Maastricht University
School of Business and Economics & Faculty of Science and Engineering

A Research Reproducibility and Replicability Study on Detecting Sponsored Posts on Instagram

Bachelor Thesis - Darian Othman

Institute of Data Science (Maastricht, NL)
Supervisor and Advisor : Dr. Adriana Iamnitchi
do.othman@student.maastrichtuniversity.nl
I6240624

Word Count: 10239

Abstract

This thesis produces a reproducibility and replicability study on discovering undisclosed sponsored post on social media on the basis of the research article "*Discovering Undisclosed Paid Partnership on Social Media via Aspect-Attentive Sponsored Post Learning*" by Dr. Kim in 2021 for the WSDM conference [18][30]. An attentive multimodal encoding structure based of three encoders (i.e. a text,graph, and image encoder) backed by state-of-the-art models (i.e. BERT, PyTorch GCN, and Inception_V3) [26][7][22][24] which can be used to perform list-wise ranked sponsorship prediction using a ListMLE optimizing function [31] is constructed and applied over two set of data to extract insights into potential biases and disparity originating from reproduction and replication of research. The study shows very good reproducibility and replicability on standardized models but exhibits hurdles once personal interpretation of application is needed.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

In the period between 2000 and 2019, the number of publications in relation to Data science has increased by a whopping 328% [23] and its interest by 100% in the last 5 years (Figure 1.1 [25]). At first glance, this substantial growth can be seen as a purely positive news. Naturally, whenever educational research is performed over a field or subject, technologies evolve. This is then known to have a direct positive impact on economic expansion [4] Which in terms increases quality of life [8].

In parallel, the book "*The Practice of Reproducible Research*" by Kitzes, published in 2017 defines a project as reproducible "*if a second investigator (including you in the future) can re-create the final reported results of the project, including the key quantitative findings, tables, and figures, given only a set of files and written instructions*". This definition is made distinct to the concept of replicability which is defined as "*the ability of a researcher to duplicate the results of a prior study if the same procedures are followed but new data are collected*" [20].

Therefore, we ask the question: Can Data Science research results be reproduced and if so is performance impacted when applying methodology to a different set of data? In other words, we conduct a study on the differences in application of reproduction and replication of research in Data Science. The reason for such study comes from the intrinsic worry that a lot of research does not necessarily mean good research. By that, we imply that research can be qualified as useful if and only if knowledge and inspiration can be extracted from it. Thus, we perform research to study whether modern research on a regarded field can be

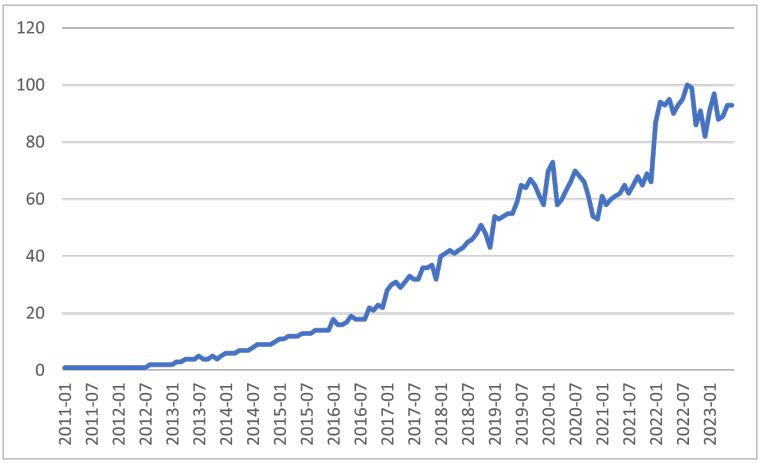

Figure 1.1: Research Frequency of Keywords "Data Science" Since 2011

reproducible and/or replicable. In other words, can knowledge be extracted from a research article.

For this, we analyze a highly developing topic: sponsorship detection on Instagram posts. Indeed, various research has been performed on post engagement for sponsored content on Instagram [6][12]. Although contradictory in the outcomes, they highlight the uncertainty surrounding the disclosure of sponsorship on social media as influencers are not properly able to predict the impact of disclosure on engagement. Consequently, internet celebrities are tempted to hide sponsorship information to their audience. Because such practice is illegal governments all around the world are increasingly finding ways to tackle this issue. As a matter of example, the Netherlands have amended an updated version of the Social Media & Influencer (RSM) Advertising Code in July 2022. The said code stipulates that information on sponsoring should be given to the public when the relationship between an influencer (here called distributor) and a brand exhibits "*a contract, sponsorship, or offering of free products to the distributors*" [1].

One research is selected for this task: the research article "*Discovering Undisclosed Paid Partnership on Social Media via Aspect-Attentive Sponsored Post Learning*" by Dr. Kim in 2021 for the WSDM conference [18][30]. This research develops a novel way of detecting undisclosed paid partnerships on social media by introducing an infrastructure composed of ranked aspect-attentive multimodal encoders using data collected from 1,601,074 posts by 38,113 influencers and

including 26,910 brands over a period of 6 years. The team outputs an average precision wandering around 95%. In Chapter 2 we explain the methodology to reproduce this research and apply it in Chapter 3. On the other hand, Chapter 4 tackles the challenge of replicating the results found in the previous chapter using a different set of data. In order to perform this, we use a dataset from an ongoing research at Maastricht University which contains over 1.2 million posts from 400 influencers of 4 different countries (i.e. Brazil, Germany, the Netherlands, and the United States).

Hence, the aim of this thesis is to analyze research reproducibility and replicability by applying a rigorous methodology for implementation and evaluation. Furthermore, we hope to create a framework for establishing and communicating research results in a scientific manner for future reproduction and replication of research.

To better understand the task of discovering undisclosed sponsored posts on social media, we now explore the work that has been already performed. Indeed, with about 500k influencers on Instagram [9] it is nearly impossible to counter non-disclosures by hand. Thus, research have been performed to develop and adapt technology for this problem with a multitude of different solutions to this task. One type of example is seen in Maastricht University which is currently conducting research on this exact topic by using artificial intelligence to assess whether or not a post is sponsored from a labelled dataset [3]. Another approach is to use learning-to-rank models with Graph Convolutional Networks (GCNs) to achieve state-of-the-art Cohen's kappa coefficient on labels of 0.784 [19] . These approaches base themselves on Natural Language Processing (NLP) techniques developed greatly, but not restricted to sentiment analysis [10] and topic modelling [34]. In this case, they are used to analyze the underlying message of captions and/or graphic aspects of the posts. The latter option was achieved using a variant of Convolutional Neural Network (CNN) which assign a node to each vector of a graph. Convolutional layers are then applied to the vectors where each layer make use of the neighbouring nodes to perform a weighted calculation in parallel of a non-linear function. This is applied to images by associating, each pixel or group of pixels with a node in a graph through a GCN and a Long Short-Term Memory (LSTM) model. Doing so will help extract the object of the said image in text form [32]. Lastly, plenty of research have been

performed in the field of image classification with usage, for example, in medical practices with novel methods to analyze pulmonary images using another CNN specialised in image classification [28]. In each of the explained research, a specific modality is used. Namely, text, metadata, and images while using specific model architecture for the desired task. In the research paper by Kim, all three modalities are used to achieve state-of-the-art results by applying attention [26] over each modality-based encoder and ranking each post's feature representation with ListMLE [31] to output the likelihood of sponsorship of every post.

In this thesis, we first develop the methodology for establishing Kim's architecture based on the research paper in details, we then go over the steps taken to reproduce to the most analogous of way the said research using Kim's dataset and we analyze the results. Penultimately we attempt to replicate the results using the second dataset from Bertaglia, and we finally compare the performances and draw conclusions. As is explained in a timely manner throughout this thesis, the entirety of the code is available on a Github repository[1].

[1]https://github.com/DarianOthman/darianothman-bach-thesis

Chapter 2

Methodology

In this section, we elaborate on the methodology applied during this research. As mentioned in Chapter 1, the methodology is separated in two parts. Namely, the research reproduction and the cross domain replication. Because the aim of this research is a study on reproducibility and replicability, it is fundamental that the elaboration of the methodology is done entirely on the basis of the research by Kim et al. [**?**]. As illustrated in Figure 2.1, four main steps can be outlined: the data pre-processing, the attentive encoding, the sponsorship prediction, and the evaluation.

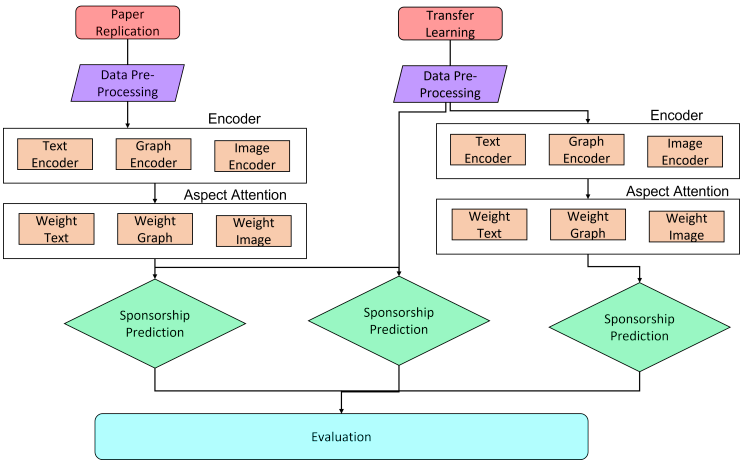

Figure 2.1: Methodology flowchart

Following this idea, the first subsection of this attempt to reproduce Kim's research focuses on the extraction and management of the data. For this subsection, the availability of data is a main focus. Data availability is defined in the book 'Principles of Information Security' by Whitman et al. as the principle under which data "enables authorized users—people or computer systems—to

access information without interference or obstruction and to receive it in the required format."[29]. This definition is the preferred and accepted one throughout this work. The reason why elaborating on this concept is paramount comes from the intrinsic difference made between reproducibility and replicability in research methodology analysis. Namely, these concepts varies from one another from the type of data and methodology adopted to achieve analog results as previously explained in Chapter 1.

The second section of the methodology pertains to the generalizability analysis. As explained previously in this section, the adopted methodology and used data make the difference between research reproduction and research replication. Following the accepted definition in this work, the second section of this thesis is defined as a cross domain research replication. That is, sponsorship predictions techniques are entirely kept from Kim's research however the data used varies. Although explained in much greater details in Section 4.1, We make use of data retrieved in a parallel research performed in Maastricht University by Thales Bertaglia [2] which possess key differences both in the data acquisition such as the definition of what constitutes an influencer and how to retrieve information from Instagram. We proceed to apply the same methodology as in Chapter 3 by encoding the data, applying attention, and predicting sponsorship while ranking these predictions.

It is important to note that the entirety of the code used in this research is available on Git Hub[1] for future usage. The data can be accessed under approval of Dr. Seungbae Kim[2]

2.1 Multimodal Encoders

To leverage the information on whether an Instagram post is sponsored or not, the data goes through three encoders on which attention is applied [27]. Each encoder focuses on one specific modality of the posts. Namely, the text present in the caption, the images, and the metadata information. First, the caption of each post goes through a text encoder which uses the bert-base-uncased model [14] and a PyTorch auto-tokenizer [21]. Then, the node features are established

[1]https://github.com/DarianOthman/darianothman-bach-thesis
[2]https://github.com/ksb2043/WSDM21˙Sponsored-Post-Detector-SPoD

and applied over the data which is inputted in a graph encoder to link each post with its influencer and potential brand or brands. A GCN is used over the newly created graph. Finally, the images of each post are analyzed by Google's Inception V3 model and a probability list is created. The information created from the three encoders are all retrieved to be inputted in the system explained in Section 2.4.

2.1.1 Text Encoder

In order to perform predictions over the contextualised caption of each post, the Bert model is used. More specifically, this research uses the bert-base-uncased transformers model to perform tokenization over the text. As per summarized in Figure 2.2, the general data discussed in Section 3.1 is divided in three random splits: a train split (50% of the original dataframe), an evaluation split (25% of the original dataframe), and a test split (25% of the original dataframe). All three splits are parsed through the auto-tokenizer using bert-base-uncased. This model can be used supposing the dataset offered by Kim should solely contain English captionned posts. If however, we were to apply this method to a dataset which contains other language, the bert-base-multilingual cased or uncased are also available. It is however important to mention that although being state-of-the-art in their domain, these two models are slightly lagging behind the general English based Bert model.

In order for data to be accepted by the BERT model, one needs to make sure the maximum length of a caption is 510+2 tokens. That is, there are 510 tokens allowed from the caption and an automatic extra tokens "[CLS]" is added at the beginning and end of each string and are used as delimiters. The caption is inputed into the auto-tokenizer which outputs three sets of information per post. Namely, the input_ids (i.e. the tokens for each word), the token_type_ids (i.e. the role played by the word in the sentence such as verb, noun, pronoun, etc.), and the attention_mask (i.e. a matrix which outputs as many 1s as there are words in the caption and fill the rest of the 510 available tokens by 0s. This is used to have square matrices of tokens). Once again, the entirety of the outputs are stored in a dataframe for future access.

Before getting into how to predict whether a post was sponsored or not, a look is given to the way in which we evaluate the results. For this, 4

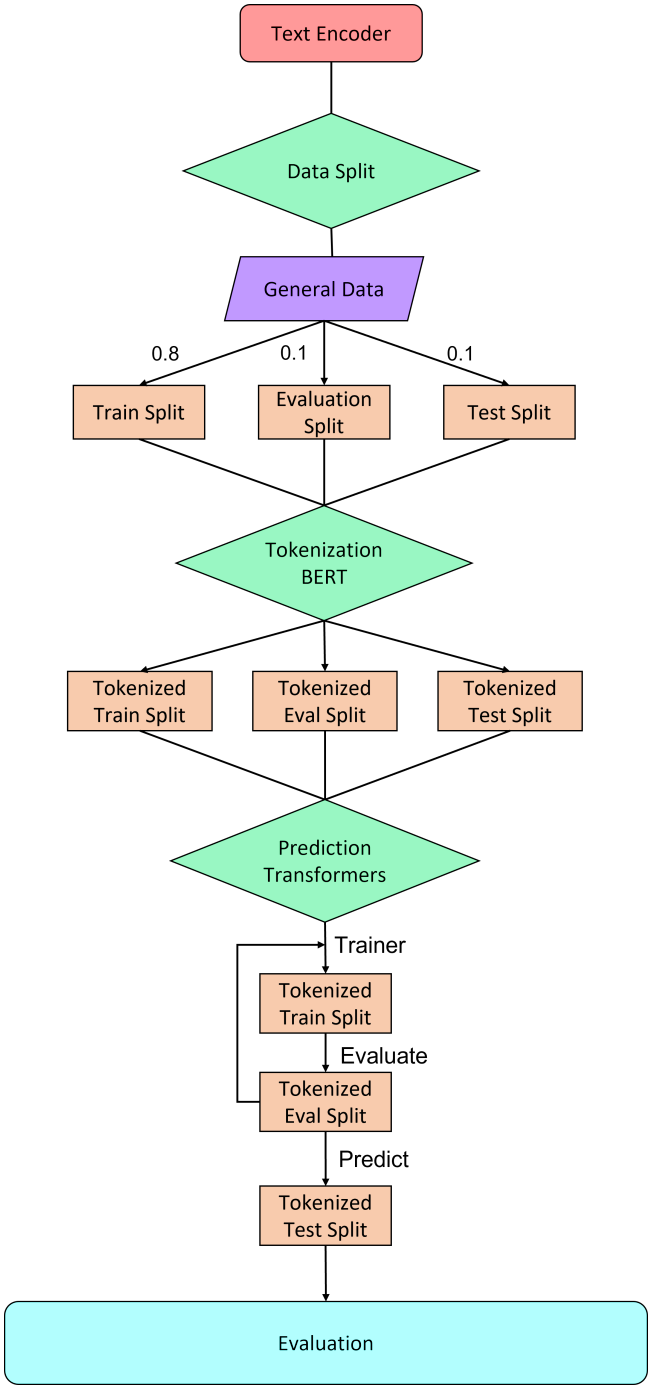

Figure 2.2: Text Encoder

statistical tools are used. These are: The accuracy, the precision, the recall, and the F1. This part of the section aims at explaining the difference between each one of these tools and why we use them. To begin, the accuracy function is most probably the most common evaluation formula used. Indeed, to calculate the accuracy of a prediction, one needs to divide the correctly predicted case over the number of total cases as seen in Equation 2.1:

$$Accuracy = \frac{True_{positives}}{Total_{cases}} \tag{2.1}$$

Following this, the precision and recall functions are linked to one another as they output two faces of the same coin. On the one side, the precision function will calculate the proportion of true positives over the entirety of positives outputted by the model. On the other side, the recall metric will show the proportion of true positives over false and true positives. In our case, the precision is particularly interesting to analyze as it gives us an idea on whether is confronted with the imbalance problem (i.e. if 90% of posts are not sponsored, the model might classify all posts as "not sponsored" to have an accuracy of 90%). The recall solves the same problem however from the other side of the imbalance (e.g. 10% of posts are not sponsored). Equations 2.2 & 2.3 summarize these concepts:

$$Precision = \frac{True_{positives}}{Total_{positives}} \tag{2.2}$$

$$Recall = \frac{True_{positives}}{True_{positives} + False_{negatives}} \tag{2.3}$$

The last metric, the F1 metric is a combination of the precision and recall metric by being their harmonic mean. In practice the F-Score will range between 0 and 1 to illustrate whether a model is able to correctly assign cases while making sure not to create false predictions. In other words, the F-score is widely used in machine learning as a more detailed accuracy score. The mathematical expression for this metric is shown in Equation 2.4:

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{2.4}$$

To predict sponsored Instagram posts, the trainer loop from the PyTorch Transformers module was used. As explained before, the model is based on the bert-

base-uncased documentation. On top of this, we specify the possible outcome by changing the ”sponsored” column to the type boolean. This means 2 labels are possible: True (i.e. sponsored) or False (i.e. not sponsored). The trainer is applied for 3 epochs on the train split and evaluated at the end of each epoch on the eval split. At the end of the 3 epochs, the newly trained model and tokenizer are saved and then applied over the data. The last hidden state of the model for each post is stored to be inputted in Section 2.2.

2.1.2 Graph Encoder

The second encoder through which the data is parsed is the graph encoder. In order to fully understand how this structure is used, the node features need to be defined. As a matter of fact, as previously mentioned, a graph encoder basing itself on an heterogeneous graph and a GCN is constructed. Moreover, in order to create the heterogeneous graph, nodes and edges are created. The earlier possesses a series of attribute and are connected to related nodes through a set of rules. These attributes and set of rules are called node features throughout this research. This section thus illustrates these node features before explaining the inner workings of the GCN for sponsoring prediction.

Three types of nodes are identified in the creation of the heterogeneous graph. The post nodes, the brand nodes, and the influencer nodes. Each type of node has its own set of features as described here: the post nodes hold the amount of likes, comments, hashtags, tags, and images in the post as well as the posting time. The brand nodes posssess the business type and the amount of followers, followees, and published posts for the corresponding brand. Lastly, the influencer nodes contain features analog to those of the brands. Indeed, they similarly hold the amount of followers, followees, and published posts alongside the interest category of the corresponding influencer as summarized in Table 2.1.

Node Type	Features
Posts	# Likes
	# Comments
	# Hashtags
	# Tags
	# Images
	Posting Time
Brands	Business Type
	# Followers
	# Followees
	# Published Posts
Influencers	Interest Category
	# Followers
	# Followees
	# Published Posts

Table 2.1: Node Features

In parallel to this, a set of rule is established regarding the construction of the heterogeneous network. First, each post node is linked to its influencer. That is, the author of the post. Second, post nodes that identifies one or multiple brands inside their caption are also linked to these said brands. Lastly, a node's name is its corresponding post shortcode (i.e. an identifier given by Instagram) to match the posts with their influencer and brand(s). This label is exchanged once the edges are created by numeric labels. Overall, these features are incorporated to create an heterogeneous graph using the dataset explained in Section 3.1. Note that the entirety of the heterogeneous network was created using the NetworkX library.

As aforementioned, a graph convolutional network is used to leverage the heterogeneous graph information. This GCN is a set of hidden layers and an activation function on which the information is propagated. For this, two objects need to be created for this task. First, an adjacency matrix is created and normalized using Equation 2.5 where A is the adjacency matrix and D is the diagonal matrix of node degrees. In a few words, the adjacency matrix holds information regarding the existing edges in a graph. This object is particularly interesting in the context of this paper as the GCN will run through each node

and its neighbors (i.e. connected nodes) to understand their similarities. This particular function is present in the different layers of the GCN meaning that the more layers we use, the further away from a node will be analyzed by the model (i.e. the neighbor of a node, then the neighbor of the neighbor of a node, and so on).

$$\hat{A} = D^{-1/2} A D^{-1/2} \tag{2.5}$$

The second object created for this task is the feature matrix Z which stores the attribute of each post (P), influencer (I), and brand (B) nodes:

$$Z = [Z^P; Z^I; Z^B] \tag{2.6}$$

As explained, these are the information which the GCN uses when running through the normalized adjacency matrix. Once these objects are fed to the model, a set of information is outputted. From these, we retrieve once again the last hidden layer defined by Equation 2.7 for future usage where σ is a nonlinear activation function, $H^{(i-1)}$ is the output of the previous hidden layer, $H^{(0)}$ is the feature matrix, and $W^{(i-1)}$ is a set of trainable weights.

$$H^{(i)} = \sigma(\hat{A} H^{(i-1)} W^{(i-1)}) \tag{2.7}$$

2.1.3 Image Encoder

Finally, the third and final encoder is described. To leverage the information present in the posts' images, the Inception_V3 is used. This Convolutional Neural Network (CNN) is the third itteration of Google's Inception model. The model is trained on 1 million images with 1000 object categories and under 25 million parameters. This state-of-the-art infrastructure exhibits a 3.5% top-5 error on the validation set [24].

In order to make the model work, each image is downloaded and re-sized to conform with the model's preferences. That is, a 299/299 size for all images. Once this step is done, the model can be applied to each image to output the probability score for each object from the imagenet class on which the model

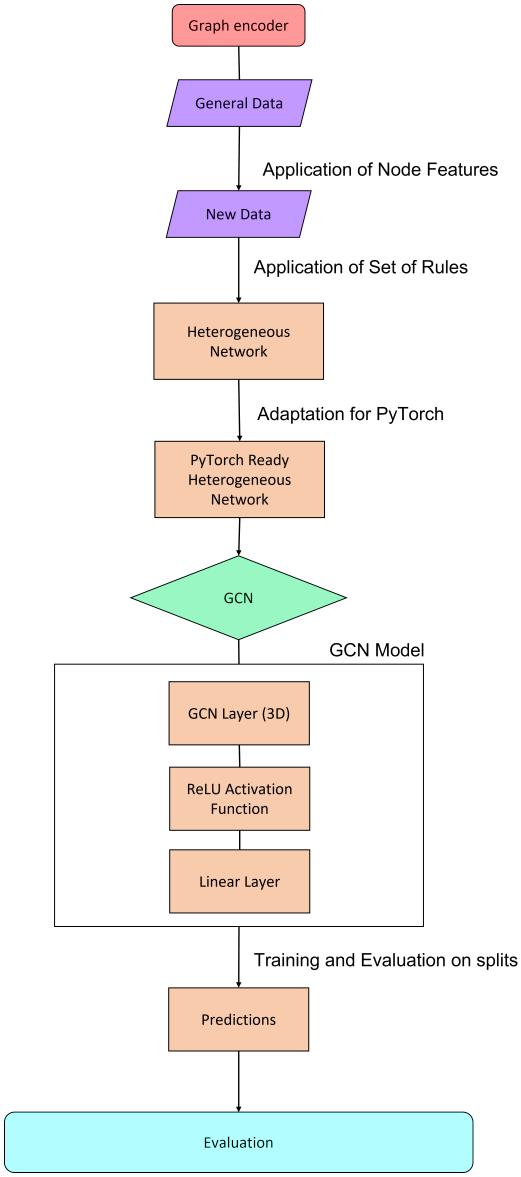

Figure 2.3: Graph Encoder

makes his predictions. This therefore creates a tensor of shape [1000,1] for each post. Just like the other two encoders, the representations of all the posts are stored together.

This being said, the usage of this last encoders is not covered in this work first due to a lack of ressource, and second due to the unavailability of data. Indeed, as is explained in a later section, the second dataset used in this work, Bertaglia's dataset, does not provide images for each posts. Moreover, even though Kim's dataset does feed images, the file size of 45GiB is not accepted by the system. Indeed, in order to maximise computing power, a virtual JupyterLab backed by three Maastricht University GPUs is used. However, this virtual access has its constraints. Namely, a user can only upload a fixed amount of data to their folder. Therefore, these two hurdles renders the usage of a third encoder not feasible. For the sake of future usage and to help undestand the reproducibility and replicability, this chapter will continue to describe the remainder of the processes with the usage of the image encoder.

2.2 Aspect Attention

As mentioned in Chapter 2, a layer of attention is applied to each encoder. Whether attention is applied on the BERT, Inception V3, or the GCN model, the last hidden layer of embeddings is retrieved and parsed using the multihead attention layer from PyTorch [27]. Once retrieved, the importance of each feature (i.e. the encoders) is measured using equation 2.8

$$\alpha_i = \frac{exp(r_i \cdot r^c)}{\Sigma_j \, exp(r_j \cdot r^c)} \tag{2.8}$$

Where $r_i = tanh(Feature_i)$ is the hidden representation of an encoder and r_c is the context vector. This last variable is highly important for importance estimation. Once the weighted scores are establish by multiplying α_i to its feature, the sum of weighted features corresponds to the representation of a selected post following:

$$X = \Sigma_i \alpha_i \cdot V_i \tag{2.9}$$

With the set of features from a particular encoding being:

$$V = [V^G, V^T, V^I] \tag{2.10}$$

By applying self attention to the encoders, the aim is to increase not only the accuracy but also the better depiction of results. To perform this, we feed to the multi-head attention layer the same data for each of the key, query, and value arguments. Doing so will enable the integrated layer to find which part of the model has the biggest impact over the result and adjust the final embedding weights according to this.

2.3 Sponsorship Prediction

With all the aspect-attentive feature representations retrieve, the sponsorship predictions can be done. For this, we concatenate the image, text, and graph representations together in a single tensor per post. The length of the total representation is therefore dictated by Equation 2.11 with L_i^G, L_i^T, and L_i^I being respectively the length L of the feature graph, text, and image.

$$L_i = L_i^G + L_i^T + L_i^I \tag{2.11}$$

The data is passed to a series of fully connected hidden layers β_p and β_h and a nonlinear activation function σ to find the predicted sponsorship score $\hat{y}_i$, as seen in Equation 2.12. For this implementation, the PyTorch library is used as it offers a wide range of application and easy creation of user defined functions [22].

$$\hat{y}_i = \beta_p(\sigma(\beta_h(X_i))) \tag{2.12}$$

2.4 List-Wise Ranking

Finally, the methodology proposed by Kim and their team includes a machine-learned ranking approach using a fined-tuned ListMLE [31]. This ML application

creates an optimization function which receives, as an input, the X set of features and the Y set of object permutations. To better understand the concept of set of object permutation, we present an example for a dataset of 4 posts where each cell can receive a positive sponsorship prediction, 1, or a negative sponsorship prediction, 0. Two possible object permutations are permutations A and B in Equation 2.13. Therefore the set of object permutations can be calculated using Formula 2.14 with n! being the factorial of the number n of posts in our dataset and $n_r!$ the factorial of r possible results. In our case, each two possibility 0 and 1 can appear as much as n times in the dataset.

$$A = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} \qquad B = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \end{pmatrix} \tag{2.13}$$

$$P(n, r) = \frac{n!}{n_1!\,, n_2!\,, ..., n_r!} \tag{2.14}$$

A ranking formula is thus represented $\hat{y} : X \rightarrow Y$ and the optimization function for the expected loss $R(\hat{y})$ can be established and derived by using a random but fixed joint probability distribution P_{XY} in the following Equation 2.16:

$$R(\hat{y}) = \int_{XxY} L(\hat{y}(X_i), y) dP(X_i, y) \tag{2.15}$$

With $L(\hat{y}(X_i), y)$ being the 0-1 loss between the ranked result $\hat{y}(X_i)$ and the position in the permutation y such that:

$$L(\hat{y}(X_i), y) = \begin{cases} 1 \Leftrightarrow \hat{y}(X_i)\, y) = y \\ 0 \Leftrightarrow \hat{y}(X_i)\, y) \neq y \end{cases} \tag{2.16}$$

In a similar but slightly different manner than what Kim's research did, we batch the n posts present in our dataset to make the process more efficient and minimize the empirical loss R_S using Equation 2.17 with the ground truth for sponsorship illustrated as y_i.

$$R_S(\hat{y}) = \frac{1}{n} \sum_{i=1}^{n} L(\hat{y}(X_i), y_i) \tag{2.17}$$

As a conclusion to this chapter, we point out that one section in Kim's research is left out of this thesis methodology: the temporal regularization. Indeed, due to resource constraints, performing this last section was unfortunately not feasible. Consequently, we emphasize that each comparison of performance between this thesis and Kim's research is done on scores without temporal regularization. We thus hope to reproduce this step in the future with more resource at hand.

Overall, this section went over Kim's ranked aspect attentive multi-modal encoding architecture for discovering undisclosed sponsored posts on social media. Namely, we elaborated on text, graph, and image encoders backed respectively by Bert, PyTorch GCN, and Inception_V3 models. The concept of attention was then explained to grasp its importance and usage, and finally ranked sponsorship prediction process was detailed using a more mathematical approach for a better understanding of the matter at hand. The aim of this chapter was to develop in a precise manner how to reproduce or replicate the said research using the published paper as the sole source of guidance. The application of this architecture respectively for reproduction and replication purposes is discussed in Chapters 3 and 4 and conclusions on reproducibility and replicability of research in data science using the knowledge gained from this methodology chapter and the two application chapters are discussed in Chapter 5.

Chapter 3

Research Reproduction

To deeply understand the architecture proposed by Kim et al. [18], we reproduce their work using solely the information and data given in their research paper. We start by extracting the dataset from the open source link provided in the paper[1]. The dataset is then modulated in analogy to the descriptions given. Doing this ensures that no external biases are introduced from this step. Two encoders are then applied to data. Firstly a contextualised text encoder using the state-of-the-art BERT model [**?**] is applied to the data. A Graph Convolutional Network (GCN) [22] is then applied on an heterogeneous graph which uses instagram posts, influencers, and brands as nodes and the relation between these as its edges. The node features and setting of the GCN are explained in further details in Section 3.2. In order for each encoder to have the correct impact on the final result, we apply attention over them which outputs different weights of importance for each of the encoder's features. Following this step, we make use of these weights to make sponsorship predictions for each post. The final steps introduced in this paper regards the list-wise ranking of the results. We apply a modified version of ListMLE to optimize the loss function in our prediction. The inner workings of each model and processes are explained in the following subsections.

3.1 Dataset

In this chapter, the dataset used is one originating directly from the aforementioned paper by Kim et al. This subsection aims at understanding the data which constitutes the dataset as well as the steps taken to process

[1] https://github.com/ksb2043/WSDM21_Sponsored-Post-Detector-SPoD

the said data solely for readability purposes but not for usage purposes, this is discussed in the next section of this chapter. It is important to mention once again that all the steps taken are directly extracted from the paper and are performed to reproduce the work in the most analogous manner.

The data used in this paper can be classified in four distinct categories. Namely, the posts, the influencers, the brands, and the posts' pictures. Each category is imported in a different way due to the format in which the data is offered by Kim. To be more precise, post data are stored in one json file per post, pictures are stored as jpg in different zip files, the brands and influencers data are stored in a text file for each brand, and influencers. However, the influencer files are renamed prior to opening as the file naming was done incorrectly. Namely, the files were given the Instagram username of each influencer. Unfortunately, this causes serious problems as it creates unreadable files (i.e. if someone is called 'name.surname' on instagram, the file would be considered as a '.surname file'). The issue is resolved by iteratively renaming all files using an external application. We extract the information from each json and text file using an os module and store each post's, influencer's, and brand's data as a row in three different pandas dataframe.

Once this step is performed, a series of data pre-processing steps are applied on the newly created dataframe for the posts data. Mainly, the steps consist of two phases: One is the isolation of the principal columns mentioned and required in the research and second is a a swift analysis of the data available for either merging purposes or any other work simplification. All-in-all, 10 changes are made to extract all the information needed from the data.

Next to this, two other dataframes are created. Namely a dataframe to store the influencers' data and one for the brands' data. Each of these are also revised and pre-processed. These said djustments are illustrated in Tables 3.1 & 3.2. The final step in the preparation of the data pertains to the merging of dataframes which is done through column mapping using the aforementioned changes on the dataframes (e.g. the 'datax' dataframe is used for merging using the brands name in both this dataframe, and the posts' captions).

#	Column Name	Change
1	Unnamed: 0	Only keeping the rows where the column is equal to 0 (properly imported rows)
2	datax	Creating a DataFrame which holds the file name of each brand (equal to their Instagram tags)
3	ind	Recreating an index
4	tag	Creating a new column in the main DataFrame with the tags from the datax DataFrame
5	tag	Adding a "" in front of each row for easy merging with the posts' DataFrame

Table 3.1: Changes Made for the Brands

#	Column Name	Change
1	Name_lower	Creating a new column which holds the names of each influencer but uncased
2	Name_lower	Transforming the column to type str in order to keep the first word only

Continued on next page

Table 3.2 – continued from previous page		
#	**Column Name**	**Change**
3	Name_lower	Splitting every word and selecting the first one for easy merging
4	Category	Dropping NA as this column is a node feature
5	tag	Creating a column which extracts the name of the file as the influencer name for merging

Table 3.2: Changes Made for the Influencers

3.2 Experimental Setup

To continue, we define the experimental setup used to reproduce Kim's research. In more details, the aim of this section is to elaborate in the clearest manner on the work which was done to apply the structure developed in Chapter 2 over this chapter's dataset. The section is divided in two parts: the attentive encoding and the ranked sponsorship predictions. In the first part, we mainly explore the changes made in columns to smoothly run the desired tasks as well as how each information is stored for future usage or evaluation. The second part of this section uses the information generated to perform sponsorship predictions for which the goal is, once again, to rigorously explain the steps taken to reproduce in the most analogous way the research made by Kim's team and how the final predictions are stored. This latter part is crucial both for the attentive encoding and the ranked sponsorship prediction subsections as Chapter 5 builds a global analysis between the work performed when reproducing and replicating research. Therefore, all the weights created when constructing models, the outputs extracted when performing our work, and the various mid-process evaluations need to be properly stored for subsequent usage.

3.2.1 Attentive Encoding

Consistent with the process exposed in Figure 2.1, the pre-processed data constructed in Section 3.1 is parsed through two encoders. First, we discuss how our data is parsed through a contextualised text encoder.

Because each set of data is constructed differently, the dataset needs to be slightly adapted in 3 ways: First the column "edge_media_to_caption" is changed to "text" and all its values are changed to strings, second the "sponsored" column becomes "label", and finally the captions in the newly created "text" column is limited to a maximum of 510 tokens. To better understand the last change, refer to Section 2.1.1. Once these changes are made, a training split is parsed through the encoder alongside an evaluation split to train the model. At the end of 3 epochs, six files are retrieved: the model's configuration and weights, and the tokenizer's configuration, mapping, weights, and vocab. These can easily be retrieved using a simple command on python [15]. The model and tokenizer are then used on the entire dataset while retrieving the last hidden state for each prediction. It is important to note that we store the last hidden state of the model in our dataframe for each post following Equation 3.1 where X_n is the representation of the n post. The results and accuracies of each encoder are further discussed in Section 3.3.

$$X = [X_1, X_2, X_3, ..., X_n] \tag{3.1}$$

To continue, we discuss the work performed to run the second encoder: the graph encoder. For this process, we start by numbering the desired columns as seen in Table 3.3 where the interest category column is modified as a matter of example. This process is very important as only numerical values are accepted as inputs. The second and maybe most important step taken in the process is related to the set of rules which governs the creation of our heterogeneous graph. As a matter of fact, each caption's tags are kept together as a string and then separated and stored in a different dataframe. By doing so, it is very easy to create edges between nodes by running a loop which maps cell values in our dataframe to node names in the NetworkX Graph. Finally, we take out the NA from each column present in the node features (see Table 2.1) which removes, after all the changes are applied, between 25 and 30 % of the dataset. Although this might sound like a big number, thanks to the original size of the dataset, the new dataset

is 1.1 million rows long. The processing being done, the data is fed to the model to create the adjacency and feature matrices and the GCN is applied. The output of the GCN is then stored in accordance with Equation 3.1

#	Old	New
1	Category Creators & Celebrities	1
2	Publishers	2
3	Personal Goods & General Merchandise Stores	3
4	General Interest	4
5	Non-Profits & Religious Organizations	5
6	Transportation & Accomodation Services	6
7	Home Services	7
8	Business & Utility Services	8
9	Home Goods Stores	9
10	Lifestyle Services	10
11	Local Events	11
12	Food & Personal Goods	12
13	Professional Services	13
14	Content & Apps	14
15	Grocery & Convenience Stores	15
16	Restaurants	16
17	Auto Dealers	17
18	Government Agencies	18
19	Entities	19
20	Geography	20
21	Home & Auto	21

Table 3.3: Changes to the Category column

With all the encoders prepared and their output properly stored, we apply self attention over each output by following the methodology from Section 2.2. Because this procedure relies on standardized data outputted by similar models, no data-specific adjustments need to be made. That being said, the output from each encoder needs to match the correct post which is why we insist on the proper and rigorous storage of data. For this task, we not only store the tensors in separate

files but also alongside each post's identification number in a dataframe.

3.2.2 Ranked Sponsorship Predictions

The methodology of Section 2.3 is then applied over each post's representation to extract a single sponsorship prediction which is then ranked using ListMLE [31]. This process uses both the standardized data from the post representation's tensors and the labelled 'sponsored' column from the dataset. This said column was given by Kim and outputs wether or not a post is sponsored following both the criterias from their papers "*Discovering Undisclosed Paid Partnership on Social Media via Aspect-Attentive Sponsored Post Learning*" and "*Multimodal Post Attentive Profiling for Influencer Marketing*" [18][17]. Once the predictions are made and ranked, we store each post's final result in the previously used dataframe.

3.3 Results

As the final section of this chapter on research reproducibility, each parts of the aforementioned process are ran and the results are discussed. In order to apply this, we use the explained evaluation criterias on the relevant portions of our work, as per Chapter 2. First, we train the text and graph encoders using train, test, and eval splits of adjustable proportions although it is important to mention that these splits are kept for the entirety of the process in order to perform an evaluation over the entire architecture and not only the specific encoders or prediction models.

Because each model used are widely known, they possess baseline standards to which we can compare our results. Moreover, The research paper by Kim further provides insights into the performance of the achitecture. Namely, the BERT documentation issues an accuracy of the different models hovering around 86% [14] with Kim showing a precision at k=150 of 67.3% [18]. On the other hand, the PyTorch GCN model outputs an average accuracy for a classification task over a set of four different datasets of 71% [19]. These results are particularly good compared to the precision scores observed by kim which go from 60% at k=10 to 38% at k=150. In order to be homogeneous in comparisons, our outputted values are all at k=150.

To go deeper in our data, we obeserve similar accuracies in the text and graph encoders. As a matter of fact, BERT model is performing at a top-tier level on a reduced dataset outputting an accuracy of almost 99% and an F1 score of 95%. These scores are then closer to the baseline when applying the model to the full dataset with an accuracy of 87% and an F1 score closing on 85%. As for the GCN classification model, the accuracy stands at 91.2% which, as we mentioned, is very comparable with the BERT model however the F1 scores lives much lower with a result of almost 25%. The reasoning behind this is that the model learns the data rather than the patterns inside it. Thus, it finds the majority class and make predictions that are severely oriented towards it to increase its accuracy. In order to compensate for this phenomenon, we can adjust the dataset to account for any class imbalance. By doing so, the accuracy decreases by almost 10 points but the other metrics (i.e. the precision, F1, and recall) all increase by 6 percent points. These results consequently show consistency with the previously adopted architectures and models.

To continue, attention is applied to the data and predictions are made. During this process, evaluation metrics are generated at two distinct moments: after making the initial predictions and after the optimised list-wise ranking. For both of these steps, baselines from the reproduced research paper are available. Namely, Kim's team observe a precision of almost 87% during the first prediction round and 96.7% when list-wise ranking is applied. These very high scores are state-of-the-art and serve as a very high baseline for discovering undisclosed sponsorship on social media. With this, we compare the results of our initial predictions which are much lower with a precision score around 60% and an F1 score of 20%. During the second round of predictions, once the ListMLE loss is optimized, the architecture exhibits a surprising results of 16% accuracy. This being said, the model is doing much better in the F1 score with 30%, a 50% improvement.

These results end this chapter on the reproduction of research aiming at discovering undisclosed sponsorship on social media by Kim [18]. While the interpretation and reflection on the work is done in Chapter 5, the aim of this chapter was to give a deeper insight into the reproduction of a research paper by describing, in a clear scientific way each steps taken to either adapt the data at

hand or construct the required architecture. In this effort, the dataset extraction and pre-processing was first explained and followed by the experimental setup separated into two distinct sections to study the construction of a multimodal attentive encoding structure using one text and one graph encoder over which attention was applied. The outputs of this structure were then used in the second section to perform ranked sponsorship predictions using fully connected hidden layers to our post representations and a ListMLE loss optimization function to leverage as much knowledge as possible from the information at hand. Finally, the results and baselines for these models were given using the metrics elaborated on in Chapter 2. The final section outputted evaluations much lower than those of Kim's which calls for many interpretations. As aforementioned, these reflections will be performed in Chapter 5.

Chapter 4

Cross Domain Replication

In this chapter, the second part of Chapter 2 is applied. That is, we perform a generalisability analysis by replicating Kim's research following the outline in Figure 2.1. We first extract the trained model from Chapter 3 and make predictions on a second dataset from a research made at Maastricht University by Bertagla et al. [2]. Second, we both train and predict using the second dataset in order to evaluate the performance of Kim's research on another set of data. The three rounds of running (i.e. Trained and predicted on Kim's dataset, trained on Kim's dataset and predicted on Bertaglia's, and trained and predicted on Bertaglia's) are then discussed and compared to one another in Chapter 5. This chapter is therefore heavily oriented towards the application of our methodology in the context of research replicability rather than the discussion of results and their interpretation.

In medical sciences, "the results of a study are generalizable when they can be applied (are useful for informing a clinical decision) to patients who present for care [...] This requires nuanced understanding of the condition that defines the population, the study intervention, and the patient" [16]. We decide to expand this notion to data science by experimenting over a different approach to what an influencer is defined as and how data is collected. The aim of this chapter is to analyze whether the definition of the sampled population has a true impact on data science architecture application.

4.1 Dataset

In the aforementioned research made by Bertaglia et al [2], the data is collected using Meta's CrowdTangle [5]. 50 micro and 50 mega influencers were selected from each of 4 designated countries (i.e. Brazil, Germany, the Netherlands, and the United States of America). Micro influencers are defined as having between 100k and 600k followers and mega as having over 600k followers. In order to make sure that each and every influencer's location did match with the 4 selected countries, a manual iterative process took place to only select influencers with a well disclosed corresponding location.

Although the entire list of information CrowdTangle is able to retrieve is made public[1], Bertaglia expands deeper on what their collection technique was. Namely, they extract every possible data and metadata from the posts and accounts but as explained in their paper, the *likes* and *comments count*, *captions*, *timestamp*, *disclosed sponsorship* information, and the entirety of the accounts data such as *amount of posts*, *followers*, *followees*, or whether or not the account is verified are the main points of emphasis. Moreover, the team processed each post based on a list of hashtags and keywords with high frequency in sponsored post (e.g. *#ad*, *#publipost*, *#sponsored*) [18][33] and classified each post that contained elements of this list as sponsored. More detailed statistics regarding the distribution of *sponsored posts*, *verified accounts*, *likes*, *comments*, and *posts* per country and type of influencers are available in Bertaglia's paper.

As can be seen from the dataset exploration, the key differences between Kim and Bertaglia's datasets reside both in the way in which the data is collected and the definition of who classify as an influencer. Indeed, following the research made by Kim, an influencer is considered an Instagram account which possesses at least one post containing the hashtag '*#ad*' in its caption and whose owner has at least 1k followers with 300 posts. Therefore, Kim's dataset is sensibly larger than Bertaglia's as it is considerably more lenient in the influencer acceptance process. In addition, we specify that no further analysis is made regarding the data collection technique of Kim because although the research makes mention of a crawler used to retrieve the data, no particular technique is explained.

[1] https://help.crowdtangle.com/en/articles/4201940-about-us

Moreover, Bertaglia's dataset differenciates itself by having accounts from 4 countries with 4 different spoken languages (i.e. English, German, Portuguese, and Dutch). This is particularly important in Section 4.2.1 as it affects the type of BERT model used (i.e. the bert-base-uncased or the bert-base-multilingual-uncased) and consequently affects the overall performance of the sponsorship prediction.

All in all, the dataset used for this chapter contains 1.2 million rows. The data is only cleaned and pre-processed in order to comply with the aforementioned models and methodology. Other than that, no fundamental changes to the data are made. The following sections runs through each part of the methodology with a particular focus on both the similiarities and differences between using this chapter's data and the one from Chapter 3 for a similar task.

Finally, the data used is comprised of text, metadata, and no images as the aim of the Bertaglia's paper was not directed towards the usage of images. Therefore, the methodology proposed in Chapter 2 needs to be slightly revised as well as the final evaluation of the models. Regarding the methodology, a simple bypass can be done by excluding the image encoder but keeping all else constant. We believe this change is, compared to bypassing any other encoder, the least impactful. As a matter of fact, in Figure 6(b) of Kim's research, precision scores are exposed. In this figure, one can see the relative importance of each encoder and see that the precision using solely an image encoder is almost half of the precision when using solely a graph or text encoder. In regards to the evaluation, as it is only objective to compare similar structures, a two encoder structure will be used for evaluation purposes between reproduction and replication of research.

4.2 Experimental Setup

4.2.1 Attentive Encoding

This section's emphasis lies in the replication of the processes defined in Chapter 2 to perform a generalizability analysis over a data science research. As explained in previous sections, an attentive multimodal encoder structure is constructed using a text and graph encoders on available data. Although the sheer creation and implementation of the structure is explained in this section, the actual usage of these output is made in subsequent sections and particularly in Chapter 5

where the structure trained on Bertaglia's data serves as a baseline for an objective comparison. In a couple of words, the aim of this section is to clearly outline the steps taken in order to replicate and apply an attentive multimodal encoder structure over a different set of data.

The first encoder to be applied on Bertaglia's data is, analog to the aforedeveloped chapters, the text encoder. First, the trained model and tokenizer exported in Chapter 3 which are backed by bert-base-uncased and the PyTorch libraries are taken into usage. The weights of the last hidden state are retrieved and stored in a pickle file for subsequent usage. Second, an experiment is performed using the best-base-multilingual-uncased [14][13]. This is done because the posts in the new dataset are written in different languages. Thus, using a model which takes into account this factor is important. For this task, the columns used are the 'label' and 'text' columns. These hold the same information as the 'sponsored' and 'edge_media_to_caption' columns from Kim's dataset.

Moving onwards, the second and last encoder to be used in this process is the graph encoder. Indeed, as already explained, the encoder relies on a GCN which takes the feature and adjacency matrices as inputs. Because Bertaglia's dataset is slightly different to Kim's we apply the code previously explained in Chapter 3 to match influencers with brands. The node features are then adapted to fit the dataset as can be seen in Table 4.1. These are inputted to keep the spirit of the replicated research paper while adapting the concept to the available data. The models are then ran using the same infrastructure as previously explained. Each tensor related to the last hidden state, the attention, and final representation are kept and stored in the dataframe.

#	**Feature**
1	Likes
2	Followers
3	Comments
4	Type (photo/video/album)
5	Country
6	Size (micro,mega)
7	Language
Continued on next page	

Table 4.1 – continued from previous page

#	Feature
8	Date

Table 4.1: Node Features for Research Replication

With all the encoders ran and attention applied, the attentive multimodal representations are concatenated in accordance to Equation 4.1, an adapted version of Equation 2.10 which takes into account the absence of images in our dataset. The focus is now shifted toward the ranked sponsorship predictions.

$$V = [V^G, V^T] \tag{4.1}$$

4.2.2 Ranked Sponsorship Predictions

With the final application section of this thesis, we discuss how the attentive multimodal representations are used to construct the ranked sponsorship predictions. As discussed in Section 3.2.1, the process used to apply attention and predict sponsorship are two which use standardized data as inputs. Therefore, we follow the methodology from Section 2.3. The final results are once again stored to be analyzed and evaluated in the following section.

4.3 Results

Analogous to what was done in Chapter 3, this section focuses on exposing the results of this chapter's models. To accomplish that, we start by comparing the found results to those of the baselines and proceed to explore whether similar results to the ones from Chapter 3 are observed. This section will focus on outputting each results by using the metrics from Chapter 2 but does not aim at interpreting these said results. The entirety of interpretation and reflection will be made in the subsequent Chapter 5.

To begin, we look at the baselines of the BERT and GCN models discussed in Chapter 3. When running the earlier on our new dataset, evaluations higher than those outputted when using Kim's dataset were found. Indeed, after running the model for 3 epochs in accordance to the methodology of this chapter, we found a state-of-the-art accuracy of 99% accompanied with an F1 score of 95%

over the entirety of the dataset. On the other hand, the GCN did not performed as well with an accuracy slightly above 90% but with a precision of merely 18%. When comparing these results to the overall baselines of the two models, one can see a clear improvement compared to the BERT baseline but a mitigated outcome in comparison to the GCN model as the precision is half that of the baseline. Additionally, a comparison can be made with the performances of our previous chapter. In that regard, this second dataset is out-performing slightly the initial model by a few percentage points when looking at the BERT model and is relatively conistent to the GCN outputs by exhibiting a lower performance while however staying in the ballpark of the work performed in Chapter 3.

To continue, we perform the last part of this chapter's methodology by performing sponsorship predictions and ranking them using a list-wise ranking approach. As a remainder, the accuracy for the initial predictions in the previous chapters hang around 60% with a result of 20% after list-wise ranking. When looking at the performances of the predictions on this dataset, one can see decline in the results with an accuracy only averaging a mere 8% in F1 score which, although against a 96% in accuracy paints a much more realistic picture of the performances. Finally, the F1 score improves when optimizing using ListMLE to 13% while the accuracy comes down to 16%. These numbers are once again in direct proximity with those of the previous chapters.

This section concludes the final part of the practical oriented approach of this work, the discussions of results stands as the final chapter to reflect over the entirety of the work done as well as the interpretation for the results outputted in Sections 3.3 & 4.3. The aim of this chapter was to use all the information gained in the chapter on reproduction of research in the field of sponsorship detection on social media and apply this knowledge on another set of data. Because the methodology had been widely explored in previous chapters, the main focus resided in showing in a unbiased and clear manner the differences in approach, application, and results when applying a methodology to another set of data. We found that each part of the process was able to be replicated if data was available. Unfortunately, we also found results much lower in performance in comparisons to the baseline. However, these results were, even if lower, comparable to those of Chapter 3. These are now discussed in the following chapter.

Chapter 5

Discussion

As part of the last chapter of this work, we reflect on the work performed and its results. We use both the data outputted throughout our work and the knowledge and observations gained when performing each processes. Although an evaluation of reproducability and replicability in data science on the topic of discovering sponsored post on social media is done, we done not aim at pointing fingers or discrediting the work performed by any other specialist. The aim of this chapter, as well as this thesis, is to dig deeper into how we can extract knowledge out of academic research. For this, we review each chapter and their processes with a special attention given to interpretability. In the field of machine learning, this notion is often defined as an explanation which can "*provide qualitative understanding between the input variables and the response*" [11]. In other words, a process explanation needs to give both great insights into the fundamental inner workings of a process while being clear and understandable to the majority.

To begin, various data was used during this work. For each dataset, a specific approach needed to be taken in order to first extract the data and then pre-process it. In the research paper by Kim [18], a direct link to a Google Survey was provided. Once authorisation from the team was given, a OneDrive could be opened with the all the raw files present. As aforementioned in Chapter 3, this process was tedious as it demanded a series of independent steps to be taken. Indeed, we found that little to no information was given on either the extraction of the files or the content of these. As a matter of example, the repository contained a README file which possessed information on the data. However, this files only gave a light insight into the data at hand (e.g. the file contained a section on the available columns which contained 6 columns out of the 52 total present in the

dataset). Consequently, this process can open the door to potential biases as the data is manipulated in a way that suits the handler.

Moreover, a second set of data was used in this work. As this dataset came from a different research [2] with different objectives, the data available was highly distinct in its form and content as explained in Chapter 4. Therefore, one focus when performing a replicability study is to find whether potential biases could be seen in either the data or the models. In fact, as was discussed when touching upon the concept of over-fitting, a well performing model could originate from data specifically adapted for this architecture. Furthermore, as the data is generated in different ways, we have seen that the definition of what makes the data highly varies. The set of all these potential variable changes or interpretations can lead to substantial disparities.

To continue, disparities can be found, not only in the data but in the interpretation of the methodology. As was mentioned in each chapter, the aim of this work is to reproduce and replicate the work by Kim on discovering undisclosed sponsored post on social media in the most analogous of ways. However, due to different interpretations of their written work and different levels of expertise in this specific field, major differences can appear in the way the proposed architecture is built. Forthwith, we believe the differences in performance between the research paper and our work originate in major parts from interpretation of model architecture. As a matter of proof, the models which exhibited high levels of comparative performances with the baselines were the Bert and PyTorch GCN models. These models are both pretrained and constructed in a standardised way. Moreover, no significant contrast in results could be observed when using Kim or Bertaglia's dataset.

Overall, we conclude that a plethora of origins can be given for biases to be inputted in a research. However, we have seen that disparity in data definition and retrieval acts as a hurdle for easy reproduction and replication of work. Nonetheless, disparity in interpretation of model construction and usage still remains, in a performance oriented analysis, the most significant source of discrepancy. Therefore, this thesis is the base of a future research at Maastricht University which will aim at constructing a clearer definition for what a sponsored content is in the context of social media as well as establishing

a baseline for discovering undisclosed sponsored post on social media using contextualised prediction methods. Once again, the aim of this chapter is to communicate on the possible limitations of the work done but not to go against any research previously made.

Chapter 6

Conclusion

The book "*The Practice of Reproducible Research*" by Kitzes, published in 2017 defines a project as reproducible "*if a second investigator (including you in the future) can re-create the final reported results of the project, including the key quantitative findings, tables, and figures, given only a set of files and written instructions*". This definition is made distinct to the concept of replicability which is defined as "*the ability of a researcher to duplicate the results of a prior study if the same procedures are followed but new data are collected*" [20].

This definition served as the base for the work done in this thesis. As a matter of fact, how often can one read an academic paper from start to finish and say "Given the data, I can redo this myself"? To study this, a reproducibility and replicability study needed to be performed. That is, recreating, out of a defined set of files, an entire research architecture for a specific task and analyze the performance when the said architecture is used on a different set of data.

To perform this, the research article "*Discovering Undisclosed Paid Partnership on Social Media via Aspect-Attentive Sponsored Post Learning*" by Dr. Kim in 2021 for the WSDM conference [18][30] is selected as it explores a highly developing field of data science while offering both a framework to construct the model architecture and the data used for training and testing.

First, the definition of the methodology was developed to construct an attentive multimodal encoding structure based of three encoders (i.e. a text,graph, and image encoder) backed by state-of-the-art models (i.e. BERT, PyTorch GCN, and Inception_V3) [26][7][22][24] which can be used to perform list-wise ranked

sponsorship prediction using a ListMLE optimizing function [31]. The steps taken to retrieve data and adjust the models to reproduce the work were then developed and followed by a similar approach to replicate the research using a different set of data. In both these chapters, model results were outputted and compared to baselines. Finally, a discussion was had over the potential limitations of the work performed.

Overall, this work can be seen as a highly positive result as it successfully reproduces and replicates a research out of a given set of files. Moreover, the aim of this study was to analyze reproducibility and replicability which gave insight into the potential origins of biases and disparity in interpretations, leading to low architecture performances. These said insights will serve as a building block for a future research paper to expand on this thesis at Maastricht University Institute of Data Science.

Chapter 7

Personal Reflections

For the final chapter of this thesis, we look back on the work in a more personal way. Indeed, starting on February 6th 2023, there was little to no way to imagine the task awaiting. Thankfully, each individual colleague with whom I have had the honour and pleasure to work with for the last 5 months have given me all the support and help needed to successfully write this thesis.

Indeed, my background as a business engineer armed me with skills and knowledge on the basics of data science. However, there is an obvious gap between completing weekly assignments on a well defined set of conditions to fulfill and producing an entire work from scratch. I have quickly found that conversing over a coffee was sometimes of better help than reading an entire documentation on a specific function as each person I have met are not only eager to perpetually increase their knowledge in the field, but most importantly are eager to share it. A concrete example happened at IDS' Women in Data Science conference in which I have met highly passionate individuals whose passion was highly contagious.

On the level of work, I have experienced as many highs as lows thanks to model completed and files corrupted overnight. In fact, half of my models got deleted in a JupyterLab crash 2 on June 21st. Thus, this thesis taught me perseverance in the work as a model which took weeks to create can be reconstructed in 2 days time thanks to the knowledge and experience gained.

Finally, I would like to first thank Dr. Iamnitchi for this incredible opportunity at IDS. Conducting academic research was a dream for me ever since

I was 9. I cannot overstate the importance of her support and advice on a daily basis. Moreover, I want to thank Dr. Bertaglia for his help in reviewing my work and providing me with the dataset for my chapter on reproducability. Lastly, I wish to make a final thank you to my girlfriend and family for the constant support and joy through the good and the bad times, thank you.

Bibliography

[1] Stichting reclame code, 2022.

[2] Thales Bertaglia. Influencer self-disclosure practices on instagram: A multi-country longitudinal study, 2023.

[3] Thales Bertaglia, Stefan Huber, Catalina Goanta, Gerasimos Spanakis, and Adriana Iamnitchi. Closing the loop: Testing chatgpt to generate model explanations to improve human labelling of sponsored content on social media, 2023.

[4] Dominic J Brewer and Patrick J Mcewan. *Economics of education*. Acadmic Press, Oxford, Uk ; San Diego, Ca, 2010.

[5] CrowdTangle. Crowdtangle — content discovery and social monitoring made easy, 2023.

[6] Lucas Machado de Oliveira and Olga Goussevskaia. Sponsored content and user engagement dynamics on instagram, Mar 2020.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[8] ED Diener and Carol Diener. The wealth of nations revisited: Income and quality of life, Nov 1995.

[9] Blake Droesch. Is everyone on instagram an influencer?, Mar 2019.

[10] Sahar A. El Rahman, Feddah Alhumaidi AlOtaibi, and Wejdan Abdullah Al-Shehri. Sentiment analysis of twitter data. In *2019 International Conference on Computer and Information Sciences (ICCIS)*, pages 1–4, 2019.

[11] Adrian Erasmus, Tyler D. P. Brunet, and Eyal Fisher. What is interpretability? *Philosophy Technology*, Nov 2020.

[12] Jana Gross and Florian von Wangenheim. Influencer marketing on instagram: Empirical research on social media engagement with sponsored posts, Oct 2022.

[13] HuggingFace. bert-base-multilingual-uncased · hugging face, Apr 2023.

[14] HuggingFace. bert-base-uncased · hugging face, 2023.

[15] HuggingFace. Transformers, 2023.

[16] Steven J. Kamper. Generalizability: Linking evidence to practice, Jan 2020.

[17] Seungbae Kim, Jyun-Yu Jiang, Masaki Nakada, Jinyoung Han, and Wei Wang. Multimodal post attentive profiling for influencer marketing, 04 2020.

[18] Seungbae Kim, Jyun-Yu Jiang, and Wei Wang. Discovering undisclosed paid partnership on social media via aspect-attentive sponsored post learning. WSDM '21, page 319–327, New York, NY, USA, 2021. Association for Computing Machinery.

[19] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv:1609.02907 [cs, stat]*, Feb 2017.

[20] Justin Kitzes. *Practice of reproducible research - case studies and lessons from the data-*. University Of California Press, 2017.

[21] PyTorch. Pytorch, 2023.

[22] Meta AI PyTorch. torch˙geometric.nn.models.gcn — pytorch geometric documentation, 2023.

[23] Daphne Raban and Avishag Gordon. The evolution of data science and big data research: A bibliometric analysis, 01 2020.

[24] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision, 2015.

[25] Google Trend. Google trends, 2023.

[26] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.

[28] Cheng Wang, Delei Chen, Lin Hao, Xuebo Liu, Yu Zeng, Jianwei Chen, and Guokai Zhang. Pulmonary image classification based on inception-v3 transfer learning model. *IEEE Access*, 7:146533–146541, 2019.

[29] M.E. Whitman and H.J. Mattord. *Principles of Information Security*. Cengage Learning, 2021.

[30] WSDM. Home — 14th acm international wsdm conference, 2021.

[31] Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. Listwise approach to learning to rank: theory and algorithm. In *International Conference on Machine Learning*, 2008.

[32] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. Exploring visual relationship for image captioning, 2018.

[33] Koosha Zarei, Damilola Ibosiola, Reza Farahbakhsh, Zafar Gilani, Kiran Garimella, Noel Crespi, and Gareth Tyson. Characterising and detecting sponsored influencer posts on instagram, 2020.

[34] Wayne Xin Zhao, Jing Jiang, Jianshu Weng, Jing He, Ee-Peng Lim, Hongfei Yan, and Xiaoming Li. Comparing twitter and traditional media using topic models. *Lecture Notes in Computer Science*, page 338–349, 2011.