POLITECNICO DI TORINO

Master's Degree Thesis in Data Science and Engineering

Master's Degree Thesis

Micro Influencer Classifier: an academic and economic approach

Supervisor

Prof. Luca ARDITO

Dott. Simone LEONARDI

Candidate

Paolo FIORIO PLA

April 2022

Abstract

The advent of social networks in the last decade has significantly changed the concept of popularity and its relationship with the public, allowing a new way of managing the market. A new key figure has emerged, the influencer: someone who has the power to affect others' purchasing decisions because of its relevance, knowledge and relationship with its audience. In this scenario, brands and small businesses are increasingly interested in collaborating with social media influencers because these partnerships can remarkably increase their exposure.
In an ever-widening roster, brands are looking for improved ways to identify suitable influencers; this is even more challenging with micro influencers, which are more affordable but difficult to discover.
Micro influencers are not prominent figures, celebrities or world-renowned experts; they specialize in a particular topic and share content about their interests only. This results in creating an hyper-engaged audience; in this way, if a company works with a highly-relevant micro influencer, it can significantly extend both the reach and the public engagement.
Micro influencers represent the emerging trend in social media marketing. Their high Return On Investment(ROI), commitment and persuasive power in their communities make them a very desired figure on the market.

The aim of the thesis is to face this individuation challenge by providing a framework for both academic and economic use.
Two different social networks are investigated: Twitter and Instagram.
Considering the difficulty in finding already preconstructed datasets with the specificity required by this thesis, the academic approach starts with the creation of ad-hoc datasets for both social networks, given an heterogeneous list of topics as input. For every topic a balanced mixture of micro and not micro influencers is selected and extended with the most suitable user's metrics concerning both the general account metadata and the effective engagement with the audience.
Specifically for the Instagram case, posts' descriptions are enriched with text obtained through Image Captioning methods applied on their respective pictures.
Once acquired the text, the calculation of the frequency of use of the topic-word among all posts is carried out, enriching the list of parameters. After an analysis based on the distributions of all the micro influencers' evaluated metrics, each user receives a score based on its positioning inside the respective distribution of each metric. In this context, only the user having a score higher than the dataset's average score receives the status of micro influencer for its specific topic.

A further analysis based on Natural Language Processing (NLP) methods is adopted to better understand the communication techniques that characterize micro influencers either for tweets and Instagram posts. The available text is subjected, in fact, to a cleaning and preprocessing step, comprehending: language detection, emoji's translation, punctuation and stopwords removal. With the preprocessed text a final step for each user to complete the dataset's creation is made: a sentiment analysis returning a positive-neutral-negative score to better understand their communication techniques.
Once the extended dataset has been built, a selection of the best classification model is performed, concluding the academic approach.
In this work, the classifier that guarantees for both social networks the best performances is the eXtreme Gradient Boosting (XGBoost) with an accuracy that reaches 100% for the training set and overcomes the 90% for the test set. Specifically, XGBoost classifier is a decision-tree-based ensemble Machine Learning algorithm that uses a gradient boosting framework.
As the last step, the selected model gets involved in the commercial outcome of this work: a framework that allows the brand to insert as input users and topic of interest, retrieves all the necessary data and evaluates if each user can be considered as general micro influencer and, most of all, as micro influencer for that specific topic by publishing as output the results.
Despite the limited amount of data available due to social networks restrictions, the framework reaches satisfactory results which can be improved in future works developing less constrained libraries.

Contents

List of Tables

List of Figures

Acronyms

AI
Artificial Intelligence

API
Application Programming Interface

CNN
Convolutional Neural Networks

COCO
Common Objects in Context

DL
Deep Learning

FN
False Negative

FP
False Positive

GPT
Generative Pre-trained Transformer

ML
Machine Learning

MLP
Multi-Layer Perceptron

NLP

Natural Language Processing

OSN

Online Social Network

ROI

Return On Investment

RNN

Recurrent Neural Networks

SGD

Stochastic Gradient Descent

SVM

Support Vector Machine

TF-IDF

Term Frequency–Inverse Document Frequency

TN

True Negative

TP

True Positive

UGC

User Generated Content

XGBoost

eXtreme Gradient Boost

Chapter 1

Introduction

This first chapter provides a general introduction to the work provided in this thesis project. An overview of the social media role inside marketing sector is made, going then specifically to investigate the figures of influencers and micro influencers, by considering both their relationships with brands and public.
The focus is then concentrated on the motivations that characterized this work and on its possible purposes, both on academic and economical side.
Finally, a short summary of the chapters' organization is proposed, to better understand the flow of this thesis.

1.1 Motivation

The rising power of social networks in the last decade has significantly shifted every method of communication from the most direct between acquaintances to the widest between brands and customers.
This translates into an evolution of the meaning of popularity, now more closely linked to direct contact with the public and even more capable of having considerable relevance also from a commercial point of view.
Social media has become the most popular and influential virtual environment where the idea of a platform is no longer based only on social networking but also on a new and better way of digital advertising. In this scenario, social media enriches the direct relationships between customers and brands, overcoming the classical one-sided advertising methods, devoid of direct interactions.
In the recent years has become even more common for consumers to learn about brands or companies on socials as it is through television and radio, even if the value of latter spots are deflating and the pandemic situation accelerated this decrease, shifting the focus on the digital space.
Consumer use of social media to discover or learn about new products or services

increased especially for Generation Z and Millenials that are strictly more related with the use of social networks. As the spending power of these digital natives increases, the size of social networks will also continue to grow, reaching over 308 million social network users in the US by 2023, according to [1].

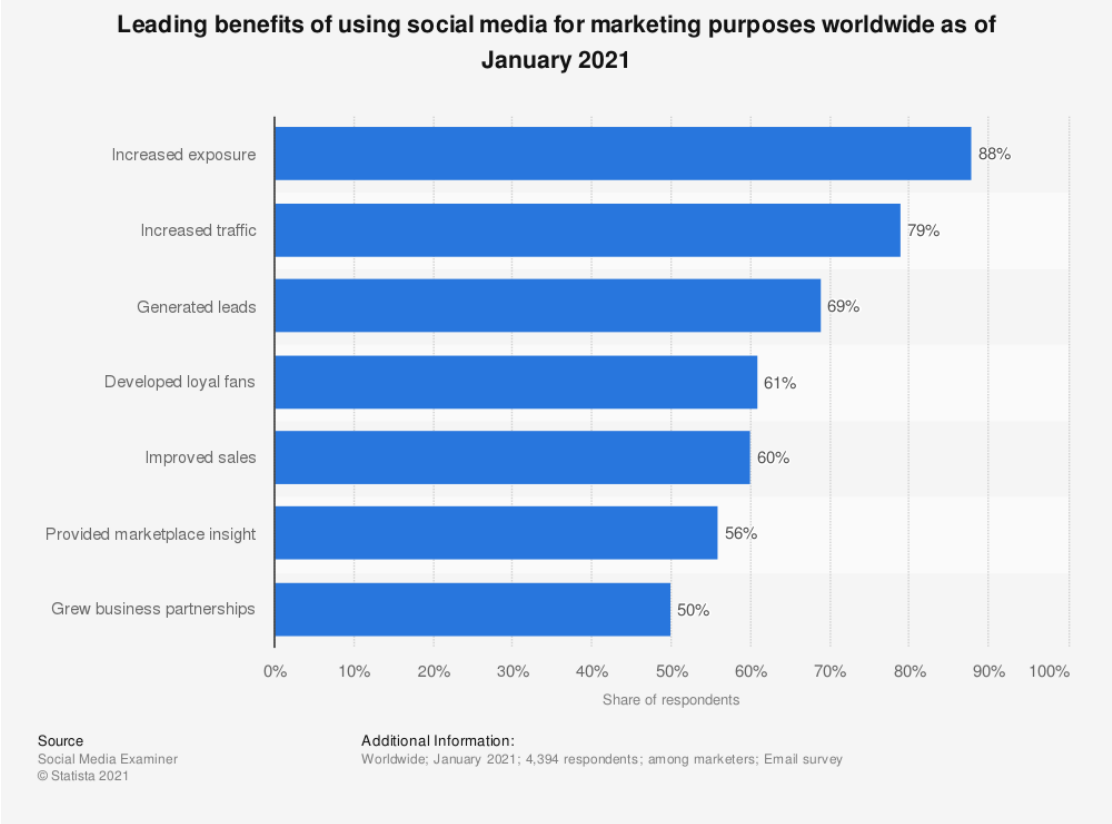

Figure 1.1: Social media marketing benefits

As reported in figure[1] 1.1, a worldwide survey proposed to marketers in 2021 states that the leading benefits of using social networks for marketing purposes are:

- **Increased exposure:** the importance of using the right platforms, the most frequented by customers in order to reach out the desired target audience at a more effective rate;
- **Increased traffic:** as described by DeMers[2], a wider presence of the brand on social media, related with an high quality of its contents, may only lead to an higher visibility generating an increasing inbound traffic;

[1]Figure taken from `https://statista.com`

- **Generated leads:** as explained in [3], external competitive pressure plays a prominent role in a brand's decision to invest on digital media for marketing purposes and this leads to the creation of a strong connection with audiences, fundamental to create an impression on their minds that puts the brand on top of their thoughts whenever they search or think about buying a similar product;

- **Developed loyal fans:** as in [2], a constant presence on social media provides a strong loyalty from the fans; the higher the loyalty of fans, the higher the possibility to receive positive reviews with other customers resulting in a further growth of the fandom;

- **Improved sales:** thanks to the ability to directly connect potential customers to products and services, social media advertising can be the quickest and most direct channel for brands to increase sales;

- **Provided marketplace insight:** keeping track of consumers' habits lets the brand to establish with them stronger bounds providing the right offers, even anticipating their own wishes to acquire;

- **Grew business partnerships:** the increased visibility opens the doors to new scenarios, not only from the costumer side but also from the strategic brand to brand alliances perspective, providing new opportunities on the market. In fact, as described by Van der Vlist and Helmond in [4] , partnerships between companies simultaneously make data widely accessible and exclusive, but also make it more difficult for new competitors to join because of the solid consolidation of both infrastructural and strategic power.

Following these results provided in fig.1.1, it becomes clear how the presence on social media of a brand can generate and determine a clear evolution of its image on today's market. It is important to use platforms that are commonly used by customers so that the target audience can be reached out at a more effective rate. Social media marketing and advertising can help the brand in increasing its ROI, as the cost of advertising on these social media is often less than the return, giving improved revenues. Larger is the public that the brand is able to reach within seconds through its campaigns, the lower the costs involved in achieving it. In this optic, brands and small businesses are increasingly interested in expanding their visibility and to achieve this become of relevant importance the collaborations with the new key figure that emerged in the digital era: the Influencer.

1.1.1 The role of Influencers

The definition of Influencer provided by the Cambridge dictionary [5] is:

> someone who affects or changes the way that other people behave, a person who is paid by a company to show and describe its products and services on social media, encouraging other people to buy them.

The fundamental aspect that makes the influencer one of the most important key figures of the daily market, in fact, is its power to affect others' purchasing decisions because of its relevance, knowledge and relationship with its audience.
Influencers built day by day a reputation with their public, thanks to their knowledge and expertise on one or more specific topics. Their ability to regularly feed their public's homepages generates loyalty, enthusiasm and a strong bond in their followers. These strong relationships makes them preferred also over the celebrities, in fact, as demonstrated in [6], participants that took part in this experiment identified themselves more with influencers than celebrities by feeling more similar to them and by trusting more in their storytelling and advertising skills.
The relevance of this figure is highlighted also in [7] , where the study demonstrates that depending on the posts shared by influencers, consumers are impacted at four levels: brand preference, increase in brand awareness, subject matter expertise and preference. From this perspective, a successful influencer marketing campaign for each brand involves a structured identification practice to obtain the right type of influencer who will offer curated advice, stories, and suggestions to create engagement with the audience.
Strictly related to this solid bond with the public there is an increasing power in the commercial sector, as depicted below in figure[2] 1.2. This study underlines the growing power that influencers reached in the last period, even doubling their value in the last three years and laying the foundations for further growth in the coming years.

[2]Figure taken from `https://statista.com`

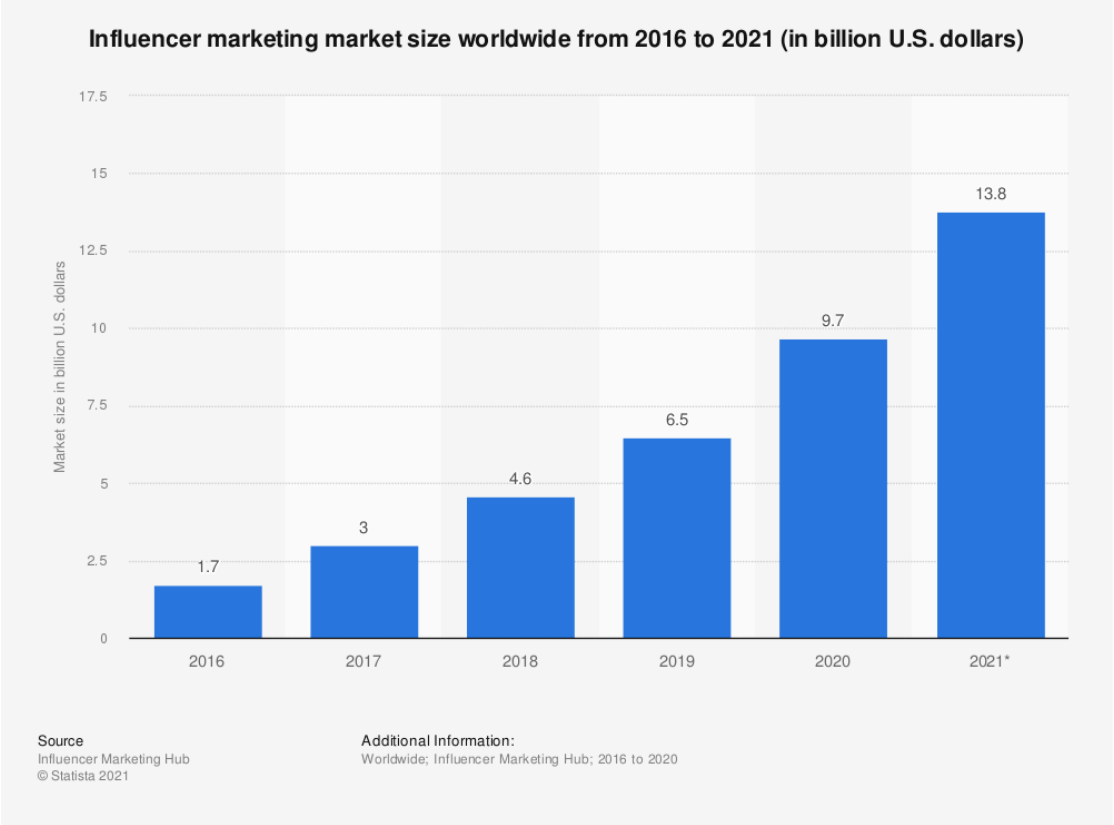

Figure 1.2: Influencers marketing value

Despite the increasing importance of the influencer figure, its subclasses have not been defined in a unique way but there is still a multiplicity of interpretations that try to adapt to the developments of the ever-changing social world. The following image[3] 1.3, does not represent the sub-classes adopted in this thesis but has been inserted to underline a specific behaviors inherent in the users' choices: despite the presence of influencers more and more followed and growing in numbers, such as *Macro* and *Mega* characterized by having a catchment area higher than 500000 users, the social media users tend to follow influencers with lower numbers like *Nano, Micro* and *Mid-tier* ones.

[3]Figure taken from `https://statista.com`

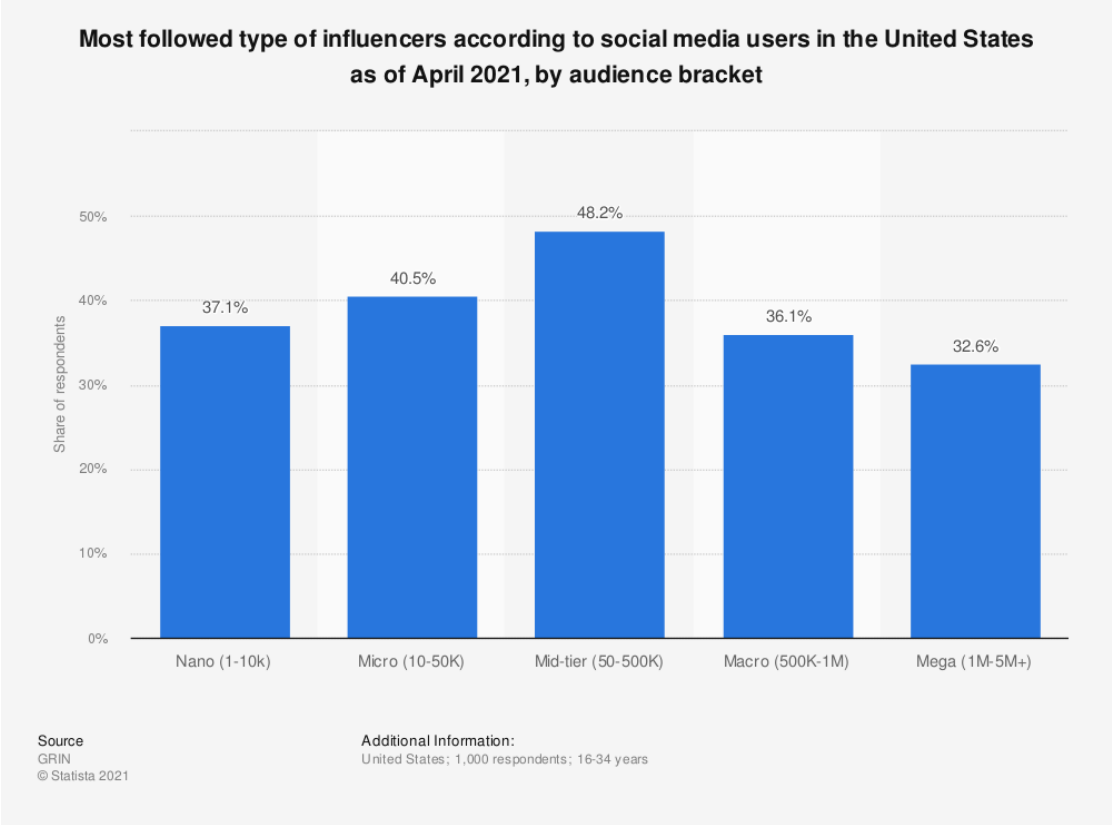

Figure 1.3: Different types of influencers

To better reflect the users' preferences depicted in the figure above, for this thesis focused on the micro influencers has been preferred an intersection of the two most followed ranges of followers: the best solution that suits this idea is the one proposed by Brewster and Lyu in 2020 [8] and visible in table 1.1 where the micro influencer are characterized by a number of followers between 5000 and 100000.

	NANO	*MICRO*	*MACRO*	*MEGA*
FOLLOWERS N°	<5000	5000 - 100000	100000 - 1000000	>1000000

Table 1.1: Different types of influencers

The following section will be devoted to the discovery of the key figure of this thesis, the micro influencer, and on its importance in the social media marketing sector.

1.1.2 Why Micro Influencers?

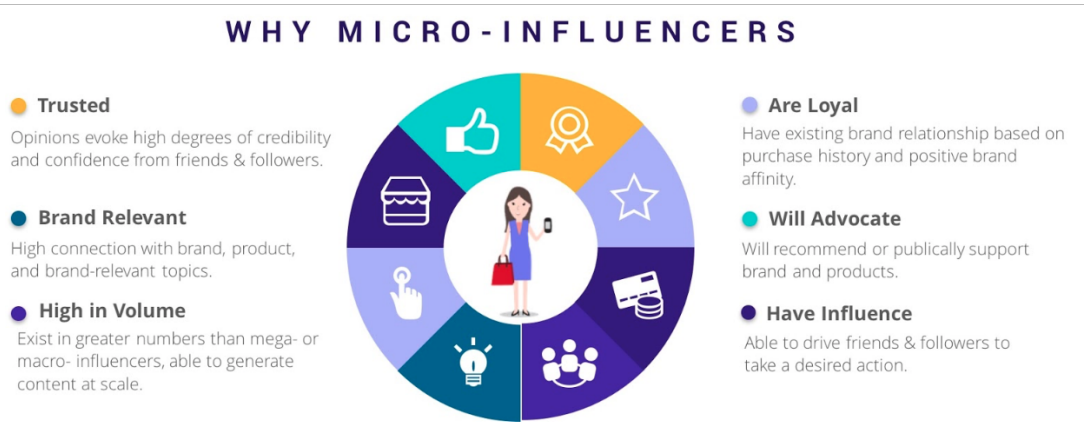

Figure 1.4: Reasons why micro influencers are preferred

The possible advantages for a brand deriving from the involvement of a micro influencer in its campaigns, preferring it to more famous people, has been the main topic of numerous research studies.
Starting from 2013, Wei and Lu [9] depicted the image of a common social network user as a person that tends to be wary of celebrities, as they take for granted an economic advantage for the promotion of certain products and, consequently, would not express their personal opinion. Differently from celebrities, influencers with a lower audience seem to be able to remain authentic, even if users are aware that they can too receive a remuneration. With the passing years this gap between celebrities and influencers has begun to be perpetrated also inside influencers' sub sections: the increasing success of the macro and mega influencers distanced them from the public due to an excessive fame that brought them closer and closer to the status of celebrity.
In support of this concept, Weinswig [10] in 2016 underlined an increasing trend followed by brands to focus more on quality rather than quantity, giving more and more space to collaborations with influencers with less following, validating in this way multiple benefits for the same brand such as: a unique and authentic point of view, a deeper narrative and the possibility to reach a more personalized audience of possible customers.
The trend expressed in this article is of a future in which they will continue to be preferred by brands thanks to their authenticity that allows them not to reach a celebrity status which is already associated with the most well-known influencers. [11] from 2020 compared the figures of macro and micro influencers underlying that, while the first is perceived by the public as more admirable through a more

professional image, the latter is associated with a simpler image having a closer affinity with the consumer thanks to its friendliness, closeness and naturalness.

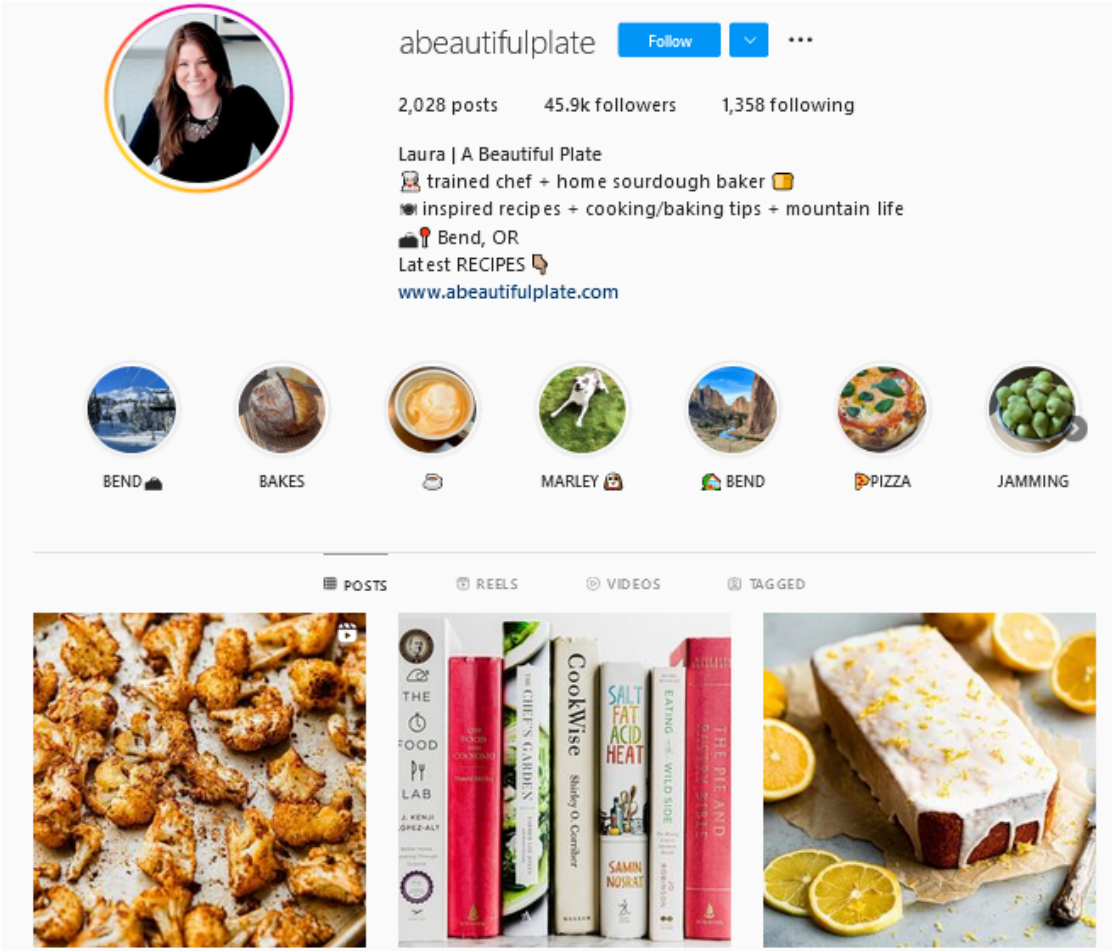

Figure 1.5: Example of food micro influencer

Nowadays, micro influencers (example in figure[4] 1.5) represent the emerging trend in social media marketing. Their commitment, high ROI and persuasive power in their communities make them a very desired figure on the market. Micro influencers are not celebrities, prominent figures or world-renowned experts; they specialize in a particular topic and share content about their interests only. This flows in the creation of an hyper-engaged audience which is tempting to many brands because if a company works with a highly-relevant micro influencer, it can significantly extend both the reach and the public engagement.

[4]Image taken from `https://www.instagram.com/abeautifulplate/`

In an ever-widening roster, where for brands it is already difficult to identify peculiar influencers, it is even more challenging with micro influencers, which are more affordable but difficult to discover.
Summing up, the main characteristics that make the micro influencers such a requested figure on the modern market are reported in the image[5] 1.4 and described according to the Forbes article [12]:

- **Trust:** the expertise and passion about their specific topic demonstrated day by day created in their audience an high confidence rate, destined to increase over time. With an higher trust the followers are likely to comment, engage and listen their advice;

- **Brand relevance:** if a brand is oriented on a specific topic, a micro influencer passionate about that area may be the best target for a highly curated advertising, not possible with the more generalist macro influencers. In this optic, micro influencers are more likely to partner with brands they actually love, providing higher support and sharing experiences with their audience;

- **High volume:** their lower number of followers related with the fact that is one of the most followed influencer sub-classes, as seen in figure 1.3, are indicators of a constantly increasing presence on social networks;

- **Loyal and avocation:** due to their small-size audience they recognise in brands a growing possibility, providing great efforts to publicly support the brand in order to create stronger marketing relationships, increase their public and demonstrate their working skills to the entire social media marketing sector;

- **Strong influence:** their peculiar work on a specific topic and their ability to entertain the public with their own passions creates an hyper-engaged audience more willing to follow their advice;

- **Cost:** the smaller size of their audience is directly proportional to their sponsorship costs, resulting more affordable partners for brands;

- **Multi market reach:** lower costs mean also that the same budget that could be used for a single macro influencer may be split between different micro influencers able to reach heterogeneous audiences.

[5]Figure taken from `https://influencerstrategies.com/`

1.2 Thesis purposes

The previous chapters explored the importance of social media marketing, analyzing the impact on this sector provided by influencer and specifically by one of its subclasses: the micro influencer.
A further analysis has underlined the main characteristics that make the micro influencers the most wanted on the market, being preferred to much more prominent figures known by the general public. Despite their relevance, one of the aspects that discourages in-depth study and research is their difficult availability.
This thesis main purpose is to facilitate this research by providing two different frameworks:

- **Academic framework:** the first part of this work is focused on the data scraping, fundamental for the academic research.
  Starting from the available Python libraries to handle data from social networks, the code allows a detailed search starting from the desired topics, providing a list of users eligible for the role of micro influencer and not, based on a series of ad hoc metrics that go beyond the simple number of followers. Successively the available users' posts are put under the magnifying glass, providing both a sentiment analysis and an evaluation of the affinity to the topic. Finally, all the obtained data become the basis of a search for an adequate classification model both for micro influencers in general and for those specific to the topic;

- **Economic framework:** the second part exploits the best model obtained from the academic side and tries to answer to a specific brand's question: can this user be considered a micro influencer and, most importantly, a micro influencer suitable for the social media campaigns concerning my topic?
  To obtain this answer the framework receives as input both the user and the topic; then, following the academic approach workflow retrieves all the necessary user data and provides the proper analysis for the sentiment of each post and on their affinity with the selected topic. Once completed the user dataset construction, all the obtained parameters are given as input to the best performing model of the academic part in order to obtain the results of the micro influencer and micro topic influencer classification.

Summing up, the main purpose of this thesis work is the creation of frameworks that can be used both in the academic and commercial fields to facilitate the search and identification of the micro influencers best suited to the requests.

1.3 Chapters organization

Chapter 1 proposes a general introduction on the reasons that motivated this work, starting from the importance of social network marketing for brands, passing through the role of influencers in this area and then going to investigate the reasons that distinguish the importance of micro influencers in social networks of today.
Chapter 2 gives an overview on the state of the art research on which this thesis is based. The first faced topic is Social Network Analysis with its main approaches proposed in literature. Then the focus goes over influencers detection methods, considering different papers that dealt with this argument and their proposed metrics that inspired the ones proposed in this thesis.
Finally, a last section focuses on sentiment analysis and on how the literature proposed different approaches to realize it.
Chapter 3 presents the main pipelines adopted in this thesis work and analyzes the theoretical aspects on which they are based.
Starting from data collection methods for both Instagram and Twitter with their related libraries, the chapter goes on with the image captioning method adopted for Instagram posts. Successively, the text preprocessing phase is deepened until it leads to sentiment analysis.
Finally, a general overview on the classification in machine learning is treated for then going into specifics with the models adopted and the methods and metrics used to select the best of them.
Chapter 4 gives an overview of the practical implementations of the work. Data collection part focuses on the informations obtainable from social networks libraries and on their relationships with the metrics adopted in classification.
After a brief review of image captioning methods, the score methods to appoint users as micro topic influencers are presented.
Finally, an in-depth analysis on the code concerning text preprocessing, sentiment analysis and classification model selection is provided.
Chapter 5 resumes the results obtained in the classification model selection part comparing the main metrics and selecting the best one for each social network involved. This part proposes also a comparison between the results obtained using two different datasets available for the image captioning section of the Instagram framework.
Chapter 6 discusses the main difficulties met during the work with social networks libraries and how they have been handled.
Chapter 7 analyses the main application of this work, the so-called economic approach. This part focuses on how the models obtained in the academic approach can be integrated in a new framework able to predict the value of a specific user and its possibility to be a general micro influencer but also a micro influencer for a specific topic, inserted as input of the framework.

This approach has its fundamentals on the methods presented in the academic approach but integrates them by providing a way to obtain some practical results that can be adopted by brands in daily social media influencers research.
Chapter 8 gives a final recap of the work done and proposes some ideas on possible future developments of the project and on possible different ways of use.

Chapter 2

State of the art

This state of the art chapter provides a review of the relevant contributions from the existing body of the literature.
Considering the main topics of this thesis, this section provides insights on some themes that can be considered essential.
The first review concerns some of the general aspects that characterize the Social Network Analysis, the approaches that characterized it during the years and its main applications.
Successively, an in depth insight addresses the more specific issue of identifying influencers and how modern research has adapted to the more restricted area of micro influencers providing ad-hoc metrics and algorithms. These different ways of interaction with the theme have influenced this work in the formulation of some specific metrics that characterize the core of the proposed framework.
The last section provides an overview of sentiment analysis fundamentals, its main approaches and its different methods of use.

2.1 Social Network Analysis

The detection of influencers is a topic of research in Social Network Analysis (SNA) that has interested the attention of many researchers groups given their increasing importance in the socio-economic world, their diffusion and consequently an exponential growth of available data accompanied by complex interpersonal interactions [13]. Despite the efforts made in this research there still lacks a universal criteria of influence since literature on influencer detection is composed by an elevated number of heterogeneous methods. For Räbiger and Spiliopoulou [14] social networks, as visible in figure[1] 2.1 are a combination of nodes and connections, in which each node indicates a user, and connections represent all the relationships between users. The corresponding graph may be modified because of the network or the problem modeling. Many influencer detection methods use the topological structure of the network and user interactions to identify the influencers.

Figure 2.1: Graph vision of a social network

[1]Image taken from `https://medium.com/analytics-vidhya/`

According to Lu et al. [15], the current influencer detection approaches can be divided into:

- influence maximization approaches: their target is the identification of influential nodes under a given diffusion model [16] in order to achieve a global optimization;
- influence measurement approaches: differently from the previous, are microcosmic problems, which target on developing an influence measurement method to detect influencers individually.

Influencer detection has been raising research issues and receiving attention in a multitude of domains such as economics, psychology, network sciences. This variety of domain has also increased the number of conventional methods for identifying individual influencers, classifiable as centrality based methods, node operation methods, diffusion-based methods and machine learning methods.

Centrality based approaches

These methods are devoted to the detection of an influencer only basing their computations on structural information. Centrality is defined in graph theory as a measure of the importance of a given node within a graph. Despite the large number of centralities described in literature, all these methods have one obvious limitation, that is a centrality which is optimal for one application is often suboptimal for a different application. Since different online social networks have different topological structure, centrality based approaches cannot maintain their effectiveness and accuracy when used on online social networks(OSNs) [17].

Node operation approaches

Nodes have the fundamental role of connecting different vertices, in this case users. When with their removal this link is broken, the network suffers from different problems related to stability and connectivity. Therefore, node operation approaches are proposed to find influential users by node removal and contraction [18]. Since the accuracy of node operation methods depend on the global input of the topological structure, also these approaches result to be limited in OSNs as the centrality based ones. Therefore, also node operation methods are lack of efficiency to be adapted onto large scale OSNs

Machine Learning approaches

Thanks to advanced machine learning algorithms, also in the influencer identification sector reached high relevance the adoption of automated classification methods.

For example, Fan et al[19] developed an approach able to find key players inside complex networks through the use of a deep reinforcement learning framework. Despite a fast detection of the target, these methods have a high dependency on the quality of the learning set and the size of the available data.

Diffusion Based approaches

As described in [20], a common approach to the influence maximization problem is to simulate influence cascades through the network based on the existence of links in the network using diffusion models. Even if these approaches are largely dependent on parameters, together with ML ones take in consideration users' behavior and information factors, which greatly improve the efficiency and accuracy on large scale networks.

2.2 Influencer detection

In recent years many studies have proposed to better adapt the modern technologies in the influencer detection frameworks, considering both realization in socio-psychological and economical spheres. The main researchers focus their work on users' general information and on studies related to their communication skills trying to extract the best patterns to establish the perfect parameters that characterize the influencer figure.
Gan et al. in [21] proposed a data-driven micro-influencer ranking scheme to solve the question of finding out the right micro influencer for a company. Brands and micro influencers are represented by fusing their historical posts' textual and visual information. To fuse visual and textual information is adopted a low-rank bi-linear pooling method[22]: is applied a linear transformation followed by a non-linear activation on each feature to reduce the difference between the size of two feature dimensions. To learn a brand-micro influencer scoring function a K-buckets sampling strategy with a listwise learning to rank model is proposed. With respect to every available brand a **competence score** is designed for micro influencers: specifically, are integrated engagement and relatedness into a this score of each micro influencer m_i respect to brand b_j:

$$cs(m_i, b_j) = \alpha Engagement(m_i) + (1 - \alpha)SIM(m_i, b_j) \tag{2.1}$$

Engagement is defined as the average number of likes and comments for the posts that micro influencer m_i used for brand b_j advertising:

$$Engagement(m_i) = \frac{AVE(likes + comments)}{followers} \tag{2.2}$$

The **similarity** function between two account representations is defined as:

$$SIM(m_i, b_j) = \frac{e^a(m_i) \cdot e^a(b_j)}{\|e^a(m_i)\| \cdot \|e^a(b_j)\|} \tag{2.3}$$

Given the final social account representation e^a defined as:

$$e^a = \left\langle \phi(e^t W_1^t + b_1^t) W_2^t, \phi(e^v W_1^v + b_1^v) W_2^v \right\rangle \tag{2.4}$$

where $\langle \cdot, \cdot \rangle$ is the inner product, ϕ is the non linear activation function, $W_1^t \in \mathbb{R}^{d_t \times d_{h1}}, W_2^t \in \mathbb{R}^{d_{h1} \times d_a}, W_1^v \in \mathbb{R}^{d_v \times d_{h2}}, W_2^v \in \mathbb{R}^{d_{h2} \times d_a}, b_1^t \in \mathbb{R}^{d_{h1}}, b_1^v \in \mathbb{R}^{d_{h2}}, e^a \in \mathbb{R}^{d_a}, d_{h1}, d_{h2}$ are the length of the hidden state vectors and d_a is the length of the final social account representation.
Finally, the K-buckets sampling method is adopted to select the samples. The bucket defines a certain pattern of positive-negative examples ratio in a sample, and the micro-influencer examples are then filled into buckets to create the length-K samples.

In the same year [23] developed a framework to identify influential users by examining their posts through text analysis and natural language processing. The main idea of this work is to identify influencers on Instagram relying only on the analysis of the User Generated Content (UGC) and not on the extent of user interactions. Given the posts text, after a preprocessing phase followed by a TF-IDF (term frequency–inverse document frequency) approach to weight each word is clear the different nature of captions and hashtags.
Considering these two types of available text data, their representation mechanisms are investigated separately:

- **captions** : for each word occurring more than 10 times in the corpus, a 100-dimensional vector is learned using either Word2Vec [24] or fastText [25]. Then the caption representation is produced by taking a weighted average over the word vectors, where the weight for each vector is the TF-IDF value of its corresponding word;
- **hashtags**: considering their sparsity, co-occurrence representations of all hashtags are used along with the Jaccard similarity metric (defined as the size of the intersection divided by the size of the union of the sample sets) to form an affinity matrix.

Finally, a binary classification problem is addressed using two support vector machine (SVM) classifiers, one for hashtags and the other for captions. The two classifiers are then combined in the kernel space by computing a weighted average over the individual kernels.

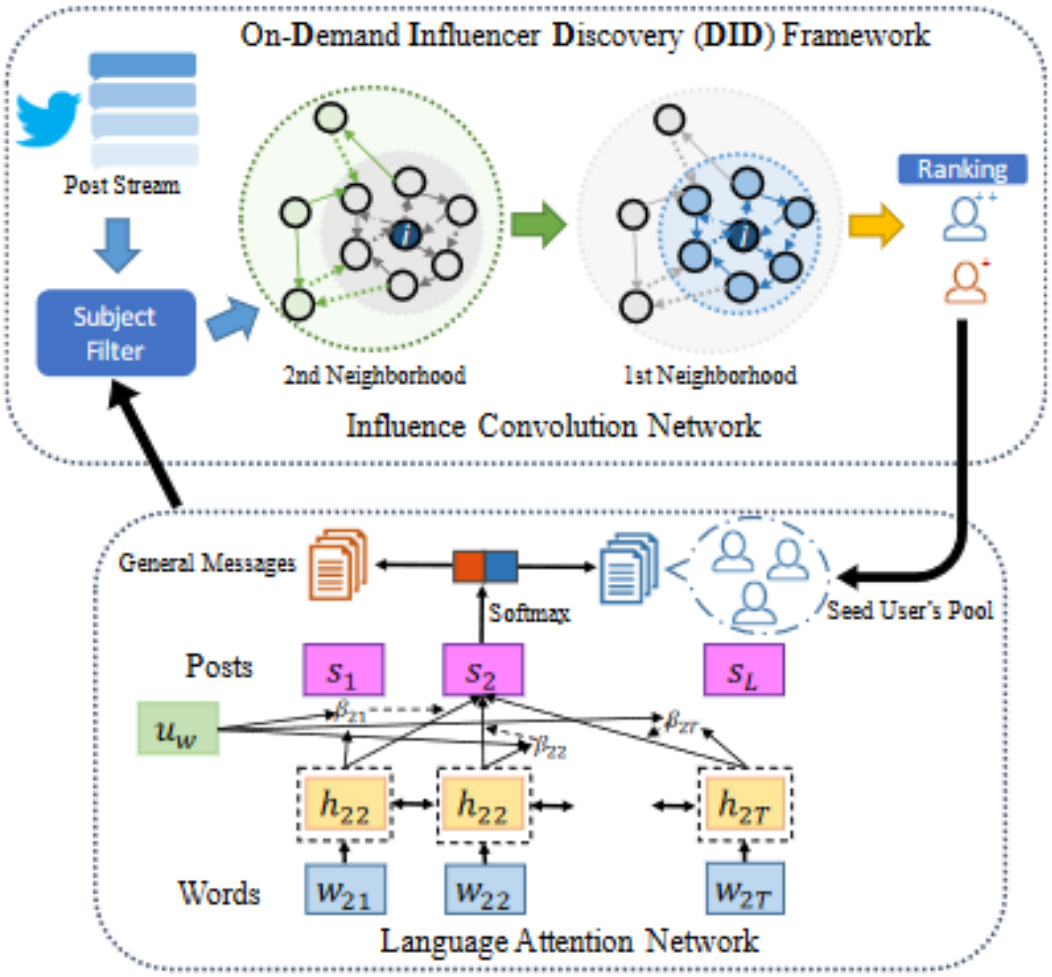

Figure 2.2: Overview of DID framework. [26]

Another detection idea proposed by [26] takes in consideration the use of keywords. This paper introduces the on-Demand Influencer Discovery (DID) framework (visible in figure 2.2) that is able to identify influencers on any subject depicted by a few user-specified keywords, regardless of its popularity on social media.
To identify between the multitude of users present on OSNs the specific-topic influencers, this model adopts a Language Attention Network to select social posts related to the given keyword and an Influence Convolution Network to mold the influence propagation on social media with neighborhood aggregation techniques. Inside Language Attention Network each word is converted into a one-hot encoding representation and all the words are embedded to vectors with an embedding matrix, then for each post, the word embeddings are fed to a bidirectional Recurrent Neural

Network to learn a hidden state of each word.
To derive the post representation,an attention layer is introduced to obtain a weighted sum of the hidden states from the network layer; finally a subject seed user pool is built to capture as much subject-related information in posts.
Following this idea: users with a larger neighborhood should have a higher influence score, considering only the first two hops; higher the concentration in keyword topics, the higher the influence score.
Recent studies from 2021 [27] provided a new way of identifying influencers by developing a multidimensional social influence (MSI) measurement approach, which can detect influencers more accurately compared to most existing approaches. This model, also evolved for the topic research (TMSI), takes into account three dimensional characteristics as shown in figure 2.3: structure-based, information-based, and action-based factors.

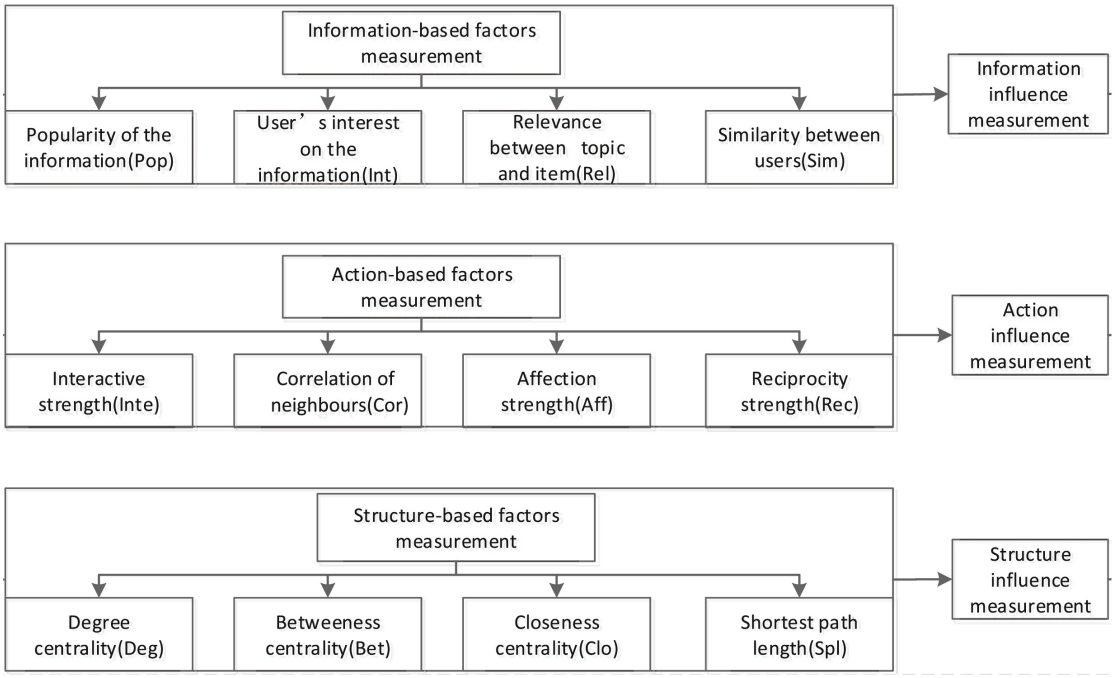

Figure 2.3: Dimensional characteristics of MSI measurement approach.

Structure factors measurements : concerns an individual's structural position in the network, that significantly affects users' information exchange behavior and users' status. Four factors are considered:

- **betweenness centrality:** *Bet(v)*, measures the average degree to which a given node lies in the shortest paths of other nodes;

- **closeness centrality:** *Col(v)*, are length-based measure that counts the length of walks;
- **degree centrality :** *Deg(v)*, counts the number of paths of fixed length beginning from a given node;
- **inverse shortest path length :** *Ispl(u,v)*, measures the indirect influence between two indirectly connected users.

Information factors measurements : estimate the contributions of users' posts. Four factors are considered:

- **popularity of information:** *Pop(v)*, the frequency of messages that contain keywords related to hot topics;
- **information type:** *Typ(v)*, the frequency of messages that contain multimedia or URLs;
- **relevance between user's messages and a topic:** *Relz(v)*, the ratio between intersection and union of topic's keywords and user's messages;
- **users' similarity:** *Sim(u,v)*, the ratio between intersection and union of messages from different users.

Action factors measurements: to determine the effect of users' actions(viewing, mentioning, tweeting, etc.) on their connection strength. Four factors are considered:

- **interactive frequency :** *Inte(u, v)*, to measure the effect of users' interactions;
- **correlation of neighbors :** *Cor(u,v)* is defined by the correlation of users' first-layer neighbors' ID list, to measure the similarity of users' friends;
- **affection strength :** *Aff(u, v)*, weighted sum of Aff@(u, v) and AffIP(u, v). Respectively, the degree of affection that arises between users sharing the same relationships and the correlation of two users' registered address to represent their affection strength based on location;
- **reciprocity :** *Rec(u, v)*, the bi-directional interaction frequency of two users.

All these metrics are combined for the TMSI model to define specific-topic dimensional influence measurements:

- **information influence :** $Inf(u, v) = Pop(v) \cdot Typ(v) \cdot Relz(v) \cdot Sim(u, v)$;
- **action influence :** $Act(u, v) = Inte(u, v) \cdot Cor(u, v) \cdot Aff(u, v) \cdot Rec(u, v)$;

- **structure influence :** $Str(u, v) = Bet(v) \cdot Col(v) \cdot Deg(v) \cdot Ispl(u, v)$;

Finally, these last influence measured are used to establish the influential strength of users' connection, depicted as:
$pi(u, v) = \beta_1 Inf(u, v) + \beta_2 Act(u, v) + \beta_3 Str(u, v)$ where β are weights.

2.3 Sentiment Analysis

Sentiment analysis is the multidisciplinary field of study that deals with identifying and analyzing people's emotions, sentiments and opinions about different entities such as specific topics, products, services, individuals. It is related to multiple fields such as natural language processing (NLP), computational linguistics, artificial intelligence and machine learning. The main purpose of sentiment analysis is the classification of writer's attitude into positive, neutral or negative categories[28] as visible in figure[2] 2.4.

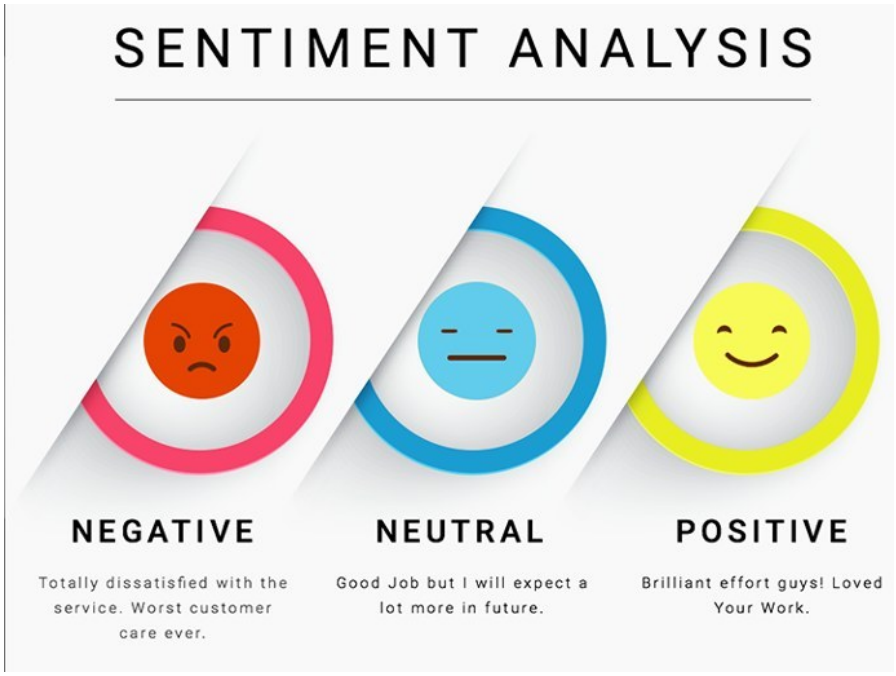

Figure 2.4: Sentiment analysis categories example

The advent and explosion of social media in the last years have remarkably increased the amount of data available on the web; billions of people have been

[2]Figure taken from `https://www.analyticsvidhya.com`

allowed to freely interact, express their thoughts and post all of this on their social pages. This countless unstructured amount of data has provided an opportunity to study and understand from both a social and economic side the thoughts and the habits of the population which resulted in the development of sentiment analysis. Especially from the commercial perspective, as in [29], sentiment analysis provided an huge help for the brands to understand customers' preferences by analyzing their reviews in order to ensure a better user-friendly experience.
Differently from a long history of studies in linguistics and NLP, the analysis of people's opinions and sentiments has been introduced only in the early years of the new millennium. However, since then, the literature witnessed an high quantity of studies due to several factors, including:

- the rise of machine learning techniques in NLP and information retrieval;
- the realization of the huge applications in industry that the area started to offer [30];
- wide access to datasets for training machine learning techniques;

This rapid academic growth of sentiment analysis topics correlated with the advent of the social media, have made it become the central point in the social media research [31]. These studies have steadily increased, to the point of branching out into various approaches and levels of specialization.
Nowadays, starting from the work of Tan et al. [32] sentiment analysis applications can be organized at four different levels:

- **sentence level** that detects positive, negative and neutral sentiment for each sentence;
- **document level** which detects the whole document sentiment as one unit or one entity positive or negative or neutral. For example given an hotel review, the task is to determine whether it expresses positive or negative opinions about the hotel;
- **aspect level** that is used in case of the availability of attributes inside text or post. Each attribute can have a different sentiment; this division in sub-problems results in a more challenging task than both document and sentence levels;
- **user level** which handles the social relationships between different users according to graph theory [32].

The continuous deepening of these levels over the years, correlated with academic developments in other AI sectors, has allowed the development of increasingly precise and performing approaches in sentiment analysis systems. Nowadays these approaches can be grouped into 4 different categories:

- **Lexicon approaches:** mainly rely on a sentiment lexicon which is a collection of known and pre-compiled words, phrases and even more complex structures (*corpus lexicon* [33]), or on dictionaries that measure the semantic orientation of words having a corresponding semantic polarity and sentiment strength (*dictionary lexicon* [34]). If each word is present inside the dictionary, this approach is then used to calculate the overall polarity/sentiment score;
- **Machine Learning approaches [35]:** are capable of extracting models and patterns in complicated data sets, including supervised learning, unsupervised learning, and semi-supervised learning methods.
  Supervised algorithms provide a sentiment classification by training models with labeled data, while *unsupervised* algorithms perform the same procedure but using unlabeled data, obtaining hidden structures from them.
  Semi-supervised methods takes advantage of training model using unlabeled data and can improve classification when labeled data are rare;
- **Deep Learning approaches [36]:** this techniques differently from ML ones automatically generate features not requiring feature engineering. In case of huge datasets this results in superior performance in many NLP tasks like machine translation, text summarization, question answering and of course sentiment analysis;
- **Hybrid approaches:** are the combination of Lexicon, Machine Learning and Deep learning approaches;

As already previously explained, social media platforms in the last decade have been definitely contributing to extending user-generated content. This made it possible to associate a detailed, sentimental response from each user interacting with it to each specific topic dealt with on the web.
Pathak et al. [37] to catch up with the speed of data generation generated on social media platforms, proposed a framework with the purpose of detecting the main topics of discussion and analyzing the sentiments of each user towards those topics. This paper proposes a deep learning based topic-level sentiment analysis model. Through this model, the authors presented a way to support dynamic topic modeling over streaming short text data and to perform a sentiment analysis at topic-level. The novelty of this approach is its focused work at the sentence level to extract each topic using online latent semantic indexing with regularization constraint and then the application of topic-level attention mechanism in a long short-term memory network to finally perform sentiment analysis.

Chapter 3

Approach

This chapter exploits the fundamentals of theory on which this thesis work is built. To perform the work in the most accurate way for both Twitter and Instagram OSNs, two different pipelines have been built: Twitter makes textual communication its strength while in the case of Instagram more prominence is given to the images. In this optic, the Twitter pipeline as depicted in figure 3.1 starts with the collection of data and a first study that evaluates specific parameters to identify micro influencers for the available topic. Later on users' tweets are preprocessed and studied with a sentiment analysis phase to provide further metrics useful for understanding their communication techniques. With these last obtained metrics the selection of the best classification model is performed.

Figure 3.1: Twitter pipeline

The Instagram pipeline as depicted in figure 3.2 starts with the collection of data, the image captioning phase to obtain text from the images and the first study that evaluates specific parameters to identify micro influencers for the available topic. Later on users' post descriptions and obtained captions are preprocessed and studied with a sentiment analysis phase to provide further metrics useful for understanding their sentiments expressed with both contents. With these last obtained metrics the selection of the best classification model is performed.

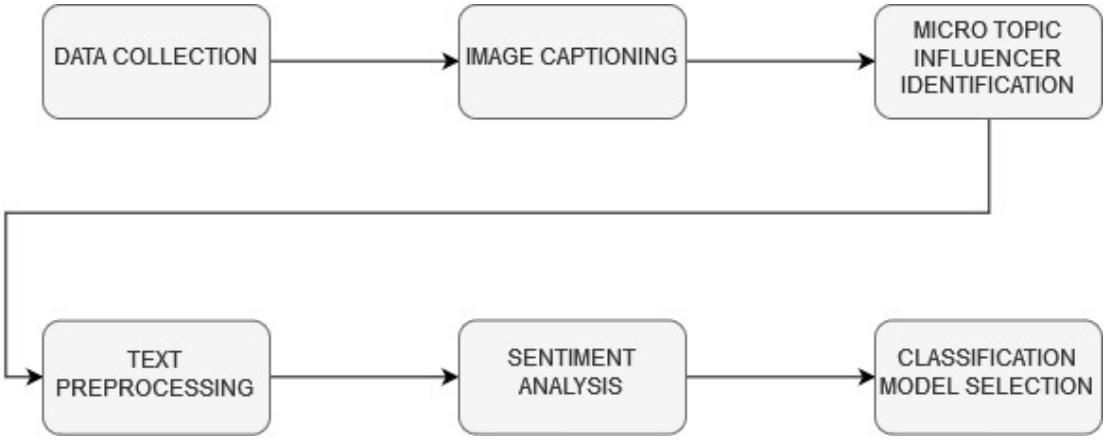

Figure 3.2: Instagram pipeline

3.1 Data Collection

Data Collection is a systematic process of gathering data from different sources to provide insights and answers for specific purposes like decision making or research. A high quality data collection provides the right information needed to answer questions, analyze business performance, predict future trends or scenarios. Depending on the purpose, two types of data can be collected:

- **quantitative data** is expressed using numbers, processed and analyzed through statistical methods;
- **qualitative data** is expressed using words, analyzed through categorizations and specific interpretation.

The data that can be collected from social networks belong to both categories. To better manage their research it is necessary to use specific APIs: to handle data coming from Twitter and Instagram this thesis takes advantage of the use of Tweepy and Instaloader.

3.1.1 Tweepy

Tweepy [38] is an open source Python library available to access Twitter API. This package includes a set of methods and classes that represent Twitter's models and API endpoints, handling various implementation details, such as: OAuth authentication, HTTP requests, data encoding and decoding, results pagination and real time streams of tweets. To use this library is requested a developer account, obtainable by specifying the reasons and purposes of use of this API on the Twitter Developer platform [39]. Once the registration as developer account is completed,

the creation of an app allows the generation of new keys and access tokens that are fundamental to obtain the platform access through the Python code.

3.1.2 Instaloader

Instaloader [40] is an open source Python library for accessing Instagram API. This package, even if not official, offers great functionalities to scrape Instagram data. Requiring only an active Instagram account, its methods allow the user to download public and private profiles, hashtags, stories, feeds and saved media, providing with each post the full list of its metadata.

3.2 Image Captioning

This technique is taken into account to maximize the information obtainable from Instagram posts. Image Captioning refers to the process of generating a textual description from a given image on the basis of the objects and actions present in it. Some examples are presented in image 3.3 taken from [41].

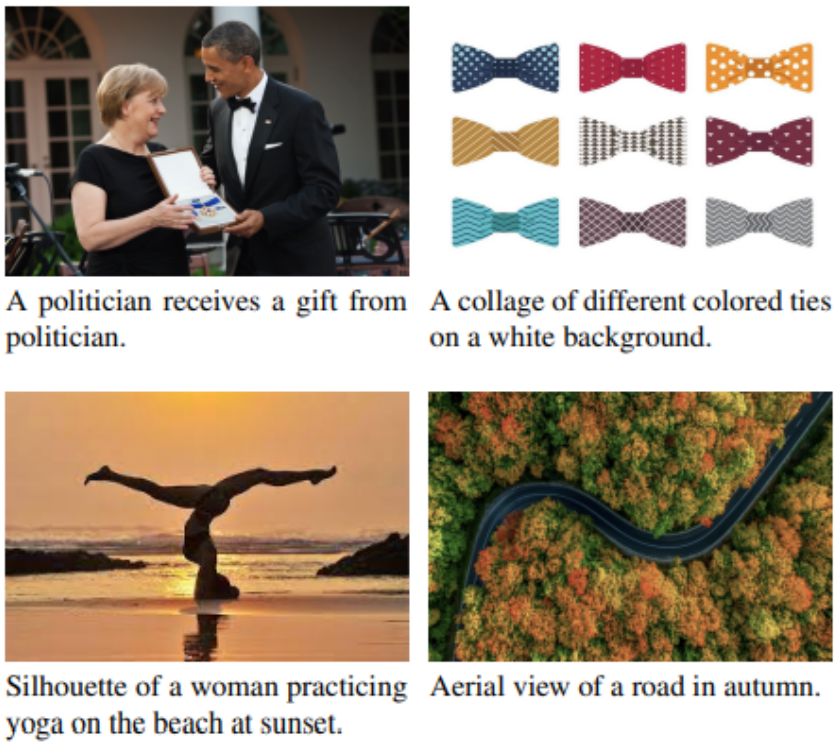

Figure 3.3: Image captioning example

Automatically describing the content of an image has become a fundamental problem in artificial intelligence, connecting computer vision and natural language processing tasks.
The main task of image captioning can be divided into two modules: a computer vision model which extracts the features out of the image, and a language based model which translates the previously obtained features and objects to a natural sentence. The computer vision part is typically performed with Convolutional Neural Networks (CNN) [42] while the language part is based on Recurrent Neural Networks (RNN) [43].
The image captioning part of this thesis is performed through the model presented by Mokady et al. [41] that guarantees a rather quick training to produce a competent captioning model. In fact, it achieves comparable results to state of the art methods while being simpler, faster and lighter.

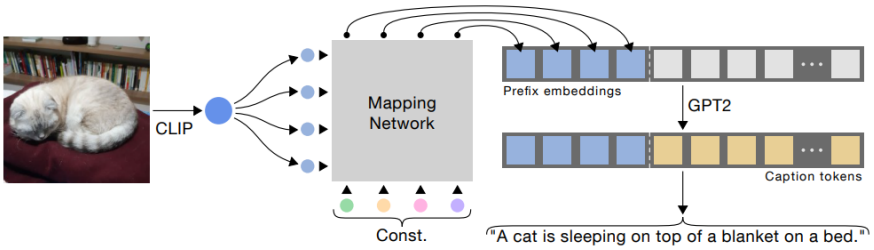

Figure 3.4: ClipCap model

The model works as presented in paper's figure 3.4 :

- the visual encoder of a pre-trained CLIP [44] model is used to extract visual informations from the given image;

- to map the obtained CLIP(Contrastive Language-Image Pre-Training) embedding to the embedding vectors a light mapping network is employed. For each caption this mapping network produces a prefix, a fixed size embeddings sequence concatenated to the caption ones;

- these embeddings are fed to the Generative Pre-trained Transformer GPT-2 [45] language model, fine-tuned with the mapping network training;

- starting from the CLIP prefix, the language model produces at inference step the captions. For each token, the language model outputs probabilities for every vocabulary tokens, which are then used to determine the next one by employing a greedy approach.

3.2.1 Datasets

The production of the final captions depends also on the dataset adopted. This model can be performed over two different datasets: COCO and Conceptual Captions.

COCO [46] stands for "Common Objects in Context" and is a large-scale object segmentation, detection and captioning dataset published by Microsoft. It results to be one of the most popular datasets used by Machine Learning and Computer Vision engineers for various computer vision projects. Between its main features can be mentioned the presence of 330K images (of whom 200K labeled), the distinction in 80 object categories and the detailed description of each image through 5 different captions. Figure 3.5 presents an example taken from [46].

Figure 3.5: COCO dataset example

Conceptual Captions [47] dataset provided by Google consists of 3M pairs of images and captions, harvested from the web and post-processed, representing a wider and more challenging variety of styles. Figure 3.6 presents an example taken from [47].

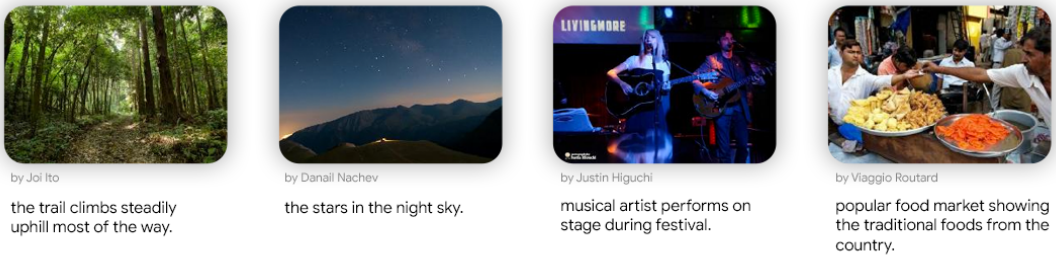

Figure 3.6: Conceptual Captions dataset example

3.3 Text Preprocessing

Natural Language Processing (NLP) is the process of extracting and analyzing information from the human language with the help of computer science and artificial intelligence. This technique comes from the semantic analysis of raw data based on the research for interconnection between the natural language and computer science.
Through the adoption of NLP methods, many real-world business problems can be solved such as summarizing documents, speech recognition, sentiment analysis, caption generator, neural machine translation and fraud detection. Textual analysis computed by NLP methods always requires some preprocessing steps; the ones adopted in this work are presented in the following subsections.

3.3.1 Language detection

The language detection process results to be of particular importance for the successive steps of text preprocessing but also for the definition of the prominent language of each user. Python offers a specific library, called langdetect [48], to perform this task through the *detect* method, as visible in the following code snippet 3.7 produced on Google Colab [49]:

```
from langdetect import detect
english_text = "This thesis is focused on micro influencers"
italian_text = "Questa tesi si concentra sui micro influencers"
print("Language of ",english_text, ": ", detect(english_text))
print("Language of ",italian_text, ": ", detect(italian_text))
```

```
Language of  This thesis is focused on micro influencers :  en
Language of  Questa tesi si concentra sui micro influencers :  it
```

Figure 3.7: Language detection example

The main turn-on of this library is its capacity to handle and detect 55 different languages using naive Bayesian filters with a precision that verges the 100% for 53 of them. The results are emitted in the form of ISO 639-1 language code.

3.3.2 Emojis conversion to text

One of the ways to express sentiment in social media is through the insertion of emojis inside tweets or posts description.
Sentiment analysis methods still lack a direct interpretation of these emojis and so, to facilitate their work, help can be provided by checking the possible presence of emojis and by converting them to text.
The Emoji [50] Python library provides a specific method, *demojize*, that can be used to handle this task as reported in the following image 3.8:

```
print(emoji.demojize('I really appreciate this place 👍 😊'))
```

```
I really appreciate this place :thumbs_up: :smiling_face_with_smiling_eyes:
```

Figure 3.8: Emojis conversion to text example

3.3.3 Text cleaning

Text cleaning is a process consisting of the preparation of raw text for NLP techniques so that it is most suitable for machine interpretation. Clean text can be considered human language rearranged into a format that machine models can understand.
Text cleaning can be performed with many different steps, the ones adopted in this work and most suitable for the sentiment analysis purpose are:

- **text normalization:** conversion of each word to its lower case;
- **URL removal:** substitution of classical link "https://www." with a blank space due to its uselessness in terms of communicative expression skills;
- **Irrelevant character removal:** substitution of numbers, symbols and punctuation with blank space because these elements are not relevant for the analysis;
- **Stopwords removal:** some words, called stopwords, are the most common in a language (for example “the”, “me”, “a”, “to”) and for this do not usually carry important meaning and usually are removed from text. The Python library stopwordsiso [51] allows to remove stopwords stopwords for multiple languages, using ISO 639-1 language code obtained from previous "Language detection" subsection;
- **Lemmatization:** this technique aims to remove inflectional endings only and return the base or dictionary form of each word, which is known as the lemma.

3.4 Sentiment Analysis

Sentiment analysis, also known as opinion mining or emotion AI, is a natural language processing (NLP) technique used to determine whether textual data is positive, neutral or negative.
This analysis is a contextual mining of text used to systematically identify, extract and study emotional states of mind and subjective information. It is often performed to help businesses monitor brand and product sentiment in customer feedback, understand customer needs in order to tailor products and services to meet them.
With the recent advances in deep learning models, the ability of algorithms to analyze text considerably improved, also concerning social media data. Online social networks, in fact, have become an important place of vent and discussion used by customers to express their opinions.

Python sentiment analysis is a methodology for analyzing a piece of text to discover its sentiment. It accomplishes this by combining machine learning and NLP methods. Many libraries with their respective methods provided ways to compute this analysis, from the simple binary polarity (positive, negative), to its development in a three-way sentiment(positive, neutral, negative), going also beyond polarity to detect specific feelings (happiness, sadness, etc), urgency (urgent, not urgent) and intentions (interested, not interested).
For this purpose, Hugging Face Hub [52] offers more than 215 sentiment analysis models publicly available and integrable on Python with just a few lines of code. The following are some of the most popular models available on the Hub are:

- **Twitter-roberta-base-sentiment[53]:** is the one adopted in this thesis and consist in a a roBERTa [54] model trained on 58M tweets and fine-tuned for sentiment analysis. With fine-tuning the roBERTa is enriched with additional training data to make it perform a second task, sentiment analysis in this case.
- **Bert-base-multilingual-uncased-sentiment [55]:** is a model developed for sentiment analysis on product reviews available for six languages: English, Italian, Spanish, Dutch, German and French;
- **Distilbert-base-uncased-emotion:** is a model fine-tuned for detecting emotions in textual data, including love, joy, sadness, anger, fear and surprise.

3.5 Classification

Classification is the process of categorizing a given set of data into classes that can be performed on both structured or unstructured data.
The goal of classification in machine learning is to generate a model able to assign the right class label to the right data. To obtain this result, a training set containing correctly labeled data is taken as input and used to teach the model how to classify. The same model is then used to classify data of a test set, containing items not yet labeled.
The main tasks addressed in this thesis concern the two classifications presented in figure 3.9: micro influencer or not micro influencer, micro topic influencer or not micro topic influencer.

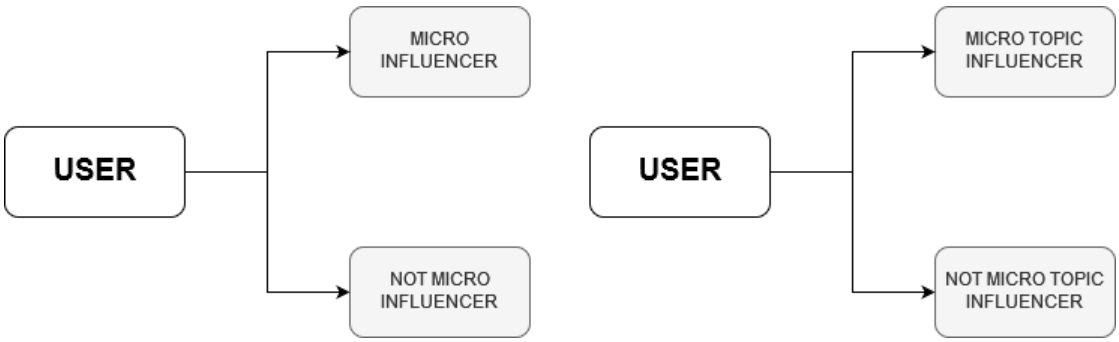

Figure 3.9: Classifications presented in this thesis

3.5.1 Models

In machine learning, classification is a supervised learning concept basically introduced to categorize a set of data into classes. Due to the high relevance and applicability of this concept, literature during the years provided an high quantity of different classification models, still increasing. Six of them have been selected to perform classification tasks in this thesis work: Random Forest Classifier, XGBoost, Support Vector Classifier(SVC), Multi-layer perceptron(MLP), Logistic Regression and Stochastic Gradient Descent(SGD).

Random Forest

Random forests or random decision forests is an ensemble learning method available for classification and regression purposes that operates by constructing a multitude of decision trees at training time. For classification tasks, the output of the random forest is the class selected by the majority of trees as visible in figure[1] 3.10.
This algorithm is called Random because the forest is randomly created with decision trees. To obtain its proper output, each node in the decision tree works with a random subset of features. Finally, the random forest combines the output of each individual decision tree to generate the final output through a mechanism of majority voting.
The advantage of this classification model is its higher accuracy with respect to simple decision trees due to the reduction in the over-fitting. The only disadvantage encountered with this classifier is its complexity in implementation that results in a slower real time prediction and in a more difficult interpretability.

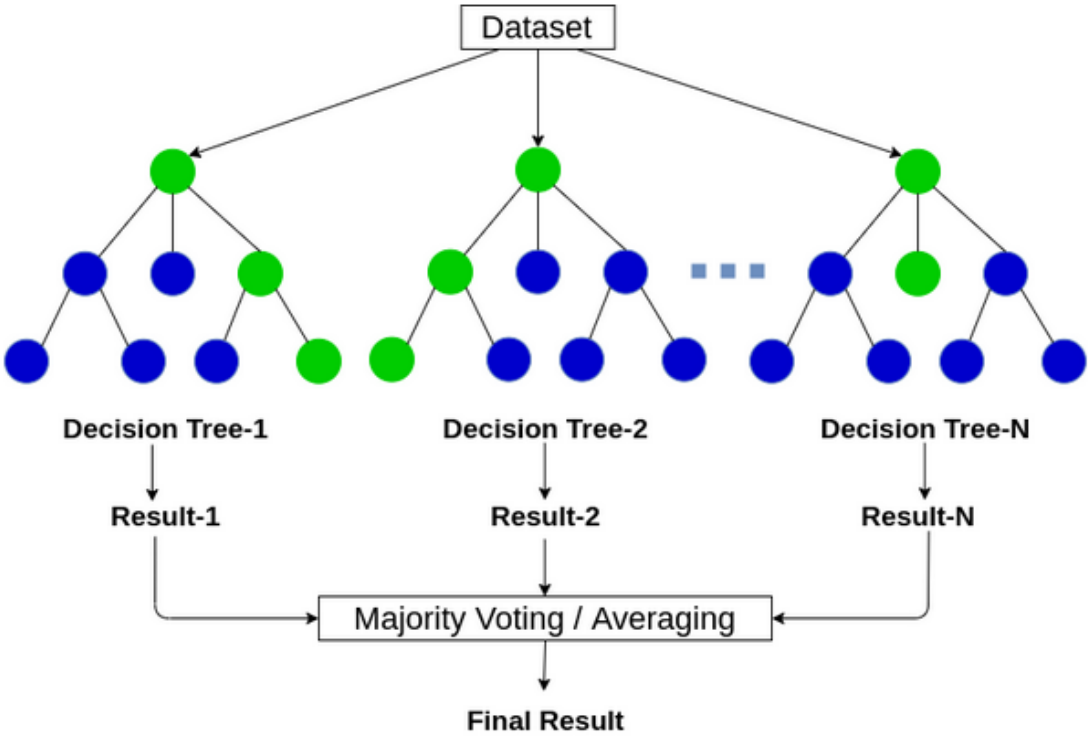

Figure 3.10: Random Forest Classifier example

[1]Figure taken from `https://www.analyticsvidhya.com`

XGBoost

XGBoost(eXtreme Gradient Boosting) started from a research project [56] is now an optimized distributed gradient boosting library [57], open-source and designed to be highly flexible, efficient and portable. It implements machine learning algorithms under the Gradient boosting framework.
Gradient boosting is a machine learning technique that gives a prediction model in the form of an ensemble of weak prediction models, typically decision trees.
A gradient-boosted trees model is built in a stage-wise fashion as in other different boosting methods, but it generalizes other methods by conceding the optimization of an arbitrary differentiable loss function.
XGBoost provides a parallel tree boosting system used to solve many different data science problems in a fast and accurate way. While the XGBoost model often achieves higher accuracy than a single decision tree, it sacrifices the intrinsic interpretability of decision trees.
Instead of training the best possible model on the data like in traditional methods, XGBoost trains a huge variety of models on different subsets of the training dataset and then selects the best performing one.
XGBoost Python implementation gives the possibility to access a vast number of inner parameters useful to get better precision and accuracy.
Some important features of XGBoost library are:

- **Parallelization:** its implementation allows a training with multiple CPU cores;
- **Regularization:** includes different regularization penalties to avoid overfitting;
- **Non-linearity:** can handle non-linear data patterns, learning from them;
- **Cross-validation:** already integrated in it;
- **Scalability:** it's available for many different programming languages and can be distributed between servers and clusters, allowing it to process a huge quantity of data.

Support Vector Machine

The support vector machine (SVM) is a classifier that represents the training data as points in space, split into categories through a gap that needs to be as wide as possible. The addition of new points inside the space is then made by predicting which category they fall into and which space they belong to.
Considering a set of training examples, each marked as belonging to one or more categories, an SVM algorithm builds a model able to assign new samples to one

category or another, developing a non-probabilistic binary linear classifier. SVM maps training examples to points in space in order to maximize the width of the gap between categories. New samples are then mapped into that same space and predicted to belong to a category based on their side of fall inside the space. Going into specific as visible in figure[2] 3.11, a support vector machine creates a hyperplane or set of hyperplanes in a high dimensional space, which can be used for classification, regression, outliers detection or or other different ML tasks.
Intuitively, the better separation is achieved by the hyperplane that has the largest distance to the nearest training samples of any class (called functional margin), since in general a larger margin is synonym of a lower generalization error of the classifier.
The use of a subset of training points in the decision function makes SVM memory efficient and highly effective in high dimensional spaces. From the disadvantage side, it does not directly provide probability estimates and it doesn't perform well with large datasets because the required training time is higher.

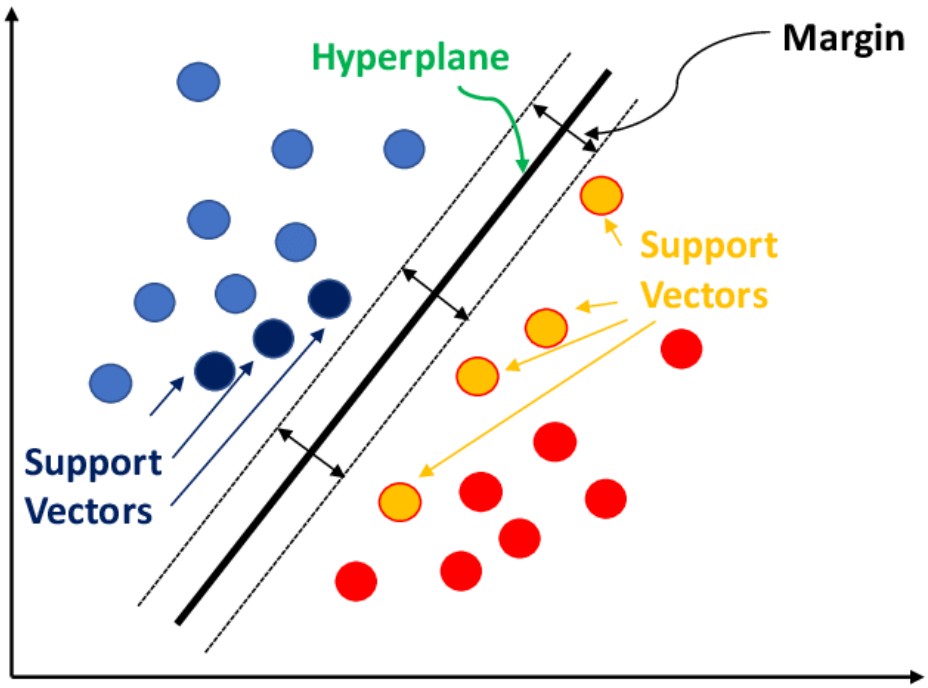

Figure 3.11: Support Vector Machine example

[2]Figure taken from [58]

Multi-layer perceptron

The multi-layer perceptron (MLP) is a feed-forward artificial neural network model that maps input data to a set of appropriate outputs. An MLP consists of at least three layers of nodes as in figure 3.12: an input layer, a hidden layer and an output layer, each of them fully connected to the following one.
Except for the input layer, each node is a neuron that uses a nonlinear activation function. Between input and output layers there is the possibility to insert one or more nonlinear hidden layers. Precisely these characteristics, so its multiple layers equipped with non-linear activation distinguish MLP from a linear perceptron.
For training purposes, MLP uses a supervised learning technique called backpropagation, a technique that computes the gradient in weight space of a feedforward neural network, with respect to a loss function. The main advantages of MLP are that it can be applied to complex non-linear problems and that it works very well with a huge quantity of input data. On the other hand, computations provided by an MLP model are difficult and time consuming.

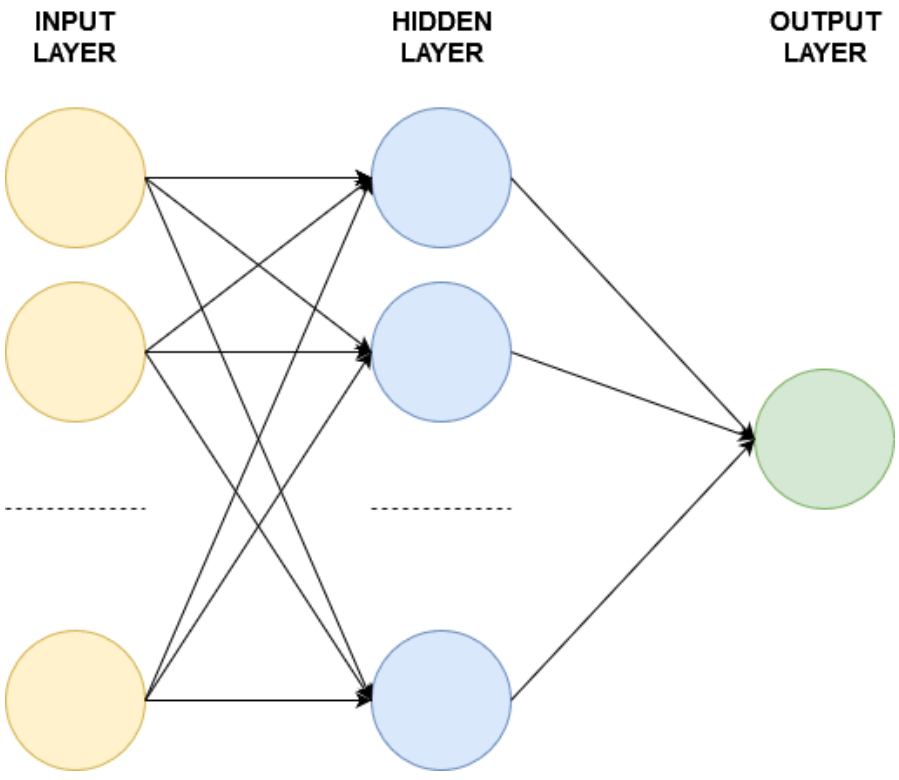

Figure 3.12: Multilayer perceptron example

Logistic Regression

Despite the name, Logistic regression is a classification algorithm in machine learning that exploits one or more independent variables to determine the final output. The outcome of logistic regression (figure [3] 3.13) is any binary value such as 1 or 0, Male or Female, Yes or No, Spam or Not Spam or in this specific case micro influencer or not micro influencer.
The goal of logistic regression is to find the best relationship between a dependent variable and a set of independent variables.
Logistic regression can be considered one of the most efficient techniques for solving classification problems since it's easier to implement, interpret, and very efficient to train. It is also very fast at classifying unknown samples and provides great performances in case of linearly separable dataset.
The main disadvantage of the logistic regression algorithm is its capacity of working only when the predicted variable is binary. In addition, it assumes that the data lacks missing values and that the predictors are independent between each other. Finally, as a last disadvantage, more powerful and compact algorithms such as Neural Networks can easily outperform its results.

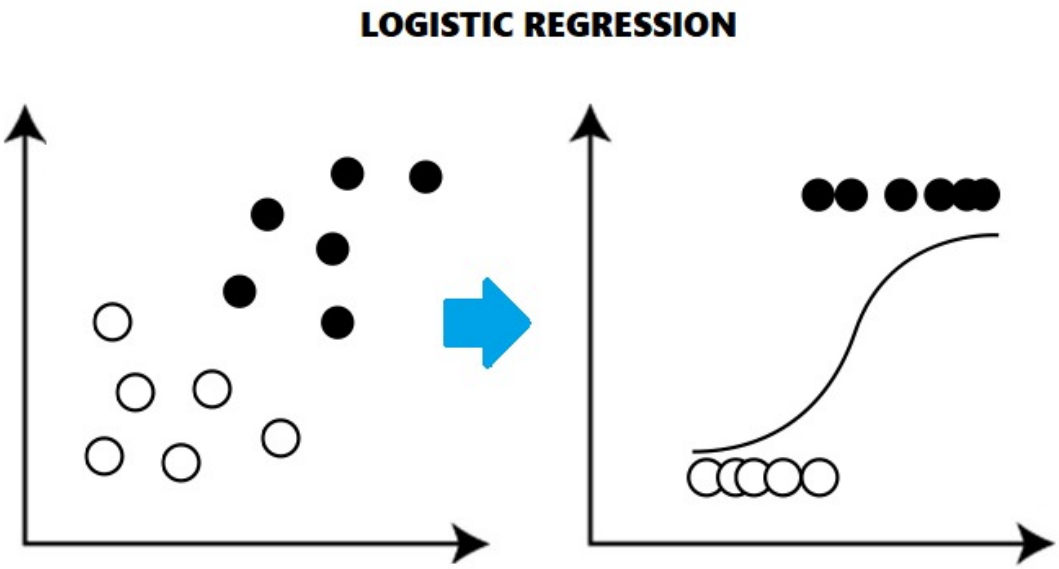

Figure 3.13: Logistic Regression example

[3]Figure taken from `https://www.equiskill.com/`

Stochastic Gradient Descent

Stochastic Gradient Descent (SGD) is a very effective and simple approach to fit linear models. It supports different loss functions (hinge, log) and penalties(l1, l2, elasticnet) for classification. SGD is simply an optimization technique, does not correspond to a specific family of machine learning models but can be considered only a way to train a model.
SGD is usually successfully applied to sparse and large-scale machine learning problems often available in text classification and natural language processing. Considering sparse data, the classifiers in this module easily scale to problems with more than 10^5 training examples and more than 10^5 features.
The main advantage is the ease of implementation and efficiency whereas the major cons with stochastic gradient descent are that it requires a high number of hyper-parameters and that it is sensitive to feature scaling.

3.5.2 Model Selection

Model selection is the process of selecting one final machine learning model from a collection of candidate machine learning models for a training dataset. This process can be applied both across models of the same type but configured with different hyperparameters and also across different types of models (e.g. Random forest, XGBoost, SVM, MLP, logistic regression, SGD). The first method makes use of the adoption of specific methods such as Grid Search while the latter is based on a comparison between different metrics.

Grid Search

Hyperparameters are the variables usually specified by the user during the building of a machine learning model. Every model has its own specific hyperparameters that must be carefully selected to obtain the best performances.
GridSearchCV is a model selection function available in the Scikit-Learn [59] library which uses the Grid Search technique for finding these optimal hyperparameters to properly increase model performances. Grid Search uses a different combination of all the specified hyperparameters and their values and calculates the performance for each combination, selecting as final result the best value for each combined hyperparameter. This makes the processing expensive and time-consuming based on the quantity of hyperparameters involved but allows to obtain the best combination to make the model work at its best.

Metrics

When the process of model selection regards the type of model involved, the comparison between different metrics becomes fundamental. Scikit-Learn [59] library proposes a specific function called **classification_report** that shows the results obtained by the model with the most useful metrics, as visible in the following output 3.14 produced on Google Colab [49].

	precision	recall	f1-score	support
not_micro_influencer	0.94	1.00	0.97	30
micro_influencer	1.00	0.93	0.97	30
accuracy			0.97	60
macro avg	0.97	0.97	0.97	60
weighted avg	0.97	0.97	0.97	60

Figure 3.14: Classification report example

Before going into specifics about metrics, a quick useful legend about the confusion matrix values on which are based:

- **True Positive (TP)**: if the actual classification is true and also the model prediction is true;
- **False Positive (FP)**: if the actual classification is false while the model prediction is true. This is known as a Type I error;
- **True Negative (TN)**: if the actual classification is false and also the model prediction is false;
- **False Negative (FN)**: if the actual classification is true while the model prediction is false. This is known as a Type II error;

The main metrics evaluated by the **classification_report** function are the following:

- **Accuracy:** is the ratio of correctly predicted observations to the total observations. The numerator is composed by the sum of TP and TN observations while the denominator by the total number of samples(TP+TN+FP+FN). It results to be accurate only if the model is balanced, inaccurate otherwise;
- **Precision:** is defined, for each class, as the ratio of TP to the sum of TP and FP. It can be seen as a measure of a classifier's exactness;

- **Recall:** is defined, for each class, as the ratio of TP to the sum of TP and FN. It can be seen as a measure of the classifier's completeness;
- **F1-score:** is a weighted average of precision and recall such that the best score is 1 while the worst is 0. It can be considered useful when dealing with unbalanced samples;
- **Support:** is the number of actual occurrences of the class in the specified dataset.

Chapter 4

Implementation

This chapter analyzes the main practical methods adopted to perform this thesis work presenting the principal code snippets implemented with their related explanations and ideas of use.
As reported in Chapter 3, two different pipelines have been implemented for Twitter and Instagram, having points in common but also small differences due to the information obtainable through their libraries and also to the different importance given to their contents. In fact, Twitter makes textual communication its strength while in the case of Instagram more prominence is given to the communication power of images.

4.1 Data Collection

The Data Collection phase provides two different interpretations due to the information obtainable by the two main libraries performing in this part, Tweepy [38] and Instaloader [40]. To follow this path, two different sections will be adopted to describe the metadata provided by these two libraries and, consequently, the metrics derived from them to provide a way of defining the micro influencer selection method. Both sections of the academic approach follow the same basic methodology: insertion of a list of topics, data scraping based on these topics, selection of 50% micro influencers and 50% not micro influencers between the users found in this research, tweets and posts download.

4.1.1 Twitter

To perform data collection on Twitter, the most suited library is Tweepy [38]. This Python library requests a developer account, obtainable on the Twitter Developer platform [39]. Once the registration as developer account is completed, the creation

of an app allows the generation of new keys and access tokens that are fundamental to obtain the platform access through the Python code.

User Object

The following table 4.1 presents the main metadata obtainable from each User object, as described in [39].
The function *tweepy.Cursor* allows to retrieve these information given as input:

- the **Search Tweet API** that searches against a subset of recent Tweets;
- the **topic** of interest;
- the **language** of tweets;
- the number of **items** to search.

ATTRIBUTE	TYPE	DESCRIPTION
id	int64	Unique identifier of the User.
id_str	String	String version of the unique identifier.
name	String	Name of the User, the one recheable through @name.
screen_name	String	Screen name of the User.
location	String	User-defined location (can be Null).
url	String	URL associated with the profile.
description	String	User-defined description (can be Null).
protected	Boolean	True if the User decided to protect its tweets, False otherwise.
verified	Boolean	True if the User has verified its account, False otherwise.
followers_count	Int	Number of accounts that follow this User.
friends_count	Int	Number of accounts this User is following.
listed_count	Int	Number of public lists that this User is a member of.
favourites_count	Int	Number of tweets that this User has liked in the account's lifetime.
statuses_count	Int	Number of tweets produced by the User including retweets).
created_at	String	The UTC datetime that the User account was created on Twitter.

Table 4.1: Twitter User Object

Tweet Object

The following table 4.2 presents the main metadata obtainable from each Tweet object, as described in [39].
The function *tweepy.Cursor* allows to retrieve these information given as input:

- the **User Timeline API** that selects the timeline of tweets of the user;
- the **user id**;
- the **tweet mode**, extended in this case to retrieve the full text of each tweet.

ATTRIBUTE	TYPE	DESCRIPTION
id	int64	Unique identifier of the Tweet.
id_str	String	String version of the unique identifier.
text	String	The actual UTF-8 text of the status update.
full_text	String	The actual UTF-8 full text of the status in extended mode (includes @RT for retweets).
user	User object	User that posted the Tweet.
retweet_count	Int	Number of times the Tweet has been retweeted.
favorite_count	Int	Number of times the Tweet has been liked.

Table 4.2: Tweet Object

Selection Metrics

As already explained in Chapter 1, this thesis follows the Micro Influencer definition proposed by Brewster and Lyu in 2020 [8].
To select micro influencers even more adequately with even more restrictive methods, specific metrics calculated ad hoc with the metadata obtainable from Tweepy are added to the number of followers proposed by the paper.
The presence of bots [60] is a problem that occurs on all social networks and for this reason the proposed metrics try to handle their presence with ad hoc metrics that try to evaluate the behavior and growth of the user throughout the life of its account. These metrics are illustrated in the following table 4.3 that explains both their obtaining and the filters that characterize them, if available.

ATTRIBUTE	DESCRIPTION	FILTER
age	User's age evaluated in days.	-
followers_count	Number of followers.	>5000 and <100000
statuses_count	Number of statuses.	>200
followers_growth_rate	Daily growth of followers. (followers_count / age)	>4
followers_following_ratio	Ratio between followers and following.	>2
tweet_freq	Daily number of tweets. (statuses_count / age)	>10
verified	True if the user is verified, False otherwise	= = False

Table 4.3: Twitter Micro Influencer selection metrics

The main reasons that lead to this specific filters selection are:

- **5000 < followers_count < 100000 :** to follow the definition provided in [8];
- **statuses_count > 200:** to select an active account having at least the number of tweets that will be evaluated in the framework;
- **followers_growth_rate > 4 :** to obtain a user able to constantly increase its followers since its account creation;
- **followers_following_ratio > 2 :** to underline an effective presence of real followers not obtained through "follow for follow" methods;
- **tweet_frequency > 10 :** to select an active user able to entertain its public everyday;
- **verified == False :** it is assumed that a micro influencer is not yet that famous to obtain verified status, that may be more typical for brands or celebrities.

For the purpose of a balanced dataset, for each topic the same number of not micro influencers is selected. To obtain valuable users, also for their metrics some filters are applied:

- **followers_count < 5000 or followers_count > 100000 :** to select users with both higher and lower audience;
- **statuses_count > 200:** to select an active account having at least the number of tweets that will be evaluated in the framework;
- **tweet_frequency > 10 :** to select an active user;

4.1.2 Instagram

To perform data collection on Instagram, the most suited library is Instaloader [40]. This open source Python library allows access to the Instagram API. This package, even if not official, offers great functionalities to scrape Instagram data. Requiring only an active Instagram account, its methods allow the user to download metadata from profiles, hashtags, stories, feeds and saved media.

Profile Object

The following table 4.4 presents the main metadata obtainable from each Profile object, Instagram Profile object can be accessed through *instaloader.Profile.from_username()* method that requires as input the Instaloader context and the username of the selected post.

ATTRIBUTE	TYPE	DESCRIPTION
userid	int64	Unique identifier of the Profile.
username	String	Name of the Profile.
full_name	String	Full name of the Profile.
biography	String	Profile-defined biography (can be Null).
is_private	Boolean	True if the Profile is private, False otherwise.
is_verified	Boolean	True if the Profile is verified, False otherwise.
followers	Int	Number of accounts that follow this Profile.
followees	Int	Number of accounts this Profile is following.
mediacount	Int	Number of posts produced by the Profile.

Table 4.4: Instagram Profile Object

Post Object

The following table 4.5 presents the main metadata obtainable from each Post object.
The function *instaloader.Hashtag.from_name().get_top_posts()* allows to retrieve these informations given as input the Instaloader context and the hashtag of interest.

ATTRIBUTE	TYPE	DESCRIPTION
owner_id	Int	Unique identifier of the Profile.
profile	String	Owner username
caption	String	Caption of the image.
typename	String	Type of post GraphImage, GraphVideo, GraphSidecar.
url	String	URL of the post.
likes	Int	Number of likes.
comments	Int	Number of comments.

Table 4.5: Instagram Post Object

Selection Metrics

As previously explained, this thesis follows the Micro Influencer definition proposed by Brewster and Lyu in 2020 [8].
To select micro influencers even more adequately with even more restrictive methods, specific metrics calculated ad hoc with the metadata obtainable from Instaloader are added to the number of followers proposed by the paper.
The presence of bots [60] is a problem that occurs on all social networks and for this reason the proposed metrics try to handle their presence with ad hoc metrics that try to evaluate the behavior and growth of the user throughout the life of its account. These metrics are illustrated in the following table 4.6 that explains both their obtaining and the filters that characterize them.

The main reasons that lead to this specific filters selection are:

- **5000 < followers < 100000 :** to follow the definition provided in [8];
- **mediacount > 200:** to select an active account;
- **followers_per_media > 2 :** to obtain a user able to constantly increase its followers with its posts;
- **followers_following_ratio > 2 :** to underline an effective presence of real followers not obtained through "follow for follow" methods.

ATTRIBUTE	DESCRIPTION	FILTER
followers	Number of followers.	>5000 and <100000
mediacount	Number of posts.	>200
followers_per_media	Ratio between followers and media.	>2
followers_following_ratio	Ratio between followers and following.	>2

Table 4.6: Instagram Micro Influencer selection metrics

For the purpose of a balanced dataset, for each topic the same number of not micro influencers is selected. To obtain valuable users, also for their metrics some filters are applied:

- **followers < 5000 or followers > 100000 :** to select users with both higher and lower audience;

- **mediacount > 200:** to select an active account.

4.2 Image Captioning

Image Captioning refers to the process of generating a textual description from a given image on the basis of the objects and actions present in it
This technique is taken into account to maximize the information obtainable from Instagram posts. In fact, as seen in the previous section, Instaloader for storage space reasons allows downloading the URL of the image not the image itself.
The image captioning part of this thesis is performed through the model presented by Mokady et al. [41] that guarantees a rather quick training to produce a competent captioning model. This part is performed on Google Colab [49] due to its high computational power required and the possibility of using GPU. The user can decide to produce image caption outputs with one of the two different datasets available, COCO [46] and Conceptual Captions [47]. This choice can affect the final result as visible in the two examples[1] 4.1 and 4.2 below.

[1]Images taken from `https://www.instagram.com/abeautifulplate/`

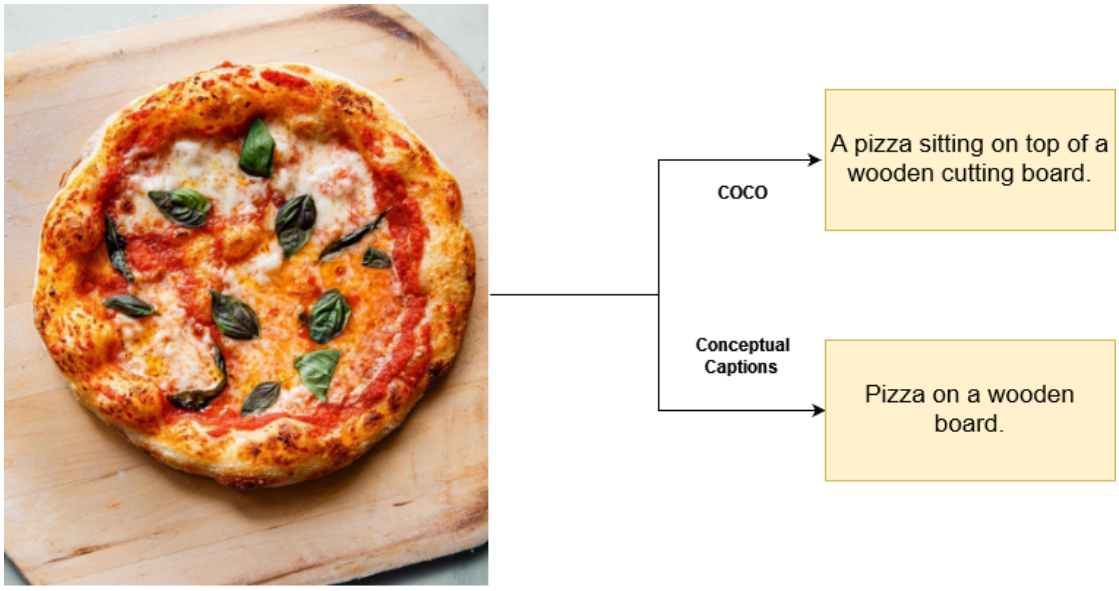

Figure 4.1: Image captioning example 1

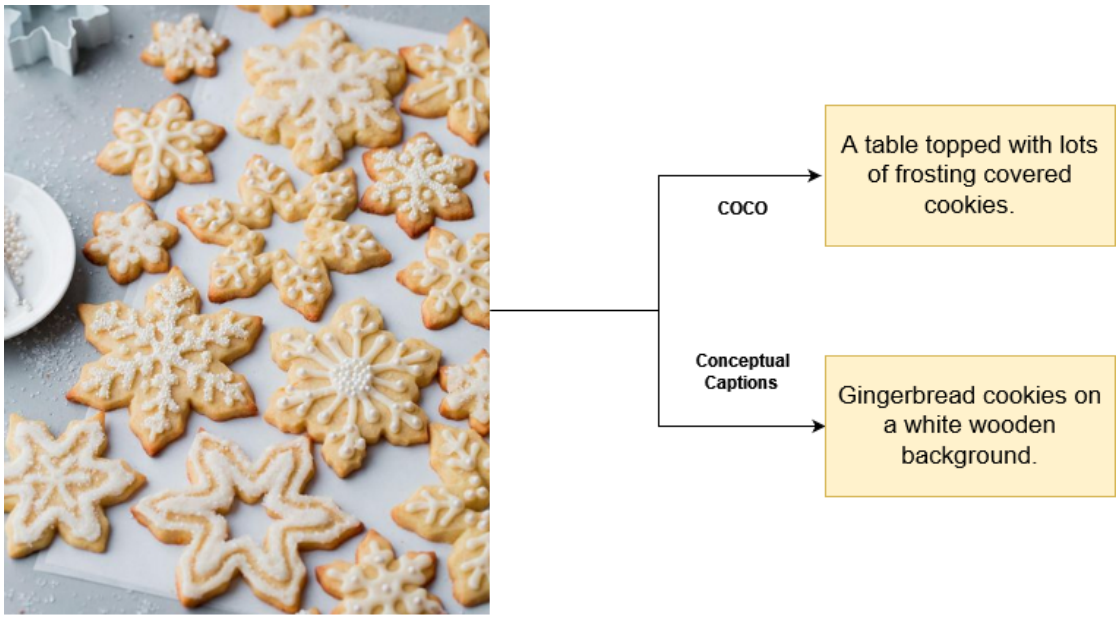

Figure 4.2: Image captioning example 2

4.3 Micro Topic Influencer selection

The goal of classification in machine learning is to generate a model able to assign the right class label to data.
To obtain this result, a balanced training set containing correctly labeled data is taken as input and used to teach the model how to classify in the proper way.
For this aim, in parallel with the micro influencer classification that uses the values obtained in data collection part, another parameter called "Micro Topic Influencer" is added to perform a more specific classification.
The individuation of this new type of parameter is explained in the following subsections.

4.3.1 Twitter

The selection of this new micro topic influencer figure for the Twitter part starts from the computation of two new metrics:

- **Topic % in Tweets:** measure of the effective presence of the topic inside all the tweets written by the user. It is computed as ratio between number of tweets containing the topic word and total number of tweets;

- **Topic % in Words:** measure of the effective presence of the topic inside all the words written by the user. It is computed as a ratio between total number of topic words and total number of words used inside tweets.

Once obtained these metrics a ranking strategy is adopted.
The selection described in table 4.7 is divided between two statistics: General and Topic, both having the same importance.
Considering only micro influencers selected in the data collection phase, for each of their metric are computed the P_{20},P_{40},P_{60},P_{80} distribution percentiles. Then, each user receives a score for each of these metrics with respect to their positioning inside the score ranges visible in the following table.
Once that every user has received its score, it is assigned the status of micro topic influencer only for those having a score greater than the general medium score.

GENERAL STATISTICS (50 %)					
CATEGORIES	POINTS				
	2	4	6	8	10
Followers	5k - P_{20}	P_{20} - P_{40}	P_{40} - P_{60}	P_{60} - P_{80}	P_{80} - 100k
Followers growth rate	4 - P_{20}	P_{20} - P_{40}	P_{40} - P_{60}	P_{60} - P_{80}	$> P_{80}$
Followers following ratio	2 - P_{20}	P_{20} - P_{40}	P_{40} - P_{60}	P_{60} - P_{80}	$> P_{80}$
Tweet frequency	10 - P_{20}	P_{20} - P_{40}	P_{40} - P_{60}	P_{60} - P_{80}	$> P_{80}$
Interactions	0 - P_{20}	P_{20} - P_{40}	P_{40} - P_{60}	P_{60} - P_{80}	$> P_{80}$
TOPIC STATISTICS (50 %)					
	5	10	15	20	25
Topic % in tweets	0 - P_{20}	P_{20} - P_{40}	P_{40} - P_{60}	P_{60} - P_{80}	$> P_{80}$
Topic % in words	0 - P_{20}	P_{20} - P_{40}	P_{40} - P_{60}	P_{60} - P_{80}	$> P_{80}$
MAX RANGE SCORE	20	40	60	80	100

Table 4.7: Twitter Micro Topic Influencer selection ranking

4.3.2 Instagram

The selection of this new micro topic influencer figure for the Instagram part starts from the computation of two new metrics:

- **Topic % in captions:** measure of the effective presence of the topic inside all the descriptions(called captions) written by the user. It is computed as a ratio between number of captions containing the topic word and total number of captions;
- **Topic % in cap words:** measure of the effective presence of the topic inside all the words written by the user. It is computed as ratio between total number of topic words and total number of words used inside captions.
- **Topic % in pics:** measure of the effective presence of the topic inside each pic-text obtained by the image captioning phase. It is computed as a ratio between number of pic-texts containing the topic word and total number of pic-texts;
- **Topic % in pic words:** measure of the effective presence of the topic inside each pic-text obtained by the image captioning phase. It is computed as ratio between total number of topic words and total number of words available inside pic-texts.

Once obtained these metrics a ranking strategy is adopted.
The selection described in table 4.8 is divided between two statistics: General and Topic, both having the same importance.
Considering only micro influencers selected in the data collection phase, for each of their metric are computed the P_{20},P_{40},P_{60},P_{80} distribution percentiles. In addition, considering the most difficult possibility to find topic words inside pic-texts, to obtain effective results for these parts are introduced also the P_{92},P_{94},P_{96},P_{98} percentiles. Then, each user receives a score for each of these metrics with respect to their positioning inside the score ranges visible in the following table.
Once that every user has received its score, it is assigned the status of micro topic influencer only for those having a score greater than the general medium score.

GENERAL STATISTICS (50 %)					
CATEGORIES	POINTS				
	2.5	5	7.5	10	12.5
Followers	5k - P_{20}	P_{20} - P_{40}	P_{40} - P_{60}	P_{60} - P_{80}	P_{80} - 100k
Followers per media	2 - P_{20}	P_{20} - P_{40}	P_{40} - P_{60}	P_{60} - P_{80}	$> P_{80}$
Followers following ratio	2 - P_{20}	P_{20} - P_{40}	P_{40} - P_{60}	P_{60} - P_{80}	$> P_{80}$
Interactions	0 - P_{20}	P_{20} - P_{40}	P_{40} - P_{60}	P_{60} - P_{80}	$> P_{80}$
TOPIC STATISTICS (50 %)					
	2.5	5	7.5	10	12.5
Topic % in captions	0 - P_{20}	P_{20} - P_{40}	P_{40} - P_{60}	P_{60} - P_{80}	$> P_{80}$
Topic % in cap words	0 - P_{20}	P_{20} - P_{40}	P_{40} - P_{60}	P_{60} - P_{80}	$> P_{80}$
Topic % in pics	0 - P_{92}	P_{92} - P_{94}	P_{94} - P_{96}	P_{96} - P_{98}	$> P_{98}$
Topic % in pic words	0 - P_{92}	P_{92} - P_{94}	P_{94} - P_{96}	P_{96} - P_{98}	$> P_{98}$
MAX RANGE SCORE	20	40	60	80	100

Table 4.8: Instagram Micro Topic Influencer selection ranking

4.4 Text Preprocessing

The main text preprocessing steps presented in Chapter 3.3 are provided for both Twitter and Instagram through the following function:

```
try:
    language = detect(tweet)
except:
    language = "Not available"

if(stopwords.has_lang(language)):
    stop = stopwords.stopwords(language)
else:
    stop = stopwords.stopwords("en")

tweet = demojize(tweet) #convert emojis to text
temp = tweet.lower()
temp = re.sub("'", "", temp) # to avoid removing contractions
temp = re.sub("@[A-Za-z0-9_]+","", temp) #remove entire
#tag (@ and also name tagged)
temp = re.sub("#","", temp) #remove only hashtag symbol
temp = re.sub(r'http\S+', '', temp)  #remove links
temp = re.sub('[()!?.,;:]', ' ', temp) #remove ()!?.,:;
temp = re.sub('\[.*\].*',' ', temp) #remove []
temp = re.sub("[^a-z]"," ", temp) #remove numbers
temp = temp.split()
temp = [w for w in temp if not w in stop] #remove stopwords
temp = [WordNetLemmatizer().lemmatize(w, pos='v') for w in temp]
temp = " ".join(word for word in temp)
return temp
```

The function detects the language of the text, if available, and selects its proper stopwords. Then the entire set of words get demojized. Once obtained the full text, it is set to its lower aspect, cleaned from numbers, symbols and links. Successively, the split method traduces the text into a list of words allowing the execution of the last steps: for each word is checked its presence inside stopwords list and, in case of correspondence, the word is removed; then, each word gets lemmatized and finally reunited with the others to rebuild the starting phrase, but cleaned.

4.5 Sentiment Analysis

Communication skills are different from user to user, there is not a basic pattern. In this optic, understanding how each user is able to communicate with its audience can help to individuate its language characteristics. On the other hand, different topics may need different communication patterns: one topic may be more suitable for positive messages, another for neutral and so on. Sentiment analysis performed for each user becomes fundamental to understand its communication skills.
The code described below specifies the techniques adopted for this purpose.

```
tokenizer = AutoTokenizer.from_pretrained
    ("cardiffnlp/twitter-roberta-base-sentiment")

model = AutoModelForSequenceClassification.from_pretrained
    ("cardiffnlp/twitter-roberta-base-sentiment")

model.save_pretrained("cardiffnlp/twitter-roberta-base-sentiment")

labels = ['negative', 'neutral', 'positive']
#for every tweet
t = clean_tweet(tweet) #text cleaning
encoded_input = tokenizer(t, return_tensors='pt')
output = model(**encoded_input)
scores = output[0][0].detach().numpy()
scores = softmax(scores)

ranking = np.argsort(scores)
ranking = ranking[::-1]
for i in range(scores.shape[0]):
l = labels[ranking[i]]
s = scores[ranking[i]]
if(l=='positive'):
    pos.append(s) #list to store positive sentiment
elif(l=='neutral'):
    neu.append(s) #list to store neutral sentiment
else:
    neg.append(s) #list to store negative sentiment

#mean of all user tweets sentiments
positiveSentiment.append(np.mean(pos))
neutralSentiment.append(np.mean(neu))
negativeSentiment.append(np.mean(neg))
```

The code starts with the downloads of the model and tokenizer presented in [53]. Then, for every user the sentiment analysis is performed as following:

- every tweet (Twitter case) or caption/pic-text (Instagram case) is retrieved;
- the text gets cleaned;
- cleaned text is tokenized and then passed to the model in the form of a Pytorch [61] tensor;
- the model outputs three scores: positive, negative, neutral;
- each score is associated with its proper label and inserted in the corresponding list;
- for every user the mean value of each sentiment is stored.

This analysis is performed for every tweet in Twitter case resulting in three new parameters (one per sentiment) while in the Instagram case is performed twice for both captions and pic-texts resulting in six new parameters.

4.6 Classification

Classification in machine learning has as its main goal the generation of a model able to assign the right class label to the right data. To perform this task data gets divided in two subsets, training set(80% of the entire dataset in this case) and test set(20%). Training set contains correctly labeled data and is used to teach the model how to classify, the test set instead is used to understand how the model has learnt, thanks to its unlabeled data.
As already seen in the previous sections, datasets of Twitter and Instagram contain some different metrics. For this purpose, two different sets of parameters are taken as input for the classification model, as visible in the following code snippet.

```
#Twitter
X = df.loc[:, ["followers", "age", "followers_growth_rate",
"followers_following_ratio","tweet_freq",
"interactions_no_retweets","topicInTweetsPercentage",
"topicInWordsPercentage", "positiveSentiment",
"neutralSentiment" ,"negativeSentiment"]]
```

```
#Instagram
X = df.loc[:, ["followers", "followers_per_media",
"followers_following_ratio","interactions",
ù"topicInCaptionsPercentage", "topicInWordsPercentage",
"topicInPicsPercentage","topicInPicsWordsPercentage",
"positiveSentimentCaptions","neutralSentimentCaptions",
"negativeSentimentCaptions","positiveSentimentPics",
"neutralSentimentPics" ,"negativeSentimentPics"]]
```

For all the models presented in chapter 3 the framework works as follows:

- adoption of GridSearchCV model selection function available in the Scikit-Learn [59] library which uses the Grid Search technique for finding the optimal hyperparameters to properly increase model performances;
- selection of the best hyperparameters for both micro influencer and micro topic influencer classification;
- execution of all the models with their best hyperparameters;
- comparison between all the classification report results;
- selection and export of the best performing model for both micro influencer and micro topic influencer classification.

The results obtained and the finality of the models export are described in the following chapters.

Chapter 5

Results

The following section presents all the results obtained in the model selection part, for both Twitter and Instagram datasets.
These results concern users and data belonging to a specific research carried out in the academic approach based on 30 topics. For this reason and for the nature of the work that allows the study of the various classification models, these results should not be considered as general but for the specific dataset. Consequently, new topics and new moments in which these searches are performed could lead to the selection of more accurate models other than those exposed in this section.
For better readability, some abbreviations have been introduced in the following tables and their translation can be traced back to the following legend 5.1.

ABBREVIATION	FULL NAME
RFC	RANDOM FOREST CLASSIFIER
XGBOOST	EXTREME GRADIENT BOOSTING
SVM	SUPPORT VECTOR MACHINE
MLP	MULTI-LAYER PERCEPTRON
LOG REG	LOGISTIC REGRESSION
SGD	STOCHASTIC GRADIENT DESCENT

Table 5.1: Results legend

5.1 Twitter

Twitter classification case is based on a specific dataset composed of 30 heterogeneous topics, 300 different users and their 60000 tweets (200 for each user).
Two main classifications are applied: one adopts as class label the "Micro influencer" parameter assigned during data collection phase presented in Chapter 4.1.1; the other adopts as class label the "Micro topic influencer" parameter assigned during its specific selection phase presented in Chapter 4.3.1.
From what will be deduced in the following tables, the classification of micro influencers is able to obtain better results in the great majority of the models used. This can be a repeatable result even with different datasets as the "micro influencer" parameter is populated in a balanced way in the first step of the computations while the "micro topic influencer" parameter varies from dataset to dataset and depends on the average score of each of them, going to produce with great probability a subdivision that is not entirely equally balanced.
Both results presented in the following tables 5.2 and 5.3 indicate eXtreme Gradient Boosting as better model while Stochastic Gradient Descent as worst.
XGBoost, in fact, trains a huge variety of models on different subsets of the training dataset and then selects the best performing one. On the other hand, SGD classifier requires a higher number of hyper-parameters and due to this lack is sensitive to feature scaling.
Despite its simplicity, random forest classifier guarantees great results in both cases, proposing itself as the second best model able to approach the more complex XGBoost using much shorter execution times.
While Multi-Layer Perceptron demonstrates being able to work better with a balanced dataset, this cannot be said for Support Vector Machine and Logistic Regression which show an opposite behavior demonstrating to be more performing in high dimensional spaces.

5.1.1 Micro Influencer

MODEL	METRICS			
	ACCURACY	**PRECISION**	**RECALL**	**F1-SCORE**
RFC	0.98	0.98	0.98	0.98
XGBOOST	**1.00**	**1.00**	**1.00**	**1.00**
SVM	0.73	0.74	0.73	0.73
MLP	0.90	0.90	0.90	0.90
LOG REG	0.77	0.77	0.77	0.77
SGD	0.52	0.31	0.52	0.39

Table 5.2: Results of Twitter micro influencers classification

5.1.2 Micro Topic Influencer

MODEL	METRICS			
	ACCURACY	**PRECISION**	**RECALL**	**F1-SCORE**
RFC	0.92	0.92	0.92	0.92
XGBOOST	**0.93**	**0.93**	**0.93**	**0.93**
SVM	0.88	0.90	0.88	0.89
MLP	0.63	0.62	0.63	0.62
LOG REG	0.88	0.90	0.88	0.89
SGD	0.65	0.69	0.65	0.66

Table 5.3: Results of Twitter micro topic influencers classification

5.2 Instagram

Instagram classification case is based on a specific dataset composed of 30 heterogeneous topics, 300 different users and their 15000 textual productions (25 posts descriptions and 25 image captions for each user).
Four main classifications are applied: two adopt as class label the "Micro influencer" parameter assigned during data collection phase presented in Chapter 4.1.2; the others adopt as class label the "Micro topic influencer" parameter assigned during its specific selection phase presented in Chapter 4.3.2.
These two additional classifications are evaluated through the adoption of two different datasets during the image captioning phase: COCO and Conceptual Captions.
From what will be deduced in the following tables, the classification of micro influencers is able to obtain better results in the great majority of the models used. This can be a repeatable result even with different datasets as the "micro influencer" parameter is populated in a balanced way in the first step of the computations while the "micro topic influencer" parameter varies from dataset to dataset and depends on the average score of each of them, resulting in the production with great probability of a subdivision that is not entirely equally balanced.
Also in the Instagram case, all results presented in the following tables, respectively for COCO dataset 5.4 - 5.6 and for Conceptual Captions dataset 5.5 - 5.7 indicate eXtreme Gradient Boosting as better model while Stochastic Gradient Descent as worst. Both cases underline rather similar results between the two datasets, with a slight prevalence of COCO one.
XGBoost, as previously explained, is able to train a huge variety of models on different subsets of the training dataset that leads to the selection of the best performing one. On the other hand, SGD classifier requires a higher number of hyper-parameters and due to this lack is sensitive to feature scaling.
Despite its simplicity, random forest classifier guarantees also in this case great results in both datasets and classifications, proposing itself as the second best model able to approach the more complex XGBoost using much shorter execution times.
While Multi-Layer Perceptron demonstrates being able to work better with a balanced dataset, this cannot be said for Support Vector Machine and Logistic Regression which demonstrate an opposite behavior resulting in more performance in high dimensional spaces.

5.2.1 Micro Influencer

COCO dataset

MODEL	METRICS			
	ACCURACY	**PRECISION**	**RECALL**	**F1-SCORE**
RFC	0.97	0.97	0.97	0.97
XGBOOST	**0.98**	**0.98**	**0.98**	**0.98**
SVM	0.60	0.60	0.60	0.60
MLP	0.65	0.65	0.65	0.65
LOG REG	0.60	0.60	0.60	0.60
SGD	0.55	0.55	0.55	0.55

Table 5.4: Results of Instagram micro influencers classification with COCO dataset

Conceptual Captions dataset

MODEL	METRICS			
	ACCURACY	**PRECISION**	**RECALL**	**F1-SCORE**
RFC	0.97	0.97	0.97	0.97
XGBOOST	**0.98**	**0.98**	**0.98**	**0.98**
SVM	0.62	0.62	0.62	0.61
MLP	0.63	0.63	0.63	0.63
LOG REG	0.58	0.58	0.58	0.58
SGD	0.50	0.50	0.50	0.50

Table 5.5: Results of Instagram micro influencers classification with Conceptual Captions dataset

5.2.2 Micro Topic Influencer

COCO dataset

MODEL	METRICS			
	ACCURACY	**PRECISION**	**RECALL**	**F1-SCORE**
RFC	0.90	0.90	0.90	0.90
XGBOOST	**0.91**	**0.91**	**0.91**	**0.91**
SVM	0.85	0.85	0.85	0.85
MLP	0.57	0.56	0.57	0.56
LOG REG	0.87	0.87	0.87	0.87
SGD	0.55	0.54	0.55	0.54

Table 5.6: Results of Instagram micro topic influencers classification with COCO dataset

Conceptual Captions dataset

MODEL	METRICS			
	ACCURACY	**PRECISION**	**RECALL**	**F1-SCORE**
RFC	0.88	0.89	0.88	0.88
XGBOOST	**0.89**	**0.89**	**0.89**	**0.89**
SVM	0.88	0.88	0.88	0.88
MLP	0.67	0.68	0.67	0.66
LOG REG	0.87	0.87	0.87	0.87
SGD	0.50	0.53	0.51	0.50

Table 5.7: Results of Instagram micro topic influencers classification with Conceptual Captions dataset

Chapter 6

Discussion

This chapter focuses on the negative aspects of this job, namely on the main difficulties met during the creation of the frameworks presented in this thesis. Despite their great importance in the realization of this project, Tweepy and Instaloader libraries caused several problems that required different methods to address, solve or handle them.

Time and usage are the most important aspects that created problems, especially in the first part, the data collection, to obtain users and their information, for both Twitter and Instagram.
Tweepy and Instaloader libraries, in fact, concede the use of a limited amount of data in a determined window of time, ending up producing a considerable quantity of timeouts that result in a very time consuming effort.
Another problem that comes directly from the Instagram section is the temporary availability of the picture's URLs. In fact, each URL has a limited life that doesn't overcome the week, even less. This problem, in addition with the limited possibilities of daily downloads, caused a lower number of post downloaded (25 per each user) with respect to the tweets (200 per each user).
To coexist with these problems, the data collection was performed step by step, day by day, creating many daily subsets that have been merged as a unique and bigger dataset. Considering this approach, it becomes obvious that an extended amount of time with respect to the thesis one, may lead to a bigger, more complex and more detailed dataset producing even more accurate results.

Another topic of discussion may concern the tools adopted to perform these frameworks. The work, in fact, had to be split between a simple Python editor as Visual Studio Code [62] and Google Colab [49].
A Python interpreter from remote is fundamental for the data collection part because the geo-localization of Colab creates a continuous production of warnings

within social networks because the latter identify a completely different position from the original and continuously intervene asking for permissions and going so far as to block the account.
On the other side, the lack of large means available capable of performing very expensive computations based on the use of GPU results in the use of Google Colab that becomes fundamental to perform some specific tasks as image captioning and model selection. This problem affects both academic approaches and also the economic one of Instagram related to the image captioning part. For this reason, only the economic framework of Twitter allows the brand to perform the micro influencer classification in one single step, differently from the three required in the Instagram case.

Chapter 7

Economic approach

The particularity of this thesis is the possibility to adopt the same methods provided by the academic approach to build a framework useful even for an economic purpose. Starting from the best models previously described in Chapter 5, this framework is able to guarantee an economic exploitation usable by brands that are interested in the research of micro influencers related to their specific topic.

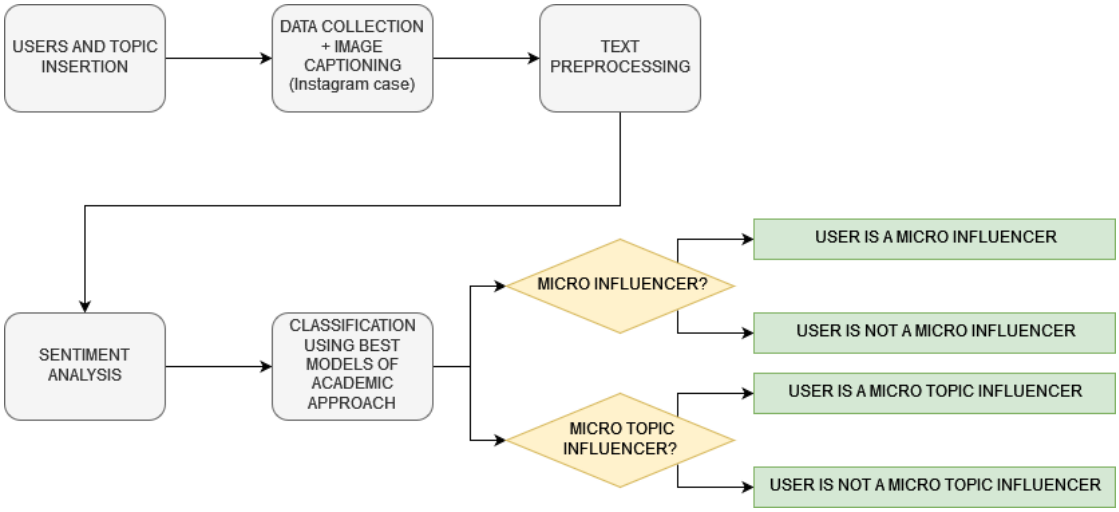

Figure 7.1: Economic approach pipeline

The framework presented in figure 7.1 works as follows:

- the brand is asked to insert the names of the possible micro influencers for whom shows interest and its specific topic;
- once the model receives these informations as input, retrieves the account's informations, previously described in Chapter 4.1, and creates a input-users-centered dataset;
- for the Instagram case a proper functionality is inserted to produce ah hoc captions from the user's pictures, as presented in Chapter 4.2;
- with the proper user's text productions obtained in the previous steps, the data cleaning process is applied as specified in Chapter 4.4, accompanied by a topic-oriented research made inside both text and words to evaluate its usage's percentage;
- finally, a sentiment analysis as described in Chapter 4.5 is performed to obtain three distinct values regarding positive, neutral and negative scores (six for the Instagram case that includes also images computation);
- with all these user's parameters available, the previously obtained models are fed with them and used to establish as final output if the inserted users may be considered by the brand as general micro influencers but, most importantly, as specific micro influencers for the topic inserted in the first step.

Chapter 8

Conclusion

8.1 Recap

This thesis presented a new framework for Twitter and Instagram, which can be adopted for both academic and economic purposes, capable of addressing a current issue in both the scientific and professional fields: the research and classification of a new figure established in recent years on social networks, the micro influencer.
The thesis has its own fundamentals on some essential theory aspects mentioned in the appropriate sections that has been taken as starting point for the dissertation but also as inspiration for the creation of ad hoc metrics for the topic in question.
Social media analysis and influencer detection have become topics of study and research that have been considerably addressed in recent years and consequently have influenced both the data collection part and the metrics evaluation one. The data obtained have been then further enriched with image captioning techniques and subsequent sentiment analysis which made it possible to trace the communication techniques that characterize the figure of the micro influencer and, more specifically, of the micro topic influencer. In fact, this work aims to investigate the correlation between micro influencers and the subject matter, trying to help brands in the search for possible advertising ambassadors for their specific field.
The main purpose of the academic approach was precisely to find the best classification model for both figures considered, both the classic micro influencer and the more specific micro topic influencer. These resulted in the selection of a XGBoost model for both cases and social networks. This choice however, must be read from the point of view of a specific dataset that can be reconstructed in different ways and with different topics.
Precisely in this perspective, the academic approach proposes a framework for choosing among the various models in order to obtain the most performing every time. So, with this final model elected of all as most accurate the computation

moves on the economic side.
A practical application proposed in this work is, in fact, an economic framework that exploits the model obtained to help brands in the search for micro influencers: the insertion as input of usernames and topic is sufficient for it to complete all the necessary computations and emit a final classification as an output on the possibility that the inserted users are micro influencers in general or even micro influencers for the inserted topic.

8.2 Future works

This thesis work could be extended and improved in several directions.
New social networks. This project is focused on Twitter and Instagram for their libraries availability and ease of use. Despite this, many other social networks may be explored like Facebook (already has a library but very limited), TikTok, Reddit, etc.
Linkedin application. This work could be an interesting starting point to produce a similar framework for brands and companies not for selecting influencers but for directly selecting staff suited to their needs.
More data, time and metrics. This framework can be a starting point for a wider research, not limited in time as a thesis, to produce a bigger dataset, introduce further metrics and compare more classification models.
Libraries evolution. As previously discussed, this work has been narrowed by libraries' timeouts and rate limits. An evolution of libraries limitation in a more permissive way may facilitate future works in this research topic.
Videos and stories. This work has limited its action field mainly on text for Twitter and on text and images for Instagram. An evolution to better understand users' communication skills may be based on the visual sector study, performing video translations to text for both Instagram Reels and Instagram stories.
Related words addition. Some of the metrics adopted in this thesis compute the presence of the topic word inside the entire obtained text. A further step may require the addition of more topic-specific words to produce these metrics and better understand the affinity of a user with the specific topic.

Bibliography

[1] Statista. *Number of social network users in the USA*. URL: https://www.statista.com/statistics/278409/number-of-social-network-users-in-the-united-states/ (cit. on p. 2).

[2] Jayson DeMers. *The Top 10 Benefits Of Social Media Marketing*. 2014. URL: http://onforb.es/1vyccu4 (cit. on pp. 2, 3).

[3] Maria Teresa Pinheiro Melo Borges Tiago and Jose Manuel Cristovao Verissimo. «Digital marketing and social media: Why bother?» In: *Business horizons* 57.6 (2014), pp. 703–708 (cit. on p. 3).

[4] Fernando van der Vlist and Anne Helmond. «Social media in the audience economy: Business-to-business partnerships and co-dependence». In: *AoIR Selected Papers of Internet Research* (2021) (cit. on p. 3).

[5] Cambridge dictionary. *Influencer definition*. URL: https://dictionary.cambridge.org/it/dizionario/inglese/influencer (cit. on p. 4).

[6] Alexander P Schouten, Loes Janssen, and Maegan Verspaget. «Celebrity vs. Influencer endorsements in advertising: the role of identification, credibility, and Product-Endorser fit». In: *International journal of advertising* 39.2 (2020), pp. 258–281 (cit. on p. 4).

[7] Anjali Chopra, Vrushali Avhad, Jaju, and Sonali. «Influencer marketing: An exploratory study to identify antecedents of consumer behavior of millennial». In: *Business Perspectives and Research* 9.1 (2021), pp. 77–91 (cit. on p. 4).

[8] Maureen Lehto Brewster and Jewon Lyu. «Exploring the parasocial impact of nano, micro and macro influencers». In: *International Textile and Apparel Association Annual Conference Proceedings*. Vol. 77. 1. Iowa State University Digital Press. 2020 (cit. on pp. 6, 46, 47, 49).

[9] Pei-Shan Wei and Hsi-Peng Lu. «An examination of the celebrity endorsements and online customer reviews influence female consumers' shopping behavior». In: *Computers in Human Behavior* 29.1 (2013), pp. 193–201 (cit. on p. 7).

[10] Deborah Weinswig. *Influencers Are The New Brands*. 2016. URL: https://www.forbes.com/sites/deborahweinswig/2016/10/05/influencers-are-the-new-brands/?sh=7d3ed6627919 (cit. on p. 7).

[11] Carmen Berne-Manero and Mercedes Marzo-Navarro. «Exploring how influencer and relationship marketing serve corporate sustainability». In: *Sustainability* 12.11 (2020), p. 4392 (cit. on p. 7).

[12] Kelly Ehlers. *Micro-Influencers: When Smaller Is Better*. 2021. URL: https://www.forbes.com/sites/forbesagencycouncil/2021/06/02/micro-influencers-when-smaller-is-better/?sh=17525270539b (cit. on p. 9).

[13] Fabián Riquelme and Pablo González-Cantergiani. «Measuring user influence on Twitter: A survey». In: *Information processing & management* 52.5 (2016), pp. 949–975 (cit. on p. 15).

[14] Stefan Räbiger and Myra Spiliopoulou. «A framework for validating the merit of properties that predict the influence of a twitter user». In: *Expert Systems with Applications* 42.5 (2015), pp. 2824–2834 (cit. on p. 15).

[15] Linyuan Lü, Duanbing Chen, Xiao-Long Ren, Qian-Ming Zhang, Yi-Cheng Zhang, and Tao Zhou. «Vital nodes identification in complex networks». In: *Physics Reports* 650 (2016), pp. 1–63 (cit. on p. 16).

[16] Siwar Jendoubi, Arnaud Martin, Ludovic Liétard, Hend Ben Hadji, and Boutheina Ben Yaghlane. «Two evidential data based models for influence maximization in twitter». In: *Knowledge-Based Systems* 121 (2017), pp. 58–70 (cit. on p. 16).

[17] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. «What is Twitter, a social network or a news media?» In: *Proceedings of the 19th international conference on World wide web*. 2010, pp. 591–600 (cit. on p. 16).

[18] Xin Chen. «Critical nodes identification in complex systems». In: *Complex & Intelligent Systems* 1.1 (2015), pp. 37–56 (cit. on p. 16).

[19] Changjun Fan, Li Zeng, Yizhou Sun, and Yang-Yu Liu. «Finding key players in complex networks through deep reinforcement learning». In: *Nature machine intelligence* 2.6 (2020), pp. 317–324 (cit. on p. 17).

[20] Iris Roelens, Philippe Baecke, and Dries F Benoit. «Identifying influencers in a social network: The value of real referral data». In: *Decision Support Systems* 91 (2016), pp. 25–36 (cit. on p. 17).

[21] Tian Gan, Shaokun Wang, Meng Liu, Xuemeng Song, Yiyang Yao, and Liqiang Nie. «Seeking micro-influencers for brand promotion». In: *Proceedings of the 27th ACM International Conference on Multimedia*. 2019, pp. 1933–1941 (cit. on p. 17).

[22] Jin-Hwa Kim, Kyoung-Woon On, Woosang Lim, Jeonghee Kim, Jung-Woo Ha, and Byoung-Tak Zhang. «Hadamard product for low-rank bilinear pooling». In: *arXiv preprint arXiv:1610.04325* (2016) (cit. on p. 17).

[23] Benyamin Bashari and Ehsan Fazl-Ersi. «Influential post identification on Instagram through caption and hashtag analysis». In: *Measurement and Control* 53.3-4 (2020), pp. 409–415 (cit. on p. 18).

[24] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. «Efficient estimation of word representations in vector space». In: *arXiv preprint arXiv:1301.3781* (2013) (cit. on p. 18).

[25] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. «Enriching word vectors with subword information». In: *Transactions of the association for computational linguistics* 5 (2017), pp. 135–146 (cit. on p. 18).

[26] Cheng Zheng, Qin Zhang, Sean Young, and Wei Wang. «On-demand Influencer Discovery on Social Media». In: *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2020, pp. 2337–2340 (cit. on p. 19).

[27] Yun-Bei Zhuang, Zhi-Hong Li, and Yun-Jing Zhuang. «Identification of influencers in online social networks: measuring influence considering multidimensional factors exploration». In: *Heliyon* 7.4 (2021), e06472 (cit. on p. 20).

[28] Ghazaleh Beigi, Xia Hu, Ross Maciejewski, and Huan Liu. «An overview of sentiment analysis in social media and its applications in disaster relief». In: *Sentiment analysis and ontology engineering* (2016), pp. 313–340 (cit. on p. 22).

[29] Lin Yue, Weitong Chen, Xue Li, Wanli Zuo, and Minghao Yin. «A survey of sentiment analysis in social media». In: *Knowledge and Information Systems* 60.2 (2019), pp. 617–663 (cit. on p. 23).

[30] Bo Pang, Lillian Lee, et al. «Opinion mining and sentiment analysis». In: *Foundations and Trends® in information retrieval* 2.1–2 (2008), pp. 1–135 (cit. on p. 23).

[31] Bing Liu. «Sentiment analysis and opinion mining». In: *Synthesis lectures on human language technologies* 5.1 (2012), pp. 1–167 (cit. on p. 23).

[32] Chenhao Tan, Lillian Lee, Jie Tang, Long Jiang, Ming Zhou, and Ping Li. «User-level sentiment analysis incorporating social networks». In: *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2011, pp. 1397–1405 (cit. on p. 23).

[33] Erik Cambria, Andrew Livingstone, and Amir Hussain. «The hourglass of emotions». In: *Cognitive behavioural systems*. Springer, 2012, pp. 144–157 (cit. on p. 24).

[34] Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. «Lexicon-based methods for sentiment analysis». In: *Computational linguistics* 37.2 (2011), pp. 267–307 (cit. on p. 24).

[35] Walaa Medhat, Ahmed Hassan, and Hoda Korashy. «Sentiment analysis algorithms and applications: A survey». In: *Ain Shams engineering journal* 5.4 (2014), pp. 1093–1113 (cit. on p. 24).

[36] Lei Zhang, Shuai Wang, and Bing Liu. «Deep learning for sentiment analysis: A survey». In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 8.4 (2018), e1253 (cit. on p. 24).

[37] Ajeet Ram Pathak, Manjusha Pandey, and Siddharth Rautaray. «Topic-level sentiment analysis of social media data using deep learning». In: *Applied Soft Computing* 108 (2021), p. 107440 (cit. on p. 24).

[38] *Tweepy*. URL: `https://www.tweepy.org/` (cit. on pp. 27, 44).

[39] *Twitter Development Platform*. URL: `https://developer.twitter.com/` (cit. on pp. 27, 44–46).

[40] *Instaloader*. URL: `https://instaloader.github.io/` (cit. on pp. 28, 44, 48).

[41] Ron Mokady, Amir Hertz, and Amit H Bermano. «Clipcap: Clip prefix for image captioning». In: *arXiv preprint arXiv:2111.09734* (2021) (cit. on pp. 28, 29, 50).

[42] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. «Overfeat: Integrated recognition, localization and detection using convolutional networks». In: *arXiv preprint arXiv:1312.6229* (2013) (cit. on p. 29).

[43] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. «Learning phrase representations using RNN encoder-decoder for statistical machine translation». In: *arXiv preprint arXiv:1406.1078* (2014) (cit. on p. 29).

[44] Alec Radford et al. «Learning transferable visual models from natural language supervision». In: *International Conference on Machine Learning*. PMLR. 2021, pp. 8748–8763 (cit. on p. 29).

[45] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. «Language models are unsupervised multitask learners». In: *OpenAI blog* 1.8 (2019), p. 9 (cit. on p. 29).

[46] *COCO Dataset*. URL: https://cocodataset.org/#home (cit. on pp. 30, 50).

[47] *Conceptual Captions Dataset*. URL: https://ai.google.com/research/ConceptualCaptions/ (cit. on pp. 30, 50).

[48] *Langdetect Python library*. URL: https://pypi.org/project/langdetect/ (cit. on p. 31).

[49] *Google Colab*. URL: https://colab.research.google.com (cit. on pp. 31, 41, 50, 67).

[50] *Emoji Python library*. URL: https://pypi.org/project/emoji/ (cit. on p. 32).

[51] *Stopwordsiso Python library*. URL: https://pypi.org/project/stopwords-iso/ (cit. on p. 32).

[52] *Hugging Face*. URL: https://huggingface.co/models (cit. on p. 33).

[53] Francesco Barbieri, Jose Camacho-Collados, Leonardo Neves, and Luis Espinosa Anke. «Tweeteval: Unified benchmark and comparative evaluation for tweet classification». In: *arXiv preprint arXiv:2010.12421* (2020) (cit. on pp. 33, 57).

[54] Yinhan Liu et al. «Roberta: A robustly optimized bert pretraining approach». In: *arXiv preprint arXiv:1907.11692* (2019) (cit. on p. 33).

[55] *NLP Town*. URL: https://www.nlp.town/ (cit. on p. 33).

[56] Tianqi Chen. *Story and lessons behind the evolution of XGBoost*. 2016 (cit. on p. 36).

[57] *XGBoost*. URL: https://xgboost.ai/ (cit. on p. 36).

[58] Onkar N Manjrekar and Milorad P Dudukovic. «Identification of flow regime in a bubble column reactor with a combination of optical probe data and machine learning technique». In: *Chemical Engineering Science: X* 2 (2019), p. 100023 (cit. on p. 37).

[59] *Scikit-Learn*. URL: https://scikit-learn.org/ (cit. on pp. 40, 41, 58).

[60] Kai-Cheng Yang, Onur Varol, Pik-Mai Hui, and Filippo Menczer. «Scalable and generalizable social bot detection through data selection». In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 34. 01. 2020, pp. 1096–1103 (cit. on pp. 46, 49).

[61] *Pytorch*. URL: https://pytorch.org/ (cit. on p. 57).

[62] *Visual Studio Code*. URL: https://code.visualstudio.com/ (cit. on p. 67).