Deteção de contas falsas na rede social *Instagram*

Tópicos de Inteligência Artificial

Carolina Pires (2021237824), Eduardo Cardoso (2021226956), Matilde Reis (2021237887)

Faculdade de Ciências e Tecnologia da Universidade de Coimbra

Licenciatura em Engenharia e Ciência de Dados

carolinapires2003@gmail.com, eduardosncardoso@gmail.com, matilde.martins.reis@gmail.com

Abstract

Neste trabalho propomos fazer uma análise detalhada sobre a deteção de contas falsas no Instagram baseada num algoritmo de machine learning. A criação de contas falsas é um dos problemas mais significativos nas redes sociais que são usadas para diferentes finalidades. A deteção destas contas é crucial para tentar preservar um ambiente saudável online já que a sua criação está altamente relacionada com o cibercrime, com os golpes e prejuízos e com o roubo de identidade. Este estudo está também relacionado com o engajamento falso no instagram pois contas falsas normalmente têm poucos seguidores, poucas publicações e bastantes perfis a seguir. O método utilizado foi um modelo de redes neuronais artificiais (ANN) que para o conjunto de dados obteve 88% de accuracy.

Palavras-chave – contas falsas, machine learning, redes neuronais, redes sociais, instagram

1. Introdução

Hoje em dia, a conectividade e a acessibilidade estão a chegar a mais países do que nunca. Elas têm o potencial de transformar a vida de crianças e adolescentes, dando-lhes acesso a oportunidades educacionais, culturais e económicas que antes eram inimagináveis. Porém, com grande frequência, crianças e adolescentes não podem usufruir dessas oportunidades, uma vez que a internet também é um espaço no qual os vulneráveis estão expostos ao risco de sofrer sérios danos, como o bullying cibernético, a exposição a conteúdos inadequados, o assédio grave, o aliciamento online, o recrutamento por movimentos extremistas, a exploração e o abuso sexual. É, portanto, responsabilidade do mundo adulto encontrar uma forma de minimizar e impedir esses riscos e perigos. Como exemplo desses perigos na internet, Pedro Vicente [22] afirma: "Faço-me passar por uma rapariga de 16 anos e tento meter conversa e pedir fotografias de miúdos de 16 anos. Quando as plataformas descobrem, normalmente bloqueiam o perfil. Mas logo a seguir faço outro perfil com o mesmo comportamento".

Atualmente, com o constante crescimento das redes sociais e com o papel que desempenham na sociedade, tanto a nível social como de negócios, torna-se importante compreender alguns problemas que se têm vindo a identificar como, por exemplo, as atividades relacionadas com o cibercrime que está muito associado com a criação de contas falsas e com a sua utilização em atividades ilícitas e potencialmente prejudiciais para os seus utilizadores (por exemplo, [10]). Uma vez que as redes sociais fazem cada vez mais parte das nossas vidas, é quase impossível não nos termos deparado com pelo menos um perfil de alguém que não condiz com a realidade. Eles diferenciam-se entre perfis falsos e perfis genuínos com base em recursos visíveis como o número de seguidores, o número de amigos, o número de gostos, o número de publicações, entre outros [5].

A Organização Mundial da Saúde (OMS) estima que, a cada ano, 200 milhões de crianças e adolescentes são abusados sexualmente [14]. Além disso, cada vez mais, grande parte desse tipo de violência ocorre online. As crianças consomem diariamente mais de 4 horas da sua vida a realizar qualquer atividade nos seus perfis. A frequência com que acedem às redes sociais, na grande maioria, é de dez ou mais vezes por dia. Aproximadamente 75% dos adolescentes possuem telemóveis, 25% utilizam-nos para as redes sociais, 54% para mensagens de texto e 24% para mensagens instantâneas.

Detetar estas contas de perfis falsos é interessante e importante para podermos agir corretamente numa situação real pois existem casos em que os "fakes" foram responsáveis por marcar a vida das vítimas, deixando cicatrizes [4].

A criação de contas falsas está a ganhar uma dimensão gigante. Em 2019, o Facebook removeu cerca de 6,5 milhões de contas falsas por dia, e esse número tende a aumentar diariamente [2]. Estas contas, geralmente, são criadas por humanos ou bots com a intenção de espalhar rumores,

violação de dados, roubo de identidade, levar informação falsa e induzir as pessoas a golpes e outros tipos de prejuízos.

De acordo com a APAV (Associação Portuguesa de Apoio à Vítima), cerca de um terço dos utilizadores da internet de todo o mundo são crianças e jovens até aos 18 anos [3]. Este fenómeno cresceu durante o período de isolamento provocado pelo COVID-19 e é uma prática que pode ser prejudicial. A internet pode mudar a vida das crianças e dos jovens se for utilizada de forma adequada, especialmente das mais isoladas. De facto, o uso das redes sociais pode-lhes dar uma ideia de oportunidades. Contudo, esse uso está cheio de riscos e dá uma visão às crianças de um mundo irreal tornando-as mais suscetíveis a diferentes formas de violência, por exemplo, um ato online denominado "grooming", que consiste num processo de manipulação em que uma pessoa adulta inicia uma abordagem não-sexual de forma a convencer uma criança ou jovem a encontrar-se consigo com o objetivo de praticar o abuso sexual. Muitas delas caem na tentação de marcar encontros online, de fornecer informações pessoais como dados de pagamento, morada ou até mesmo dados telefónicos levando, deste modo, a prejuízos financeiros e psicológicos (por exemplo, [3, 7]). É, portanto, um dos maiores riscos que os jovens correm na internet, alerta Tito Morais, que há 10 anos trabalha na segurança "online" de jovens. Tito Morais é o responsável pelo projeto 'Miúdos Seguros na Net' [1] que tem como objetivo minimizar os riscos e maximizar os benefícios, contendo dicas e formas de como devemos proteger os jovens destes riscos.

A deteção de contas falsas já é possível através da implementação de algoritmos e métodos de machine learning [11, 12, 13]. Em [16] o método k-means foi utilizado para detetar contas falsas em redes sociais. No entanto, esse algoritmo permite apenas dividir os dados em grupos sendo, por isso, usado em problemas não supervisionados. Dado o problema de classificação de contas em perfis falsos ou não ser supervisionado, o k-means não é o método mais adequado [17]. Por outro lado, modelos como as redes neuronais artificiais (ANN) constituem um método supervisionado sendo, por isso, mais compatível com o problema em estudo.

No trabalho [18] foram utilizadas 91 imagens em que 80 delas eram de treino e 11 eram de teste. A precisão do k-means foi de 27,27%, enquanto a precisão do algoritmo de ANN foi de 54,54%. Neste caso, as redes neuronais artificiais têm uma melhor precisão do que o k-means. Há também alguns trabalhos em que o Instagram é estudado do ponto de vista das interações falsas. O Instagram tornou-se uma das principais plataformas sociais que atingiu cerca de 1 bilião de utilizadores ativos mensais com 4,2 biliões de mensagens diariamente e 2 milhões de anunciantes mensais (por exemplo, [9]). Por conseguinte, é crucial identificar perfis falsos para preservar o ambiente saudável numa importante plataforma social. Tenta-se determinar os gostos falsos e a principal preocupação é estimar qual é a probabilidade de um utilizador poder gostar do post de outro utilizador.

O trabalho a analisar e implementar contém código e datasets que se encontram em [19]. O método utilizado para detetar contas falsas foi um modelo de redes neuronais artificiais (ANN). O algoritmo está dividido em três partes: limpeza e exploração dos dados; pré-processamento e treino do modelo; avaliação do modelo.

Para avaliar os resultados foi usada uma matriz de confusão. A partir desta matriz é possível obter algumas métricas como a sensibilidade e a especificidade, que permitem quantificar a performance do modelo.

Os datasets são utilizados para que os testes permitam ter boas indicações sobre o desempenho do algoritmo, designadamente ao nível do número de falsos positivos, falsos negativos, verdadeiros positivos e verdadeiros negativos.

O modelo ANN é eficiente. Logo, é esperado que se consiga classificar corretamente a maioria das contas analisando o perfil de cada indivíduo de forma a identificar se contém um perfil de conta falsa ou não.

Neste relatório, focamos a situação atual de crianças e adolescentes no ambiente online. Quais são os riscos e perigos que eles enfrentam? Como é que esses riscos os impedem de ter acesso às oportunidades mencionadas? E o que podemos fazer para assegurar que isso não aconteça? O nosso principal objetivo consiste em refletir sobre como priorizar a segurança de todas as crianças e todos os adolescentes que enfrentam algum problema online.

No restante documento apresentamos uma análise detalhada sobre o trabalho implementado para a deteção de contas falsas do Instagram usando algoritmos de machine learning, desde os métodos utilizados até aos resultados obtidos.

2. Trabalho relacionado

A deteção de contas falsas não é um tema inédito. Já existem diversos trabalhos publicados que abordam e utilizam diferentes métodos para realizar essa identificação. Em [16], como já foi referido, a classificação de contas de redes sociais foi possível através do k-means, que permitiu agrupar os dados em dois grupos, com base nas suas características: contas reais ou contas falsas. No entanto, este algoritmo não constitui o método mais adequado para problemas supervisionados.

Em [6] o estudo desenvolvido tinha como objetivo a deteção de contas falsas e automatizadas. Vários métodos foram utilizados, desde métodos tradicionais como Naïve Bayes, regressão logística e suport-vector machine (SVM) até modelos baseados em redes neuronais, o qual será utilizado neste trabalho. Os métodos SVM e ANN obtiveram os melhores resultados, com um F1-score de 86% e 95%,

respetivamente. Assim, dado estes valores, espera-se que neste trabalho o modelo ANN também alcance bons resultados.

O trabalho [20] aborda igualmente o tema da deteção de contas falsas, utilizando o método gradient boosting com árvores de decisão contendo três atributos. Esse algoritmo é implementado com o objetivo de ultrapassar métodos como Naïve Bayes ou SVM que, com o aumento da criação de contas falsas, têm-se tornado cada vez mais ineficientes.

Em [21] os métodos usados foram SVM, redes neuronais (NN) e um método desenvolvido, SVM-NN, que combina os dois anteriores e usa um menor número de atributos. Para os dados de treino utilizados nesse trabalho, este algoritmo classificou corretamente 98% das contas. Também o trabalho [8] obteve uma precisão de 98% com o uso de redes neuronais. Estes resultados demonstram a eficiência de métodos baseados em redes neuronais.

3. Materiais

O projeto foi implementado em Python e foram utilizadas as bibliotecas *pandas, matplotlib, numpy, seaborn, tensorflow e sklearn*.

O dataset utilizado pelos autores no desenvolvimento deste projeto [19] é composto por dois ficheiros csv sendo um deles formado pelos dados de treino (576 casos) e o outro pelos dados de teste (120 casos). Todos os dados são caracterizados pelos seguintes atributos:

- **profile pic** - 1 se a conta tem foto de perfil, 0 caso contrário.
- **nums/length username** - proporção de números presentes no username da conta.
- **fullname words** - número de palavras no nome da conta.
- **nums/length fullname** - proporção de números presentes no nome da conta.
- **name==username** - 1 se o nome da conta é igual ao username, 0 caso contrário.
- **description length** - número de caracteres presentes na descrição da conta.
- **external URL** - 1 se a conta possui um URL para um site externo, 0 caso contrário.
- **private** - 1 se a conta é privada, 0 caso contrário.
- **#posts** - número de publicações da conta.
- **#followers** - número de seguidores da conta.
- **#follows** - número de páginas que a conta segue.
- **fake** - 1 se a conta é falsa, 0 caso contrário.

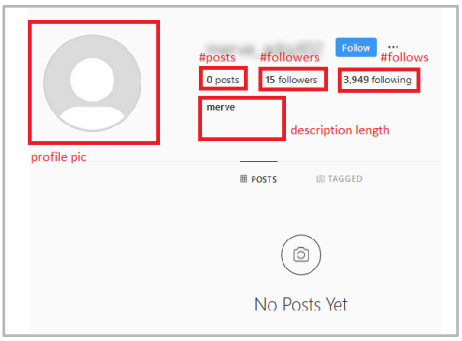

Figura 1: Exemplo de uma suspeita de conta falsa a partir do dataset utilizado.

As contas falsas são, desde logo, contas que são utilizadas para aumentar a métrica de popularidade de outros utilizadores. Assim, a ausência de imagem de perfil e *usernames* estranhos são características comuns deste tipo de contas.

A primeira etapa realizada pelos autores do trabalho foi uma análise exploratória de dados (EDA). Nesta parte eles determinaram algumas características dos vários atributos dos dados (média, desvio padrão, etc.), verificaram se existiam valores nulos e recorreram a histogramas, gráficos de barras e uma matriz de correlação para visualizar os atributos dos dados e as relações entre eles.

A característica **private** considera o valor 1 se a conta se apresenta privada e 0 caso contrário. Como podemos ver na figura 2 (que eles obtiveram através da linha de código *sns.countplot(x = instagram_df_train['private'])*) existem no dataset de treino cerca de 350 contas públicas e aproximadamente 230 contas privadas.

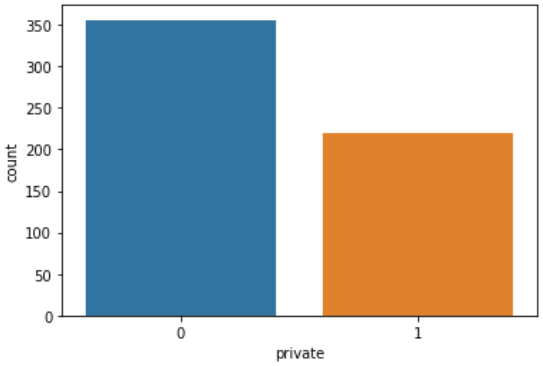

Figura 2 - Gráfico de barras referente à característica **private.**

Em relação à característica **fake** retorna-se 1 se a conta é falsa e 0 se não é. Ao analisarmos o gráfico da figura 3 (presente na linha de código *sns.countplot(x = instagram_df_train['fake'])*) percebemos que existe aproximadamente a mesma quantidade de contas falsas e de contas verdadeiras.

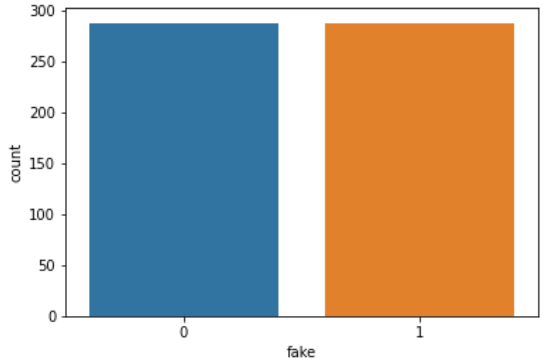

Figura 3 - Gráfico de barras referente à característica **fake**.

A característica **profile pic** possui o valor 1 se a conta apresenta foto de perfil e 0 caso não apresenta.

Na figura 4 apresentada abaixo (obtida através da linha de código *sns.countplot(x = instagram_df_train['profile pic'])*) reparamos que a maior parte das contas em análise não tem foto de perfil, cerca de 400.

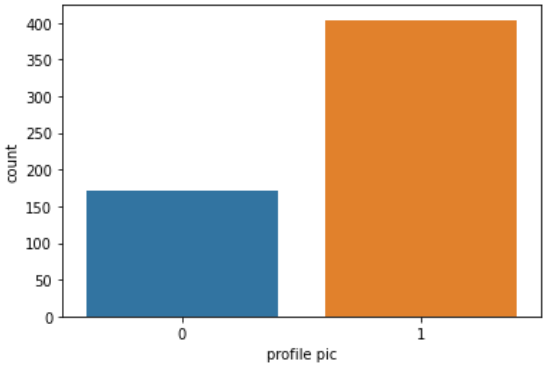

Figura 4 - Gráfico de barras referente à característica **profile pic.**

Os três gráficos anteriores foram também obtidos para os dados de treino.

A característica **nums/length username** representa a proporção de números presentes no username da conta.

De acordo com o histograma da figura 5 percebemos que, considerando apenas esse atributo, existe uma grande diferença entre o número de contas. A uma menor proporção de números no username corresponde um elevado número de contas (cerca de 300) e a uma maior proporção corresponde um reduzido número de contas (cerca de 10). Eles obtiveram este gráfico através da linha:
sns.histplot(instagram_df_train['nums/length username']).

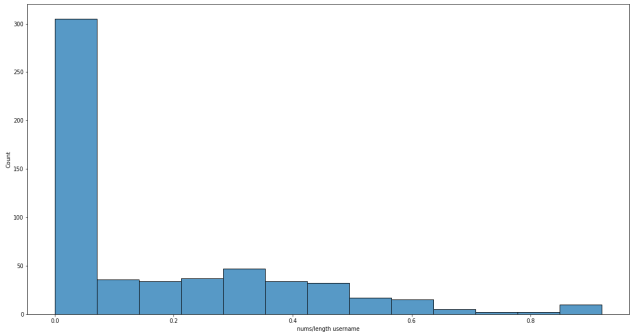

Figura 5 - Histograma relacionado com a característica **nums/length username.**

Por fim, fizeram a matriz de correlação das variáveis com o objetivo de ver o quão relacionadas elas estão entre si. Analisando essa matriz, presente na figura 6, verificamos que as cores mais claras são as que apresentam uma maior correlação pois representam os valores de correlação de todos os atributos com eles próprios, apresentando um valor igual a 1 que implica uma elevada correlação positiva.

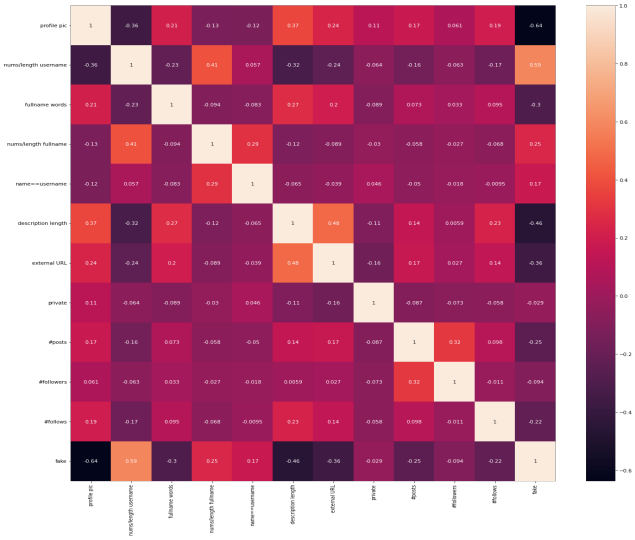

Figura 6 - **Matriz de Correlação**

4. Métodos

A parte dos métodos envolve a configuração e procedimento experimental. Trata-se de uma etapa fundamental que compreende a realização de análises e modelações que serão determinantes para a qualidade final dos dados que serão analisados e para a resolução do problema.

Os autores começaram por dividir os dados de treino e de teste em dois conjuntos: o atributo **fake** foi separado das outras variáveis, para obter um único vetor com a saída desejada e os atributos restantes passaram a constituir o conjunto das variáveis de entrada. Estas ações são realizadas,

respetivamente, nas seguintes linhas de código (correspondentes aos dados de treino):

Y_train = instagram_df_train['fake']
X_train = instagram_df_train.drop(columns = ['fake'])

De seguida realizaram a transformação dos dados. As variáveis de entrada foram centradas e reduzidas - subtrair a média e dividir pelo desvio padrão - e o vetor da saída foi transformado numa matriz de classificação binária: cada valor inicial foi transformado na lista [1 0] ou [0 1], caso esse valor seja 0 ou 1, respetivamente. Estes dois processos permitem que o modelo funcione de forma correta e mais eficiente [23, 24], e são efetuados nas linhas de código abaixo (correspondentes aos dados de treino):

X_train = scaler_x.fit_transform(X_train)
y_train = tf.keras.utils.to_categorical(Y_train, num_classes = 2)

O passo seguinte consistiu na criação do modelo. Eles começaram por criar uma rede neuronal (*model = Sequential()*) e adicionaram 8 camadas, através do método *add*. 5 camadas são do tipo *Dense*, ou seja, cada neurônio da camada seguinte recebe os outputs dos neurónios anteriores, enquanto as restantes 3 são do tipo *Dropout*, isto é, alguns outputs aleatórios calculados na camada anterior passam a ter o valor 0, o que permite prevenir o overfitting [25].

A função de ativação '*relu*' presente, por exemplo, em *model.add(Dense(150, activation='relu'))* indica que um neurônio da camada só é ativado se o valor que recebe for positivo; já a função *'softmax'* presente em *model.add(Dense(2,activation='softmax'))* diz-nos que esta camada recebe o valor anterior e transforma-o em 0 ou 1. Esta camada apresenta apenas 2 neurónios, cada um correspondente a um desses números.

Após este desenvolvimento, eles configuraram o modelo através de *model.compile(optimizer = 'adam', loss = 'categorical_crossentropy', metrics = ['accuracy'])*. O otimizador *'adam'* permite implementar o algoritmo do gradiente decrescente, a função *loss* diz-nos se o modelo está ou não "baralhado" nas suas decisões (avalia o quão próximo dos dados de treino está o modelo) e o parâmetro *metrics* corresponde às medidas que serão utilizadas para avaliar o modelo durante o treino/validação (neste caso a medida utilizada é a *accuracy* - nº total de acertos).

A seguir executaram o modelo, utilizando a função *model.fit(X_train, y_train, epochs = 50, verbose = 1, validation_split = 0.1)*. Foram realizadas 50 iterações (*epochs = 50*) e em cada uma delas foram usados 10% dos dados de treino para validação (*validation_split = 0.1*).

Por fim, após a execução do modelo calcularam os valores de saída previstos para os dados de teste, através de *model.predict(X_test)*. A comparação destes valores com os reais permitiu avaliar o funcionamento e a eficiência do modelo.

5. Resultados

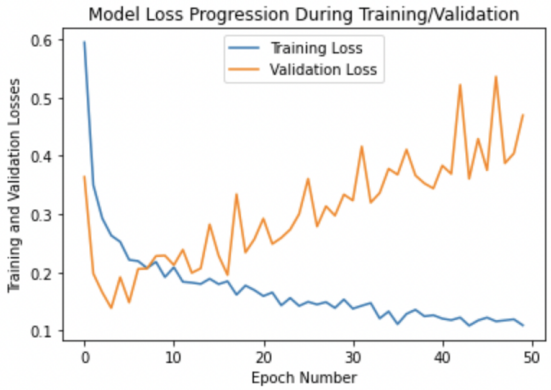

Figura 7 - **Loss do modelo** durante o treino e validação

Na deteção de contas falsas, para comparar e testar a eficácia das técnicas implementadas, a *Precision,* o *Recall* e o *F1-Score* foram usados como métricas de avaliação, conforme indicado e apresentado na figura 8.

	precision	recall	f1-score	support
0	0.87	0.90	0.89	60
1	0.90	0.87	0.88	60
accuracy			0.88	120
macro avg	0.88	0.88	0.88	120
weighted avg	0.88	0.88	0.88	120

Figura 8 - **Métricas de avaliação**

Observamos que a *Precision* apresenta 87% para contas verdadeiras e 90% para contas falsas. O *Recall* apresenta 90% para a frequência de aparecer uma conta verdadeira, e 87% para a frequência de aparecer uma conta falsa. Já o *F1-score* apresenta 89% para contas verdadeiras e 88% para contas falsas.

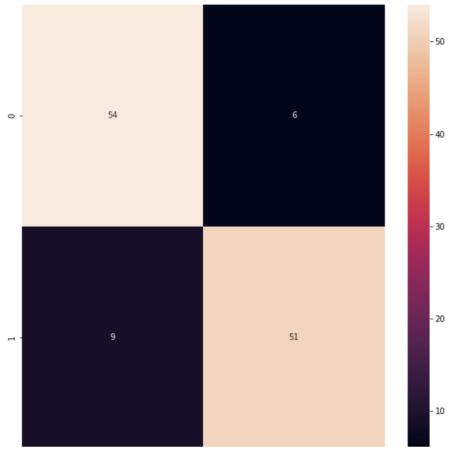

Figura 9 - **Matriz** de confusão

Na matriz de confusão presente na figura 9, podemos observar que o termo TP (verdadeiro positivo) apresenta o valor 51, FP (falso positivo) apresenta 6, FN (falso negativo) apresenta 9 e, por fim, o termo TN (verdadeiro negativo) apresenta 54.

6. Discussão

Podemos observar no gráfico da figura 7, na função "Validation Loss", a presença de overfitting. Observa-se que durante o treino deste modelo há um bom desempenho (valor de loss baixo), porém, durante a validação, o resultado não é tão bom (valor de loss elevado).

A validação apresentada nesse gráfico não corresponde aos dados de teste, mas sim aos dados de treino, pois, como foi referido antes, uma pequena percentagem desses dados (10%) foi utilizada para validar.

Neste caso, o overfitting pode ser causado pela taxa de *dropout* utilizado na criação do modelo. O valor de *dropout* utilizado pelos autores foi 0.3, o que significa que a probabilidade de treinar cada neurónio é 0.3. Este número é muito baixo, dado que o valor de *dropou*t ideal é entre 0.5 e 0.8 [29]. Uma alteração no código para um valor no intervalo ideal poderia resolver este problema.

No entanto, como se pode observar nas figuras 8 e 9, a validação do modelo utilizando os dados de teste permitiu obter resultados elevados (precisão média de 89%). Assim, o problema anterior não parece influenciar a performance do modelo, embora os resultados pudessem ser ligeiramente superiores se fosse corrigido.

Os termos TP, TN, FP e FN correspondem a Verdadeiro Positivo, Verdadeiro Negativo, Falso Positivo e Falso Negativo, respetivamente. Na figura 10 pode-se observar as fórmulas das métricas de avaliação relativamente a estes termos.

$$
\begin{aligned}
precision &= \frac{TP}{TP + FP} \\
recall &= \frac{TP}{TP + FN} \\
F1 &= \frac{2 \times precision \times recall}{precision + recall} \\
accuracy &= \frac{TP + TN}{TP + FN + TN + FP}
\end{aligned}
$$

Figura 10 - **Fórmulas** das métricas de Avaliação

Tal como se pode ver representado na Figura 10, a fórmula da *Precision* é o resultado da divisão dos Verdadeiros Positivos pela soma dos Verdadeiros Positivos com Falsos Positivos.

A *Precision* é utilizada para indicar a percentagem de contas identificadas corretamente como verdadeiras, de entre todas as contas classificadas como verdadeiras (independentemente se o são ou não na realidade), ou seja, o modelo conseguiu acertar com precisão que 87% das contas classificadas como verdadeiras eram realmente verdadeiras. Na fórmula, o dividendo é o número de contas classificadas corretamente como verdadeiras e o divisor é a soma do número de contas classificadas como verdadeiras (e que o são na realidade) com o número de contas que foram classificadas como verdadeiras, mas na realidade são falsas.

A fórmula de *Recall* mede a capacidade do modelo de detetar amostras positivas, ou seja, quando uma conta verdadeira é classificada como sendo verdadeira. Neste caso, o valor obtido é 90%.

O *F1-Score* combina a *Precision* e o *Recall* numa única métrica a fim de trazer um número único que determine a qualidade geral do modelo, calculando a média harmónica entre os dois. É uma medida melhor que a *Accuracy*, principalmente em casos onde falsos positivos e falsos negativos possuem impactos diferentes para seu modelo [26, 27, 28]. Permite verificar se os dados utilizados estão ou não balanceados.

A *Accuracy* indica uma performance geral do modelo, ela mede quantas observações, tanto positivas como negativas, foram classificadas corretamente. Ou seja, quantas contas falsas foram classificadas como falsas e quantas contas verdadeiras foram classificadas como verdadeiras. Neste caso, o seu valor é de 88% [27].

Na figura 9, apresentamos uma matriz de confusão que mostra as frequências de classificação para cada classe do modelo. Observamos que TN é o termo que apresenta um valor mais elevado dos outros, o que significa que, em geral, as contas verdadeiras foram classificadas corretamente pelo

modelo. O termo TP apresenta um valor também elevado, ou seja, também a maioria das contas com a classe que estamos à procura (classe 1 - conta falsa) foi prevista de forma correta.

7. Conclusão

É cada vez mais importante dotar as pessoas e as empresas de ferramentas e informação acerca dos perigos da Internet. Nos últimos tempos, a existência de perfis falsos nas redes sociais tem vindo a crescer drasticamente.

Neste relatório analisamos a implementação de um trabalho relacionado à deteção das contas falsas, que levam a um falso envolvimento no Instagram. Essa deteção é estudada como um problema de classificação binária, ou seja, se estivermos perante a presença de um perfil falso retornamos o valor 1, caso contrário retornamos o valor 0.

Foram elaborados vários testes que permitiram demonstrar a validade e viabilidade do modelo apresentado para a deteção desses perfis, permitindo atingir uma precisão de 87% para contas verdadeiras e de 90% para contas falsas. O modelo desenvolvido baseia-se em redes neuronais, pelo que podemos concluir, como vimos anteriormente (secção 2), que este método possui uma elevada eficácia na resolução de problemas de *machine learning*.

Referências

[1] Tito de Morais MiudosSegurosNa.Net. http://www.miudossegurosna.net

[2] Tchilian, Felipe (2021) "Entenda sobre contas falsas nas redes sociais e como se proteger" in ClearSale. https://blogbr.clear.sale/contas-falsas

[3] "Violência Sexual Online" in APAV. https://apav.pt/care/index.php/informacao-para-adult-s/violencia-sexual-online

[4] Vieira, Nathan (2020) "O que leva uma pessoa a criar um perfil fake?" in Canaltech. https://canaltech.com.br/redes-sociais/por-que-as-pessoas-fazem-perfis-fakes-161172/

[5] "Como Saber se uma Conta do Instagram é Falsa" in wikiHow. https://pt.wikihow.com/Saber-se-uma-Conta-do-Instagram-é-Falsa

[6] Cagatay Akyon, Fatih; Kalfaoglu, Esat (2022) "Instagram Fake and Automated Account Detection" (pdf). https://arxiv.org/pdf/1910.03090v3.pdf

[7] Oliveira, Israel; Scipioni, Manuela (2019) "Os riscos dos perfis fakes nas redes sociais" in Avoador. https://avoador.com.br/pagina-central/perfis-fakes-de-empresas-surgem-no-instagram/

[8] "Fake Social Media Profile Detection" in Wiley Online Library. https://onlinelibrary.wiley.com/doi/abs/10.1002/9781119769262.ch11

[9] Day, Andy (2020) "Instagram's Problem With Fake Users Might Be Much Bigger - and Far Darker - Than You Think" in Fstoppers. https://fstoppers.com/originals/instagrams-problem-fake-users-might-be-much-bigger-and-far-darker-you-think-485579

[10] Morais do Amaral Campos, Bárbara (2021) "Os efeitos negativos das redes sociais na adolescência" (pdf). https://ubibliorum.ubi.pt/bitstream/10400.6/11317/1/8208_17628.pdf

[11] "Social Networks Fake Profiles Detection Using Machine Learning Algorithms" in SpringerLink. https://link.springer.com/chapter/10.1007/978-3-030-37629-1_3

[12] "Fake Account Detection Using Machine Learning" in SpringerLink. https://link.springer.com/chapter/10.1007/978-981-15-5258-8_73

[13] Van Der Walt, Estée; Eloff, Jan "Using Machine Learning to Detect Fake Identities: Bots vs Humans" in IEEE Xplore. https://ieeexplore.ieee.org/abstract/document/8265147

[14] "Segurança Online de Crianças e Adolescentes" in Childhood Brasil. https://www.childhood.org.br/seguranca-online

[15] "Segurança online de crianças e adolescentes: minimizar o risco de violência, abuso e exploração sexual online", Unesco. https://unesdoc.unesco.org/ark:/48223/pf0000374356

[16] "Fake profile detection in social media using image processing and machine learning". http://dspace.bracu.ac.bd/xmlui/handle/10361/15002

[17] "K-Means Clustering Algorithm: Applications, Types, and How Does It Work? by Maynak Banoula. https://www.simplilearn.com/tutorials/machine-learning-tutorial/k-means-clustering-algorithm

[18] Comparison Analysis of the Artificial Neural Network Algorithm and K-Means Clustering in Gorontalo Herbal

Plant Image Identification System by Yulita Salim, Mukhlisulfatih Latief, Novri Kandowangko, Rampi Yusuf. https://ieeexplore.ieee.org/document/8878665

[19] "Fake Instagram Profile Detection using ANN". https://github.com/DURGESH716/Fake-Instagram-Profile-Detection

[20] S. P. Maniraj, Harie Krishnan G, Surya T, Pranav R (2019) "Fake Account Detection using Machine Learning and Data Science" (pdf). https://www.ijitee.org/wp-content/uploads/papers/v9i1/A4437119119.pdf

[21] Sarah Khaled; Neamat El-Tazi; Hoda M. O. Mokhtar "Detecting Fake Accounts on Social Media" in IEEE Xplore. https://ieeexplore.ieee.org/document/8621913

[22] Patrícia Pires (2022) "Há predadores sexuais nos jogos da moda. Abusos online a crianças aumentaram 20% durante o confinamento" in CNN Portugal. https://cnnportugal.iol.pt/abusos-sexuais/pornografia-de-menores/ha-predadores-sexuais-nos-jogos-da-moda-abusos-online-a-criancas-aumentaram-20-durante-o-confinamento/20220201/61e6fbaa0cf2cc58e7ddddb8

[23] "when to use to_categorical in keras" in Stack Overflow. https://stackoverflow.com/questions/44110426/when-to-use-to-categorical-in-keras

[24] "Why normalization is necessary in ANN?" in ResearchGate. https://www.researchgate.net/post/Why_normalization_is_necessary_in_ANN

[25] Tensorflow. https://www.tensorflow.org/

[26] "What is the F1-score?" in https://www.educative.io/answers/what-is-the-f1-score

[27] "Métricas de avaliação" in https://vitorborbarodrigues.medium.com/m%C3%A9tricas-de-avalia%C3%A7%C3%A3o-acur%C3%A1cia-precis%C3%A3o-recall-quais-as-diferen%C3%A7as-c8f05e0a513c

[28] "Machine Learning -Métricas de avaliação: Acurácia, Precisão e Recall, F1-score" in https://medium.com/@mateuspdua/machine-learning-m%C3%A9tricas-de-avalia%C3%A7%C3%A3o-acur%C3%A1cia-precis%C3%A3o-e-recall-d44c72307959

[29] Brownlee, Jason (2018) "A Gentle Introduction to Dropout for Regularizing Deep Neural Networks" in Machine Learning Mastery. https://machinelearningmastery.com/dropout-for-regularizing-deep-neural-networks/