.2999
Regressão de Lasso	91.1118	9.5452	6.8943	0.2998
Árvore de Decisão	81.3809	9.0211	6.3894	0.3746
Bagging	131.3328	11.4601	8.5720	-0.0093
Boosting	87.6635	9.3629	6.9248	0.3263
Random Forest	77.7842	8.8195	6.2419	0.4022
K-NN	93.6225	9.6759	6.7605	0.2805
SVM	93.7397	9.6819	6.3183	0.2796
MLP	83.9911	9.1647	6.5326	0.3546

Como podemos observar, de todos os modelos utilizados, o modelo Random Forest possui o melhor desempenho em todas as métricas utilizadas.

Para a tarefa de previsão do tópico dominante de uma publicação, obtivemos os seguintes resultados:

Tabela 6.2: Desempenho dos diferentes modelos para a tarefa de classificação (experiência 1)

Modelo	Exatidão	Precisão	Sensibilidade	F1
Regressão Logística	0.4082	0.3681	0.3791	0.3675
Árvore de Decisão	0.3691	0.3423	0.3390	0.3221
Bagging	0.4043	0.3616	0.3636	0.3266
Boosting	0.4102	0.3868	0.3753	0.3456
Random Forest	0.4336	0.4235	0.3981	0.3784
K-NN	0.4043	0.3928	0.3752	0.3508
SVM	0.4160	0.3854	0.3820	0.3583
MLP	0.4238	0.3823	0.3933	0.3791

Nesta tarefa, o modelo que apresenta melhor resultados de exatidão, precisão e sensibilidade é o Random Forest, enquanto que o MLP apresenta a melhor F1-*score*, por uma diferença de 0,0007 unidades.

6.3.2 Experiência 2

Do mesmo modo, para a tarefa de previsão do número de retweets, obtivemos os seguintes resultados para a experiência 2:

Tabela 6.3: Desempenho dos diferentes modelos para a tarefa de regressão (experiência 2)

Modelo	MSE	RMSE	MAE	R2
Regressão Linear	96.9085	9.8442	7.0110	0.2814
Regressão de Ridge	96.8996	9.8438	7.0112	0.2815
Regressão de Lasso	96.9085	9.8442	7.0110	0.2814
Árvore de Decisão	87.4559	9.3518	6.6444	0.3515
Bagging	83.7966	9.1540	6.6214	0.3786
Boosting	90.4698	9.5116	7.0274	0.3292
Random Forest	82.6370	9.0905	6.4214	0.3872
K-NN	84.7157	9.2041	6.4311	0.3718
SVM	94.1320	9.7022	6.2479	0.3020
MLP	84.0360	9.1671	6.4982	0.3769

Mais uma vez, o modelo Random Forest é o que apresenta melhor desempenho em todas as métricas utilizadas. Para a tarefa de previsão do tópico dominante de uma publicação, os resultados foram os seguintes:

Tabela 6.4: Desempenho dos diferentes modelos para a tarefa de classificação (experiência 2)

Modelo	Exatidão	Precisão	Sensibilidade	F1
Regressão Logística	0.2788	0.2173	0.2122	0.2086
Árvore de Decisão	0.3091	0.1642	0.1805	0.1487
Bagging	0.3414	0.1966	0.2042	0.1789
Boosting	0.3414	0.1903	0.2033	0.1762
Random Forest	0.3515	0.2099	0.2117	0.1893
K-NN	0.3414	0.1886	0.2033	0.1772
SVM	0.3455	0.2057	0.2140	0.1937
MLP	0.3475	0.2022	0.2153	0.1988

O modelo Random Forest apresenta os melhores resultados de exatidão e sensibilidade, enquanto que o modelo de regressão logística apresenta a melhor precisão e F1-*score*.

Capítulo 7

Conclusões

Nesta dissertação, utilizámos o ranking da CWUR para identificar as 12 HEI com melhores classificações, das quais recolhemos 18727 tweets, para o período de 01/09/2019 a 31/08/2020. De seguida, explorámos diferentes técnicas de identificação de tópicos de acordo com modelos editoriais pré-definidos ou obtidos através de aprendizagem não-supervisionada. Utilizando estes modelos, caracterizámos as estratégias de comunicação de cada HEI através da identificação das suas áreas editoriais, comparámos as estratégias resultantes entre si e analisámos as suas diferenças antes e depois da declaração pandémica da Covid-19 por parte da OMS. Com base nas estratégias de comunicação determinadas, realizámos duas tarefas de aprendizagem supervisionada: i) previsão do número de retweets de uma publicação e ii) previsão da área editorial de uma nova publicação.

Com base nos resultados obtidos, conseguimos determinar que o modelo da experiência 2, criado a partir de conhecimento do domínio e os critérios de avaliação da CWUR, demonstra ser o modelo editorial de melhor qualidade e que melhor representa as áreas sobre as quais as instituições de ensino superior publicam. Analisando os resultados obtidos pelos modelos de ambas as experiências, podemos concluir que as estratégias editoriais identificadas pelo modelo da experiência 2 apresentam um caráter mais constante, onde a mudança de tópico dominante apenas ocorre em 3 instituições, em contraste com as 9 instituições que mudam de tópico dominante nas estratégias da experiência 1. Os resultados obtidos através da análise de sentimento posterior fortalecem a estratégias identificadas pelo modelo da experiência 2, onde as alterações observáveis no sentimento das publicações são apenas por parte da instituição Johns Hopkins University a partir do mês de março e podem ser explicadas pela sua mudança de tópico dominante para a área da Saúde. No caso da experiência 1, deveria ser possível identificar 7 instituições diferentes (Johns Hopkins inclusive) onde estas alterações de sentimento fossem também observáveis, mesmo que em dimensões diferentes, o que não foi o caso para as evoluções de sentimento estudadas.

Sendo assim, descobrimos que as estratégias editoriais da HEI no Twitter podem ser caracterizadas através de seis tópicos-chave de conteúdo ou domínios editoriais (educação, emprego, corpo docente, investigação, saúde e sociedade), e que existem mudanças estratégicas significativas na distribuição de tópicos em algumas HEI após a declaração pandémica da Covid-

19, indicativas de uma mudança na estratégia de conteúdo nas redes sociais devido à atual situação pandémica. Ao longo do período estudado, pudemos observar uma tendência crescente para o tópico de Saúde, com um pico de número de publicações diferenciado e observável em praticamente todas as HEI a partir da declaração do Covid-19 como uma pandemia. Apesar disso, apenas verificámos a alteração do foco estratégico para essa área por parte de uma instituição, Johns Hopkins University. Além disso, pudemos também observar que a pandemia da Covid-19 teve um impacto global positivo na interação pública com as publicações da HEI no meio de comunicação social estudado, sendo que as áreas que revelaram um maior aumento de interação foram Educação e Saúde. Na área da educação, isto pode possivelmente ser devido a um interesse crescente nas ferramentas/conteúdo de ensino partilhado pelas HEI, proveniente das restrições pandémicas que levam ao ensino remoto. Na área da Saúde, pode possivelmente ser devido a um aumento na procura de informação sobre a pandemia (coincidente com o aumento de publicações nessa área) proveniente da incerteza e falta de conhecimento característicos dessa fase inicial que o mundo experienciou. Contudo, este aumento global de interação pode também simplesmente ser devido apenas ao aumento da presença online do público, resultante das restrições pandémicas já mencionadas que levam ao trabalho e ensino remoto.

A área editorial identificada como Imagem no modelo da experiência 1 é uma representação enganadora do conteúdo nela contido. De acordo com o modelo da experiência 2, essa área encontra-se melhor representada através da sua divisão nas áreas de Corpo Docente, Emprego e Sociedade. O termo Imagem sugere um comportamento de autopromoção, e o seu número de publicações na experiência 1 era ao nível de todas as outras áreas. No entanto, na experiência 2, áreas mais relacionadas à auto promoção, como o Corpo Docente (área relativa ao grupo de docentes e às suas distinções) ou Emprego (área relativa aos ex-alunos e os seus cargos atuais) são as duas áreas editoriais menos presentes no conteúdo de uma HEI.

Em relação aos resultados da análise preditiva, concluímos que a informação recolhida não é suficiente para conseguir realizar as tarefas propostas. Na tarefa de previsão do número de retweets de uma publicação, os melhores modelos obtêm um RMSE superior a 8.8 retweets. Num conjunto de publicações com alcance de 0 a 76 retweets, o erro obtido é significativo o suficiente para não considerarmos os modelos obtidos capazes de realizar esta tarefa. Sendo assim, podemos concluir que existem mais fatores do que aqueles presentes nos dados recolhidos que possuem um impacto não negligenciável no número de retweets de uma publicação. Na tarefa de previsão do tópico dominante de uma publicação, os melhores modelos não passam de 44% de exatidão, e por isso também os consideramos incapazes de realizar a tarefa proposta. Mais uma vez, concluímos que existem mais fatores para além das publicações anteriores e alguns metadados das mesmas que definem o conteúdo de uma nova publicação. Apesar dos modelos de previsão obtidos a partir da experiência 1 possuírem ligeiramente melhores resultados nas tarefas propostas, isso não reflete diretamente a qualidade do seu modelo editorial, sendo que a avaliação dos modelos anteriores apenas classifica a sua capacidade de prever o tópico dominante de uma publicação de acordo com as áreas editoriais estabelecidas. Ou seja, a única característica a ser avaliada nesta análise é a capacidade preditiva dos atributos em relação às áreas editoriais obtidas, e não a qualidade do modelo editorial em representar as publicações recolhidas.

Finalmente, é de salientar que, apesar de ter o menor número médio de publicações por dia, Stanford conseguiu atingir alguns dos melhores números de interação, com a média mais alta de retweets por tweet (120) e a 3ª média mais alta de favoritos por tweet. Pelo contrário, apesar de ter o maior número médio de publicações por dia, a Imperial College ficou em 9º lugar em média de retweets e em 7º lugar em média de favoritos, o que constitui um forte argumento para que a frequência de publicações não seja um bom preditor de envolvimento. Para além das conclusões anteriores, a nossa contribuição neste documento é também sob a forma de uma proposta de metodologia geral para a compreensão e medição de estratégias de conteúdo das redes sociais nas Instituições de Ensino Superior.

Como trabalho futuro seria interessante explorar o envolvimento do publico através de uma outra perspetiva, passando pela análise das respostas às publicações em questão. Acreditamos que através respostas às publicações conseguiríamos identificar com mais detalhe, e de forma mais diferenciada, os diferentes tipos de respostas e os sentimentos associados a elas e, deste modo, possuir uma melhor perspetiva do envolvimento associado a cada publicação. Para além disso, seria interessante explorar também o tipo de publicação efetuada, ou seja, se se trata de uma publicação com conteúdo em formato de vídeo, fotografia, status, link, etc.

Parte do trabalho desenvolvido nesta dissertação foi tema de um artigo cientifico aceite pela 21ª Conferência Internacional de Ciências de Computação e as Suas Aplicações (ICCSA)[1], de nome 'Covid-19 Impact on Higher Education Institution's Social Media Content Strategy', e consequentemente incluído na Springer Lecture Notes in Computer Science (DOI: 10.1007/978-3-030-86960-1_49) assim como indexado por Scopus, EI Engineering Index, Thomson Reuters Conference Proceedings Citation Index (incluído no ISI Web of Science), e vários outros serviços de indexação.

[1] https://iccsa.org/

Referências

[1] Wikimedia commons: File:logistic-curve.svg. https://commons.wikimedia.org/wiki/File:Logistic-curve.svg, . Acedido: 09/09/2021.

[2] Wikimedia commons: File:svm separating hyperplanes.png. https://commons.wikimedia.org/wiki/File:Svm_separating_hyperplanes.png, . Acedido: 09/09/2021.

[3] Wikimedia commons: File:multilayerperceptron.png. https://commons.wikimedia.org/wiki/File:MultiLayerPerceptron.png, . Acedido: 09/09/2021.

[4] World university rankings 2019-20. https://cwur.org/2019-20.php, . Acedido: 09/09/2021.

[5] World university rankings methodology. https://cwur.org/methodology/world-university-rankings.php, . Acedido: 09/09/2021.

[6] Twitter api tweet object exemple. https://developer.twitter.com/en/docs/twitter-api/v1/data-dictionary/object-model/tweet, . Acedido: 09/09/2021.

[7] Gensim topic modelling python library. https://radimrehurek.com/gensim/, . Acedido: 09/09/2021.

[8] Glove twitter pre-trained word vector. https://nlp.stanford.edu/data/glove.twitter.27B.zip, . Acedido: 09/09/2021.

[9] Google news pre-trained word vector. https://drive.google.com/file/d/0B7XkCwpI5KDYNlNUTTlSS21pQmM/edit?usp=sharing, . Acedido: 09/09/2021.

[10] Twitter api. https://developer.twitter.com/en/docs/twitter-api, . Acedido: 09/09/2021.

[11] Textblob sentiment analysis python library. https://textblob.readthedocs.io/en/dev/, . Acedido: 09/09/2021.

[12] Actigraph. GT3X+ Activity Monitor. Product specification, Actigraph, 2011 [visited June 2011].

[13] Fouad Nasser A Al Omran and Christoph Treude. Choosing an nlp library for analyzing software documentation: A systematic literature review and a series of experiments. In *2017 IEEE/ACM 14th International Conference on Mining Software Repositories (MSR)*, pages 187–197, 2017. doi:10.1109/MSR.2017.42.

[14] Rania Albalawi, Tet Yeap, and Morad Benyoucef. Using topic modeling methods for short-text data: A comparative analysis. *Frontiers in Artificial Intelligence*, 3, 07 2020. doi:10.3389/frai.2020.00042.

[15] Ethem Alpaydin. *Introduction to Machine Learning (Adaptive Computation and Machine Learning series)*. The MIT Press, August 2014. ISBN: 0262028182.

[16] Sio-Iong Ao, Burghard B. Rieger, and Mahyar Amouzegar. *Machine Learning and Systems Engineering (Lecture Notes in Electrical Engineering Book 68)*. Springer, October 2010. ISBN: 9048194199.

[17] B. Apolloni, A. Ghosh, Alpaslan, and S. F., Patnaik. *Machine Learning and Robot Perception (Studies in Computational Intelligence, 7)*. Springer, September 2005. ISBN: 354026549X.

[18] Christy Ashley and Tracy Tuten. Creative strategies in social media marketing: An exploratory study of branded social content and consumer engagement. *Psychology & Marketing*, 32(1):15–27, 2015. doi:https://doi.org/10.1002/mar.20761.

[19] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, December 2006. ISBN: 0387310738.

[20] David Blei, Andrew Ng, and Michael Jordan. Latent dirichlet allocation. volume 3, pages 601–608, 01 2001.

[21] David M. Blei. Probabilistic topic models. *Commun. ACM*, 55(4):77–84, April 2012. ISSN: 0001-0782. doi:10.1145/2133806.2133826.

[22] G.E.P. Box. Robustness in the strategy of scientific model building. In ROBERT L. LAUNER and GRAHAM N. WILKINSON, editors, *Robustness in Statistics*, pages 201–236. Academic Press, 1979. ISBN: 978-0-12-438150-6. doi:https://doi.org/10.1016/B978-0-12-438150-6.50018-2.

[23] Leo Breiman. Bagging predictors. *Machine Learning*, 24:123–140, 1996.

[24] Jacques Bughin and Michael Chui. The rise of the networked enterprise: Web 2.0 finds its payday. *McKinsey Quarterly*, Dec 2010.

[25] Pew Research Center. Social media use in 2018: Appendix a: Detailed table. Online, March 2018. Acedido: 09/09/2021.

[26] Jonathan Chang, Jordan Boyd-Graber, Sean Gerrish, Chong Wang, and David Blei. Reading tea leaves: How humans interpret topic models. volume 32, pages 288–296, 01 2009.

[27] Vladimir Cherkassky and Filip M. Mulier. *Learning from Data: Concepts, Theory, and Methods, 2nd Edition*. Wiley, September 2007. ISBN: 0521699096.

[28] T. Clausen and P. Jacquet. Optimized Link State Routing Protocol (OLSR). RFC 3626, IETF, October 2003.

[29] Ton J. Cleophas and Aeilko H. Zwinderman. *Machine Learning in Medicine*. Springer, December 2013. ISBN: 9400758243.

[30] John Doe, Jane Doe, and Mr Director. A paper about the universe, life and everything. *Galaxy 2000*, 42:41–43, June 2000. ISSN: 1541-4612. doi:10.1016/j.compedu.2011.12.001.

[31] Issam El Naqa and Martin J. Murphy. *What Is Machine Learning?*, pages 3–11. Springer International Publishing, Cham, 2015. ISBN: 978-3-319-18305-3. doi:10.1007/978-3-319-18305-3_1.

[32] ETSI. ETSI TR 102 732 V1.1.1 (2013-09) Machine-to-Machine Communications (M2M); Use Cases of M2M applications for eHealth, 2013.

[33] Alan H. Fielding. *Machine Learning Methods for Ecological Applications*. Springer, December 2012. ISBN: 1461552893.

[34] Álvaro Figueira. A three-step data-mining analysis of top-ranked higher education institutions' communication on facebook. In *Proceedings of the Sixth International Conference on Technological Ecosystems for Enhancing Multiculturality*, TEEM'18, page 923–929, New York, NY, USA, 2018. Association for Computing Machinery. ISBN: 9781450365185. doi:10.1145/3284179.3284342.

[35] Álvaro Figueira. Uncovering social media content strategies for worldwide top-ranked universities. *Procedia Computer Science*, 18, 10 2018. doi:10.1016/j.procs.2018.10.088.

[36] Yogesh Girdhar, Philippe Giguère, and Gregory Dudek. *Autonomous Adaptive Underwater Exploration using Online Topic Modeling*, pages 789–802. Springer International Publishing, Heidelberg, 2013. ISBN: 978-3-319-00065-7. doi:10.1007/978-3-319-00065-7_53.

[37] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *AISTATS*, 2011.

[38] Yihong Gong and Wei Xu. *Machine Learning for Multimedia Content Analysis*. Springer Nature, August 2007. ISBN: 0387699422.

[39] L. Györfi, G. Ottucsák, and H. Walk. *Machine Learning for Financial Engineering (Advances in Computer Science and Engineering: Texts)*. Imperial College Press, February 2012. ISBN: 1848168136.

[40] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *Introduction To Optimization (translations Series In Mathematics And Engineering)*. Springer, December 2003. ISBN: 0387952845.

[41] Tin Kam Ho. Random decision forests. In *Proceedings of the Third International Conference on Document Analysis and Recognition (Volume 1) - Volume 1*, ICDAR '95, pages 278–282, M, 1995. IEEE Computer Society. ISBN: 0-8186-7128-9.

[42] Liangjie Hong and Brian D. Davison. Empirical study of topic modeling in twitter. In *Proceedings of the First Workshop on Social Media Analytics*, SOMA '10, page 80–88, New York, NY, USA, 2010. Association for Computing Machinery. ISBN: 9781450302173. doi:10.1145/1964858.1964870.

[43] David W. Hosmer and Stanley Lemeshow. *Applied Logistic Regression (Wiley Series in Probability and Statistics)*. Wiley-Interscience Publication, December 2000. ISBN: 0471356328.

[44] Daniel Jurafsky and James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition*. Prentice Hall, February 2000. ISBN: 0130950696.

[45] Hillol Kargupta, Jiawei Han, Philip S. Yu, Rajeev Motwani, and Vipin Kumar. *Next Generation of Data Mining (Chapman & Hall/Crc Data Mining and Knowledge Discovery Series)*. Chapman and Hall/CRC, October 2019. ISBN: 0367386054.

[46] R. Kohavi and D. Wolpert. Bias plus variance decomposition for zero-one loss functions. In *ICML*, 1996.

[47] L.I. Kuncheva and C.J. Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning*, 51:181–207, 2003.

[48] Daniel D. Lee and H. Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, October 1999. ISSN: 0028-0836. doi:10.1038/44565.

[49] L. Liu, L. Tang, and W. et al. Dong. An overview of topic modeling and its current applications in bioinformatics. *SpringerPlus*, 5, March 2016.

[50] James D. Malley. *Statistical Learning for Biomedical Data (Practical Guides to Biostatistics and Epidemiology)*. Cambridge University Press, February 2011. ISBN: 0521699096.

[51] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013.

[52] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013.

[53] Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751, Atlanta, Georgia, June 2013. Association for Computational Linguistics.

[54] Tom M. Mitchell. The need for biases in learning generalizations. Technical report, Rutgers University, New Brunswick, NJ, 1980.

[55] Tom M. Mitchell. *Machine Learning*. McGraw-Hill Education, March 1997. ISBN: 0070428077.

[56] Mitchell, Michael and Leachman, Michael and Masterson, Kathleen. A Lost Decade in Higher Education Funding State Cuts Have Driven Up Tuition and Reduced Quality. State budget report, Actigraph, 2017 [visited September 2021].

[57] Sushmita Mitra, Sujay Datta, Theodore Perkins, and George Michailidis. *Introduction to Machine Learning and Bioinformatics (Chapman & Hall/ CRC Computer Science & Data Analysis)*. Chapman and Hall/CRC, September 2019. ISBN: 0367387239.

[58] NetMarketShare. Mobile/tablet operating system market share. Online, December 2016. Dezembro de 2016.

[59] Jakob Nielsen. Usability inspection methods. In *Conference companion on Human factors in computing systems*, pages 413–414. ACM, 1994.

[60] Jakob Nielsen. *Usability engineering*. Morgan Kaufmann, 1994. ISBN: 0125184069.

[61] Jakob Nielsen and Thomas K Landauer. A mathematical model of the finding of usability problems. In *Proceedings of the INTERACT'93 and CHI'93 conference on Human factors in computing systems*, pages 206–213. ACM, 1993.

[62] Nonin Medical Inc. Onyx II® Model 9560 Bluetooth® Fingertip Oximeter. OEM Specification and Technical Information 6470-000-01, Nonin Medical Inc, 2008 [visited June 2011].

[63] Luciana Oliveira and Álvaro Figueira. Benchmarking analysis of social media strategies in the higher education sector. 10 2015. doi:10.1016/j.procs.2015.08.628.

[64] Luciana Oliveira and Álvaro Figueira. Improving the benchmarking of social media content strategies using clustering and kpi. *Procedia Computer Science*, 121:826–834, 2017. ISSN: 1877-0509. CENTERIS 2017 - International Conference on ENTERprise Information Systems / ProjMAN 2017 - International Conference on Project MANagement / HCist 2017 - International Conference on Health and Social Care Information Systems and Technologies, CENTERIS/ProjMAN/HCist 2017. doi:https://doi.org/10.1016/j.procs.2017.11.107.

[65] David Opitz and Richard Maclin. Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research*, 11:169–198, 1999.

[66] Pentti Paatero and Unto Tapper. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5:111–126, 1994.

[67] Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October 2014. Association for Computational Linguistics. doi:10.3115/v1/D14-1162.

[68] Adam Peruta and Alison Shields. Social media in higher education: understanding how colleges and universities use facebook. *Journal of Marketing for Higher Education*, 27:1–13, 07 2016. doi:10.1080/08841241.2016.1212451.

[69] S. Madeh Piryonesi and Tamer E. El-Diraby. Role of data analytics in infrastructure asset management: Overcoming data size and quality problems. *Journal of Transportation Engineering, Part B: Pavements*, 146(2):04020022, 2020. doi:10.1061/JPEODX.0000175.

[70] R. Polikar. Ensemble based systems in decision making. *IEEE Circuits and Systems Magazine*, 6(3):21–45, 2006. doi:10.1109/MCAS.2006.1688199.

[71] Boris T. Polyak. *Introduction To Optimization (translations Series In Mathematics And Engineering)*. Optimization Software, December 1987. ISBN: 0911575146.

[72] Veronica Popovici and Ramona Nicoleta Buna. Web 2.0 tools in the context of integrated communication: New technologies revolutionizing the business environment. In *2009 International Conference on Management and Service Science*, pages 1–4, 2009. doi:10.1109/ICMSS.2009.5305338.

[73] David M. W. Powers. Evaluation: From precision, recall and f-measure to roc, informedness, markedness & correlation. *Journal of Machine Learning Technologies*, 2:37–73, 2011.

[74] J.R. Quinlan. Induction of decision trees. *Machine Learning*, 1:81–106, 1986.

[75] Prajit Ramachandran, Barret Zoph, and Quoc V. Le. Searching for activation functions, 2017.

[76] Michael Röder, Andreas Both, and Alexander Hinneburg. Exploring the space of topic coherence measures. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, WSDM '15, page 399–408, New York, NY, USA, 2015. Association for Computing Machinery. ISBN: 9781450333177. doi:10.1145/2684822.2685324.

[77] L. Rokach. Ensemble-based classifiers. *Artificial Intelligence Review*, 33:1–39, 2010.

[78] D. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning internal representations by error propagation. 1986.

[79] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Pearson, December 2009. ISBN: 0136042597.

[80] A. Samuel. Some studies in machine learning using the game of checkers. *IBM J. Res. Dev.*, 3:210–229, 1959.

[81] Carson Sievert and Kenneth Shirley. Ldavis: A method for visualizing and interpreting topics. 06 2014. doi:10.13140/2.1.1394.3043.

[82] Peter Sollich and Anders Krogh. Learning with ensembles: How over-fitting can be useful. In *Proceedings of the 8th International Conference on Neural Information Processing Systems*, NIPS'95, page 190–196, Cambridge, MA, USA, 1995. MIT Press.

[83] William Strunk. *The elements of style*. Bartleby, July 1999. ISBN: 1-58734-060-7.

[84] Symeon Symeonidis, Dimitrios Effrosynidis, and Avi Arampatzis. A comparative evaluation of pre-processing techniques and their interactions for twitter sentiment analysis. *Expert Systems with Applications*, 110:298–310, 2018. ISSN: 0957-4174. doi:https://doi.org/10.1016/j.eswa.2018.06.022.

[85] Vladimir N. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, December 1998. ISBN: 0471030031.

[86] Ulrike von Luxburg and Bernhard Schölkopf. Statistical learning theory: Models, concepts, and results. In Dov M. Gabbay, Stephan Hartmann, and John Woods, editors, *Inductive Logic*, volume 10 of *Handbook of the History of Logic*, pages 651–706. North-Holland, 2011. doi:https://doi.org/10.1016/B978-0-444-52936-7.50016-1.

[87] C. White. Consolidating, accessing, and analysing unstructured data. 2005.

[88] Joe Wolfe. How to write a phd thesis. Online, November 2006. Acedido hoje.

[89] X. Wu, V. Kumar, Ross Quinlan, and J. et al. Top 10 algorithms in data mining. *Knowledge and Information Systems*, 14:1–37, 2008.

[90] Elias Yaacoub and Adnan Abu-Dayya. Enhancing the QoS of real-time video streaming over LTE MBMS using D2D communications. In *Proceedings of the 8h ACM symposium on QoS and security for wireless and mobile networks*, pages 41–46. ACM, December 2012.

[91] Zheng Rong Yang. *Machine Learning Approaches to Bioinformatics (Science, Engineering, and Biology Informatics)*. World Scientific Publishing Company, May 2010. ISBN: 981428730X.

[92] Jun Yu and Dacheng Tao. *Modern Machine Learning Techniques and Their Applications in Cartoon Animation Research*. Wiley-IEEE Press, December 2013. ISBN: 1118115147.

[93] İrem Eren Erdoğmuş and Mesut Çiçek. The impact of social media marketing on brand loyalty. *Procedia - Social and Behavioral Sciences*, 58:1353–1360, 2012. ISSN: 1877-0428. 8th International Strategic Management Conference. doi:https://doi.org/10.1016/j.sbspro.2012.09.1119.