sustainability

MDPI

Article

A Hybrid Model for the Measurement of the Similarity between Twitter Profiles

Niloufar Shoeibi [1,*], Nastaran Shoeibi [2], Pablo Chamoso [1], Zakieh Alizadehsani [1] and Juan Manuel Corchado [1]

[1] BISITE Research Group, University of Salamanca, 37007 Salamanca, Spain; chamoso@usal.es (P.C.); zakieh@usal.es (Z.A.); corchado@usal.es (J.M.C.)
[2] Faculty of Science, University of Salamanca, 37008 Salamanca, Spain; nastaran@usal.es
* Correspondence: niloufar.shoeibi@usal.es; Tel.: +34-617-939-365

Abstract: Social media platforms have been an undeniable part of our lifestyle for the past decade. Analyzing the information that is being shared is a crucial step to understanding human behavior. Social media analysis aims to guarantee a better experience for the user and to increase user satisfaction. To draw any further conclusions, first, it is necessary to know how to compare users. In this paper, a hybrid model is proposed to measure the degree of similarity between Twitter profiles by calculating features related to the users' behavioral habits. For this, first, the timeline of each profile was extracted using the official TwitterAPI. Then, three aspects of a profile were deliberated in parallel. Behavioral ratios are time-series-related information showing the consistency and habits of the user. Dynamic time warping was utilized to compare the behavioral ratios of two profiles. Next, the audience network was extracted for each user, and to estimate the similarity of two sets, the Jaccard similarity was used. Finally, for the content similarity measurement, the tweets were preprocessed using the feature extraction method; TF-IDF and DistilBERT were employed for feature extraction and then compared using the cosine similarity method. The results showed that TF-IDF had slightly better performance; it was therefore selected for use in the model. When measuring the similarity level of different profiles, a Random Forest classification model was used, which was trained on 19,900 users, revealing a 0.97 accuracy in detecting similar profiles from different ones. As a step further, this convoluted similarity measurement can find users with very short distances, which are indicative of duplicate users.

Keywords: Twitter; social media; social networking; social network analytics; DistilBERT; text similarity; natural language processing; character computing

Citation: Shoeibi, N.; Shoeibi N.; Chamoso, P.; Alizadehsani, Z.; Corchado, J.M. A Hybrid Model for the Measurement of the Similarity between Twitter Profiles. *Sustainability* **2022**, *14*, 4909. https://doi.org/10.3390/su14094909

Academic Editor: Andreas Kanavos

Received: 9 February 2022
Accepted: 14 April 2022
Published: 19 April 2022

1. Introduction

Social media platforms are now a part of the lifestyle of people of all ages. This popularity has its advantages and disadvantages. The great benefit is faster and easier communication and overcoming physical limitations [1]. Users' connections have a significant influence on what users know and how they think [2]. Social network analysis allows us to quantify the connections between individual points. It helps to find patterns in the connections that sustain society [3]. It reveals the ways in which the individual is connected to or isolated from people, groups, or populations; in other words, social network analysis shows how individuals distribute their energy across different social groups over time or explores how an idea, belief, or disease passes through the individual's network [4].

Social media analysis aims to understand people's behavior, provide more safety, and achieve higher user satisfaction. There are different types of measures that aim to counteract malicious activities, such as rumor control [5], detecting fake news and stopping its propagation [6], fake profiles and bot detection [7], and detecting duplicate profiles, which are profiles that act similarly.

As much as it is essential to have a safe society, it is crucial to guarantee the safety of the users of the virtual community. One step to make community more secure is by detecting and removing these duplicate profiles. These profiles can induce unethical thinking or activities, such as sexist ideologies [8]. Sometimes, the aim is to detect specific topics in real-time [9]. Other times, ensuring cybersecurity is a challenge because the safety of social networks is a tremendous concern due to mass user engagement. The right path towards ensuring cybersecurity involves detecting criminal activities and eliminating profiles with malicious behavior [10–12]. Considering the possibilities that social media platforms create for illegal activity, it is crucial to analyze the behavior of users.

To detect duplicate profiles, it is mandatory to know how to compare two profiles [13]. The idea of seeing identical profiles led to the implementation of a methodology for comparing Twitter users, considering three aspects: behavioral similarity, audience similarity, and content similarity. Having the information derived from the user comparison enables detecting duplicate profiles, counteracting fraud, protecting users, enforcing user-related policies, and protecting privacy, leading to a secure virtual society to make life more comfortable, safer, and more updated [14].

The questions this research aims to answer are as follows:

- How do we define the similarity of the profiles?
- In what aspects are two profiles similar?
- Which similarity measurements can calculate the distance of the two profiles?
- Which features that define the selected aspects of a profile should be used to calculate the similarity?

When it comes to social media users, it is very challenging to analyze their behavior because of the nature of human beings. It should be considered as a stochastic environment with numerous features affecting people and their behavior. The actions can be taken rationally, emotionally, or randomly. Furthermore, the chain of transformation of each act flows through all the users worldwide, as in the butterfly effect, creating a stochastic dynamic environment, which makes it more challenging to analyze and find behavioral patterns. However, the biggest challenge in this kind of research is defining the aspects by which profiles can be similar to each other [15].

This article focuses on Twitter, a social media platform enabling two-way communication and interacting with other users quickly and easily. Twitter allows users to generate content by posting "tweets" and sharing other users' content by "retweeting" [16]. The proposed architecture considers three aspects of similarity measurement, calculates the similarity, and decides whether the profiles are replicated or not. These aspects are *the audience similarity*, which interacts with the profile and its contents, calculated using the network of audience. *Behavioral similarity* is the ratio of the activities of an account, for example two accounts that constantly post the same amount of tweets in the morning between 9:00 and 12:00 a.m. *Content similarity* is measured through two strategies—the number of the same tweets/retweets and the content similarity. Content similarity is calculated using a TF-IDF text vectorizer and the cosine similarity.

To do this, the user's timeline was extracted as a list of tweet objects, which are the entities that contain all the tweet information. Then, the audience, the users interacting with the primary user, were obtained. Besides the behavioral ratios, the time-series-related features were calculated. Moreover, the content, the tweets, and the retweets the user posted were collected. To compare the audience, the inter-communications network of the primary profile was created and later compared to the other profiles' audience, the overlap of a user's audience with another was measured.

The frequency of the user's activities over time was calculated to measure the users' behavioral features and characteristics. Then, this was compared to another user's features using dynamic time warping (DTW) [17]. For instance, if a user's timeline is full of retweets, they tend to share each others' ideas. This user has a different character from those who post original posts, even though he/she shares the same concept.

To check the content similarity, two aspects were taken into account:

- How many tweets are exactly the same?
- To what degree are the concepts of the posts on different users' timelines similar? (they can be in the same language or not.)

One of the essential techniques in natural language processing (NLP) is text feature extraction, which turns the texts into a numerical vector representing the words and the sentences. In this research, two strategies were examined: a straightforward technique called TF-IDF, which calculates the vectors based on the frequency of appearance of the words, and a technique called DistilBERT [18], which is the encoding part of the Transformer architecture of the language model. DistilBERT is a pre-trained language model. Thus, it calculates the cosine similarity between these vectors. It is necessary to mention that the preprocessing for these two methods is different.

A language model is a distribution over sequences of tokens or symbols of words in a language. A good language model of English can look at a sequence of words or characters and tell the likelihood of the text being in English, explain why the phrase or sentence may be in English, and then, use this information for many different tasks. To generate text, users can sample from that distribution, put conditions on the probability distribution for the other words, and keep giving it the output. Language models are used in many tasks such as translation, summarization, chatbox, and enhancing many language-related tasks [19].

One of the developments in language models handle dependencies of any kind, but especially long-term dependencies. Recurrent neural networks (RNNs) suffer from short-term memory. Therefore, for longer paragraphs, RNNs may miss important information from the beginning [20]. Long short-term memory (LSTM) networks are a particular kind of RNN capable of learning long-term dependencies. They were designed to fix the long-term dependency problem. LSTM has internal mechanisms called gates that have the stream of information and decide which parts of the input to pay attention to, which features to use in the calculation, and which parts to ignore [21].

DistilBERT is lighter and faster, have 40% of the size of standard BERT, saving 97% of its language understanding capacity, but 60% faster. The output vector size is 768, meaning each sentence will have a fixed-size vector with 768 values [18].

The novelty of the proposed method lies in defining a new distance metric to be used to measure the similarity between Twitter profiles. The proposed hybrid model covers similarity from three aspects: graph of audience, character computation and behavioral measurement, and content similarity. Apart from considering a more comprehensive range of information, the novelty of the work lies in the features considered in the model.

This paper is organized as follows: In Section 2, the related work is presented. Then, in Section 3, the architecture of the proposed method is described. In Section 4, a successful case study is outlined, and its results are overviewed. Finally, in Section 5, conclusions are drawn, and future lines of research are discussed.

2. Review of the State-of-the-Art

Each social media platform has its unique characteristics and privacy and policy rules, which cover issues ranging from the user's usage to the developers' manual. For instance, Twitter allows researchers to extract public information via official Twitter APIs by signing up on the Twitter developer team's platform, generating the "tokens" for data extraction, and conducting academic research to make improvements. However, in general, most of the research in this area is related to taking advantage of social network data and applying artificial intelligence algorithms, such as machine learning methods (supervised and unsupervised), deep learning, and graph theory. In this paper, the focus is on calculating the similarity of the profiles on Twitter by defining a more sophisticated concept of distance between Twitter users.

A social network can be interpreted as a complex network graph connected by edges. The nodes represent the users in the network, and the edges define the connections between these users. Social network analysis requires specific analysis tools. Akhtar et al. in [22]

conducted a comparative study of these tools in general graph analysis and social network analysis. They conducted a comparative study of four social network analysis tools (NetworkX, Gephi, Pajek, and IGraph) based on platform, runtime, graph type, algorithm complexity, input file format, and graph features.

The broad variety of users and content on online social networking sites (OSN) has led to the fear of identity theft attacks (profile cloning), malware attacks, or structural attacks performed by cybercriminals. Profile cloning involves the stealing of their identities and creating duplicate accounts with the existing users' credentials. Chatterjee et al. proposed a means of supervising the threat of profile cloning in social networks. Users can use it to prevent cloned and fake profiles and identity theft [23].

Choumane et al. in [24] proposed the characterization of the Twitter users and measurement of similarity between two different aspects of the social graph and the user's content. They extracted the graph metrics from the network of followers and stated that the more friends of user "u" following user "v," the more similar users "u" and "v" were. They also applied LDA [25] to the content shared by the users and compared the top-10 topics with each other.

In Ref. [26], Vajjhala et al. proposed a novel recommender system that works on the basis of comparing the users with each other to determine their preferences from their Twitter profiles. They recommended personalized products depending on the user's likes derived from analyzing the tweets they shared on their timeline.

In Ref. [27], Dahiya et al. proposed a framework for finding the similarity between the Twitter users. They took eight parameters into account and calculated similarity measurements for each of these eight parameters. The weighted average of these values (floats between 0 and 1) was returned as the similarity between the two Twitter users. These parameters were extracted from the primary features extracted from the Tweet objects.

Semantic analysis is a powerful technology in NLP applications, among them, text similarity estimation, text classification, and speech recognition. Chen et al. introduced a framework for semantic similarity detection for deep reinforcement learning for the Siamese attention structure model (DRSASM). It automatically detected the word segmentation and word distillation features and proposed a new recognition mechanism model to improve the semantics [28].

Semantic similarity detection in text data is one of the challenging obstacles of NLP. Due to the versatility of natural language, it is challenging to represent rule-based methods for detecting semantic similarity patterns. Chandrasekaran et al. in [29] determined the evolution of several existing semantic similarity methods and reviewed their pros and cons. They were classified by the underlying policies as corpus-based, hybrid approaches, knowledge-based, and deep-neural-network-based methods.

Park et al. presented a cosine-similarity-based methodology to enhance text classification performance. To increase the precision of the classifiers, this methodology merges cosine similarity and conventional classifiers, and then, the conventional classifiers with cosine similarity were named enhanced classifiers. The enhanced classifiers were applied to famous datasets such as 20NG, R8, R52, Cade12, and WebKB, and they had notable accuracy improvements. Furthermore, word count and term frequency-inverse document frequency (TF-IDF) are more suitable in terms of the performance of the classifier [30].

In [31], a detection technique was proposed for discovering fake and cloned profiles on Twitter. To detect profile cloning, they used two methods: similarity measures and the C4.5 decision tree algorithm. The similarity of characteristics and the similarity of network relations were analyzed. C4.5 applies a decision tree by considering information gain. These two methods helped detect and prevent clone profiles.

A framework for finding cloned profiles in social networks was stated in [32]. It analyzes user profiles, friends and follower networks, and posting habits. This framework has three parts: Twitter crawler, attribute extractor, and cloning detector. The best classification performance was obtained with the decision tree, and the average accuracy of classifying the real or fake posts was 80%.

BERT [33] is a method of merging topics by pre-trained contextual representations. Peinelt et al. proposed a unique, topic-informed BERT-based structure for pairwise semantic similarity detection between two short documents by combining topic modeling and BERT. This advanced architecture outperformedstrong neural baselines beyond different classes of English language datasets. Adding topics to BERT helps determine domain-specific problems [34].

Knowing which text feature extraction strategies perform better depends on the text being analyzed, mostly on its length. For example, in [35,36], D. Carun et al. performed sentiment classification on financial news, and they discovered that Distilbert performs better than TF-IDF in text feature extraction, with a 7% improvement in accuracy. On the other hand, in [37], I. Vogel et al. investigated methods of monitoring and defining the factors that would help detect the Twitter users who spread hate speech. They realized that the simple n-grams feature extraction and traditional machine learning models, such as SVM, perform better than BERT's feature extraction and Bi-LSTM. The variety of the results of the investigations in selecting the best text feature extraction methodology for NLP tasks was one of the motivations for examining both methods in this work.

Table 1 represents the aspects covered by the most recent research related to this area. The "X"s show that the article covers the similarity aspect, and the "-"s indicate the opposite. It is worth noting that the proposed method considers more features and covers all three aspects of similarity between two users.

Table 1. A brief summary of some of the aspects covered by the most related publications.

Article	Network of Audience Similarity	Behavioral Habits Similarity	Content Similarity
Akhtar et al. [22]	X	-	-
Choumane et al. [24]	X	-	X
Vajjhala et al. [26]	-	-	X
Chen et al. [28]	-	-	X
Dahiya et al. [27]	-	X	X
Chandrasekaran et al. [29]	-	-	X
Park et al. [30]	-	-	X
Chatterjee et al. [23]	X	-	X
Sowmya et al. [31]	X	-	-
Punkamol et al. [32]	X	X	-
D. Carun et al. [35,36]	-	-	X
Vogel et al. [37]	-	-	X
Our Proposed Hybrid Model	X	X	X

The following section presents the proposed method, combining the information extracted from the three aspects of the profile behaviors. A few examples of each aspect are reviewed in the related work. These ideas helped design and implement the proposed system.

3. Proposed Model

This model focuses on analyzing Twitter users. However, the proposed methodology can be used on different social media platforms with modifications. Twitter has a unique feature among all social media: on Twitter, users can respond to each others' tweets and "like" each other's tweets or leave a comment to share their opinions and viewpoints. Tweets can include text, photos, videos, and links. Users can also share the status of other users' tweets by retweeting them. The data related to the tweets and some information about the profiles are provided as a Tweet object in JSON format [38]. This section presents the platform's architecture for measuring the similarity of profiles. It extracts data from a user's Twitter timeline, analyzes them, and transforms them into meaningful information. Below, an overview of the functionality of the proposed method is given.

The first step is extracting the recent tweets of a profile by extracting the user's timeline and storing it as a list of JSON files. Then, the data are restructured in the character computation and behavioral measurement component. More advanced calculated features are created from the selected primary features, including time series features and ratios indicating the users' behavior, characteristics, and habits, such as the interaction level of the user with others during the time and the number of tweets, retweets, and likes per hour, day, and month. A user whose timeline consists of retweets more than tweets, considered a spreader, is different from someone who posts original tweets, even though he/she is talking about the same concept.

In parallel, from the JSON file extracted from the user's timeline, the audience is the people who are interacting with the chosen profile, defined by the replies and the retweets. In parallel, all tweets of a profile are selected by its language. Non-English ones are translated to English; depending on the feature extraction method, they are preprocessed and turned into vectors. The results showed that the two selected feature extraction strategies of TF-IDF and DistilBERT have very similar results. TF-IDF has slightly better performance; therefore, TFIDF was chosen to be implemented in the platform. The output of all these components will be handed to the Similarity Measurements component. A respective similarity measurement was applied for each aspect of these new features, explained in detail in each relative sub-section. Algorithm 1 represents the process as pseudo code.

Algorithm 1 Proposed method's algorithm.

Input: Screen_names of the Twitter profiles.
Output: Similarity of the profiles
Step 2, 3, and 4 are executed in parallel **Begin**
for **each** Twitter user
1. Extract the timeline
2. Character computation and behavioral measurement
2.1. Extracting primary features
2.2. Extracting advanced calculated features
3. The audience network
3.1. Extracting the audience of the user
3.2. Building a set of users who are interacting with the user
3.3. Finding the strong friendships of the user
4. Content preprocessing
4.1. Text preprocessing
4.2. Text feature extraction (text vectorization)
5. Similarity checking
5.1. Character and behavior similarity approximation
DTW
5.2. The audience network similarity detection
The strong friendship overlaps
The overlap between sets of audiences using the Jaccard similarity
5.3. Content similarity measuring
The number of same tweets/retweets
Cosine similarity of the whole content
end for
End

This architecture is efficient because it has been designed in the most parallel way possible and consists of four components described in the following subsections. The outputs of this model aim to provide a better understanding of how similar two profiles are by calculating the distance between the users from the mentioned points of view.

The proposed architecture is presented in Figure 1. The designed system aims to calculate the similarity of the profiles based on their network of the audience, behavioral

traits, and content similarity. This architecture consists of five main components: timeline extraction, character computation and behavioral measurements, the audience network, and content processing. These aspects are discussed in their respective subsections.

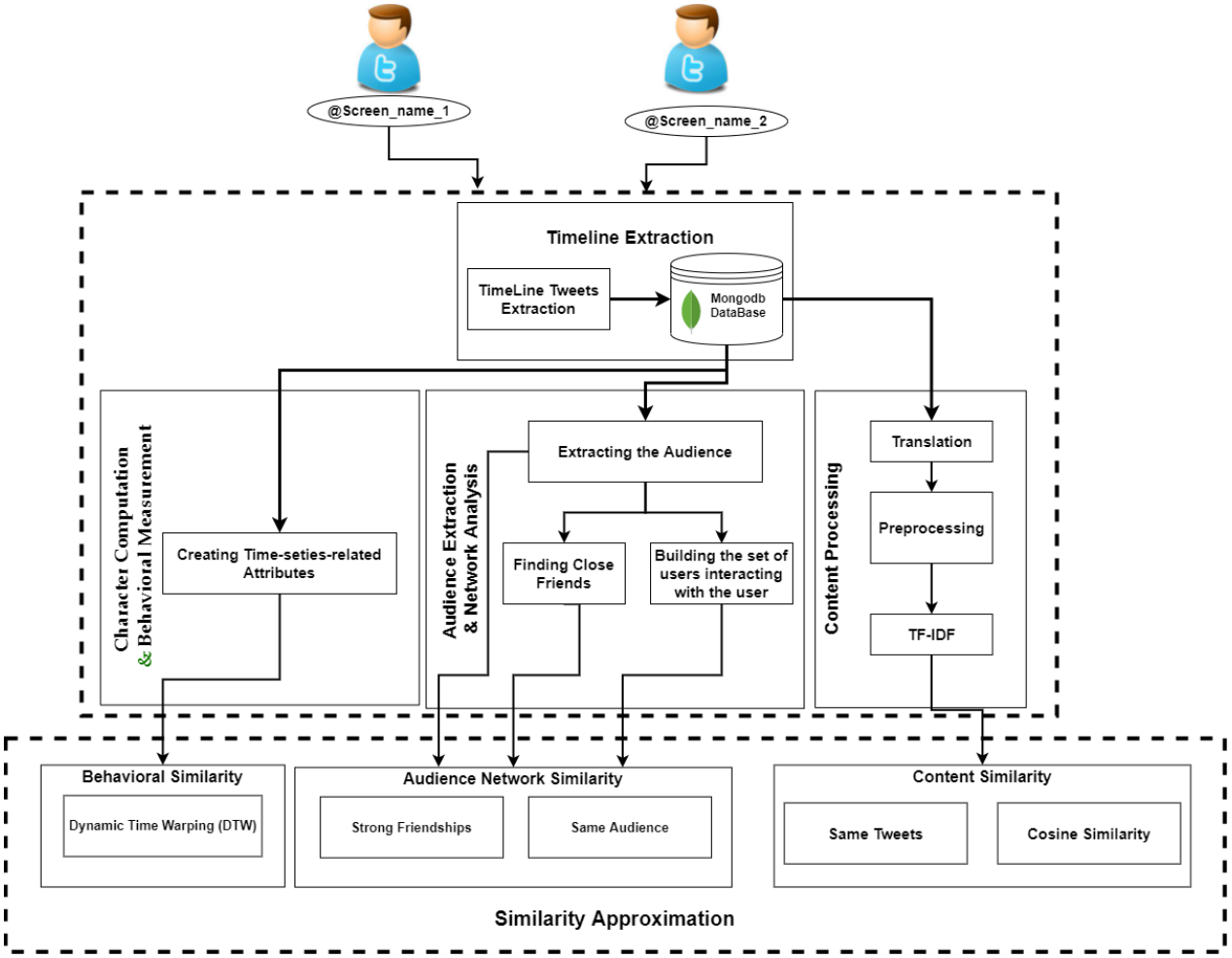

Figure 1. The proposed model for similarity measurement of the profiles.

This architecture is efficient because it has been designed in the most parallel way possible and consists of four components described in the following subsections.

3.1. Timeline Extraction

In this research, the aim is to calculate the similarity between different Twitter profiles. The first step is to extract the user's timeline using a component called Timeline Extraction. There are many scraping-based ways to capture this information. Still, the one aligned with the data ethics provided by the Twitter development team is data extraction using official Twitter APIs, which were used for the implementations. In this component, the official Twitter API, which is provided by the Twitter development team, was utilized [39]. To this end, the Twitter developer platform considered the research proposal and its motivations. After reviewing the request, the tokens were provided by Twitter. Therefore, the study is ethical, and the data are tracked by Twitter.

When extracting Twitter data using official Twitter APIs, the information of each tweet is saved in a JSON file called a tweet object. The tweet object holds information regarding each tweet: the text it contains, the time at which it was posted, the geo-location from which it was posted, the number of interactions, and also the information about the user, such as the number of followings/followers, user bio, name, and last name. This information is considered the primary raw information collected from each tweet.

One of the challenges of using the official API is that it allows for little information [40]. When extracting the users' timeline, it provides the 3200 most recent tweets. The profiles' timeline contains all the users' activity information and the content they shared during that time. The process of timeline extraction is a query with the user's screen_name and related timeline, containing tweet objects, holding the information about each tweet and the user. The advanced features are calculated from the raw data by considering the number of activity ratios during a given time: the number of tweets per hour/day, the number of retweets per hour/day, and the number of mentions per hour/day. These features are presented in Table 2. The output of this component is a list of tweet objects saved in JSON format.

Table 2. The advanced calculated features extracted from the user's most recent 3200 posts.

Feature	Description
Original tweets per (hour/day)	The number of original tweets the user has written himself/herself per (hour/day) and posted among the recent posts on his/her timeline
Retweets per (hour/day)	The number of retweets the user has posted per (hour/day) among the recent posts on his/her timeline
Quotes per (hour/day)	The number of retweets the user has posted with an additional opinion per (hour/day) among the recent posts on his/her timeline
Mentions per (hour/day)	The number of times the user has mentioned others per (hour/day) among the recent posts on his/her timeline
Replies per (hour/day)	The number of replies the user has made per (hour/day) among the recent posts on his/her timeline
Statuses per (hour/day)	The number of times the user has published statuses (tweets + retweets + quotes + replies) per (hour/day) among the recent posts on his/her timeline
Likes per (hour/day)	The number of times the user has liked others' tweets/retweets per (hour/day) among the recent posts on his/her timeline

3.2. Character Computation and Behavioral Measurement

The science of character is the merging point of computer science and psychology, in which the criteria of the personalities are defined and conducted. Every person is a unique combination of these core virtues. Going in the direction of strengthening these criteria leads to a happier experience and higher satisfaction level. Character computing is a psychological technique whose computational models incorporate stable personality attributes and cognitive, affective, and motivational state features and behavioral indicators to explain the dynamic relations among the situation (S), person (P), and behavior (B). Figure 2 shows a comprehensive relation chain of situation, person, and behavior [41]. A person's personality affects his/her behavior and the situation; for example, a positive extroverted person can radiate happiness. A person's behavior and actions can put them into various situations and affect their personality; for example, lying could become a habit if it takes place over a longer period of time. Additionally, a situation such as war could affect someone's personality and behavior. Therefore, it can be considered as a chain where each aspect entails another, creating a loop.

In the case of applying character computing to social media user behavior mining, there is no information about the users' personalities. Still, the user's behavior is interpretable by considering his/her interactions during a period, considered as cognitive character computing for users of social media platforms. The methodology is explained in the following.

In the proposed method, character computation and behavioral measurement component, data are restructured, and more complex concepts are derived from the primary features existing in the tweet objects. Monitoring and measuring the users' behavior during time helps derive the behavioral tendencies and compute the user's character. The behav-

ioral inclination of the user is defined by considering the ratios of activities during the time. Extracting these time-series-related features enables us to extract behavioral patterns.

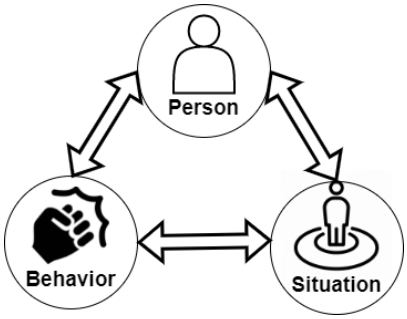

Figure 2. The impact cycle of person, situation, and behavior [41].

The biggest motivation behind the proposed model is to distinguish users from different characteristic points of view. A spreader tends to retweet from others more than writing original tweets. It is understandable that an early riser is more active during the day, especially early in the morning. Alternatively, a nocturnal person is one who has the highest activity ratios at night. Even though these profiles share the same content, they do not represent the same character. These features are presented in Table 2.

After measuring the behavioral features, the distance between the same time series features in two profiles is calculated. This distance represents how similar the behavioral character of users is without considering the content. DTW [42] is a suitable distance similarity measure that allows the comparison of two time series sequences with different lengths and speeds. This algorithm is a perfect choice because the time series's length in various features varies and depends on how much the user was active recently. In other words, a user with a high ratio of activities can make 3240 tweets in one month; however, another user has this amount of posts in three months. The DTW algorithm computes and returns a DTW similarity measure between (potentially multivariate) time series [43]. In the proposed model, the distance between two behavioral features exposes a quantitative value representing how similarly profiles act during the extracted timeline. For example, the distance between the mean number of posted statuses per day is higher between an early riser user and a nocturnal one.

Table 3 is an example of the similarity of two different features. Noticeably, @User_4 and @User_5 are more similar in the behavioral ratios (likes, posted statuses). The results of applying the DTW algorithm are presented in Table 3. As the output of DTW represents a distance, a lower distance indicates a higher similarity level.

Table 3. The distance between behavioral features of different users calculated using DTW.

Users	Statuses Posted	Retweets Posted	Tweets Posted	Likes	Retweets Received
@User_1 & @User_2	104.50	10.23	10.23	634,612.10	73,611.20
@User_2 & @User_3	114.26	6.78	12.13	48.90	611.73
@User_4 & @User_5	102.44	0.40	0.05	0.10	0.00

3.3. Audience Network

The attitude of any human being towards his/her environment and others can tell much about them. One of the most significant sources of information is the users connected to a specific user. An audience is a group of users who interact via retweeting, quoting, replying, and mentioning. The aim of this component is to calculate audience-based similarity measurements. Mapping this information on a directed graph makes analysts derive further information by simply considering the nodes as users and the edges as the connection. That is one of the possible ways of retweeting, quoting, replying, and

mentioning. Categorizing the audience based on the frequency of links, called weights of a digraph, is a way of measuring the acquaintanceship of the profile audience. In this sample case, the scenario is as below:

- @User_1 has mentioned @User_2, 4 times;
- @User_2 has retweeted from @Use_3 twice;
- @User_1 has quoted a tweet from @User_3, 2 times;
- @User_4 has replied on the statuses of @User_1 and @User_3, once each.

The relationship matrix of this scenario is shown in Table 4 as well as the directed graph of the example scenario in Figure 3. It should be noted that the source node is the user who has posted the status, the target node is the user whose screen_name is mentioned in the status, and the weight is the frequency of time the source user reached the target user.

Table 4. The network relationship of the sample scenario.

Source	Target	Weight
@User_1	@User_2	4
@User_2	@User_3	2
@User_1	@User_3	2
@User_4	@User_1	1
@User_4	@User_3	1

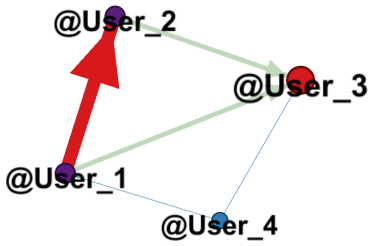

Figure 3. The directed graph of users interacting with each other in the sample scenario.

Strong friendships between users are defined on the basis of the frequency of repetition of the links between two users; in this case, based on the network relationship sample, the *weight* determines the strength of a friendship. Having each user's close friends makes it possible to compare these sets and see how they overlap.

In the "audience network" component, first, the relationships between people and the list of the screen names of the audience in contact with the primary user are extracted. The main user whose timeline is extracted in the network is in the middle of all the other nodes that are connected to this node. This graph has a sparse or crowded star shape, depending on its audience's size. Therefore, it is transformed into a set interpreted as all users interacting with the content of the main profile.

Two approaches are applied to compute the overlap of the users' audiences. One compares the users' strong friendships with each other, and the other considers the directed graph of the primary user audience as a set and compares the overlap of the settings.

Jaccard's similarity has been applied to the set of the audience of different profiles to calculate the similarity between the two sets. It is a classical measure of the similarity between two sets, introduced by Paul Jaccard [44]. Given two sets of audiences of @User_A and @User_B, Jaccard's similarity is measured by dividing the number of nodes from the audience of @User_A that exists in the audience of @User_B as the total number of nodes minus the same nodes in both sets. It should be noted that the number of users in an audience is not the same; hence, the Jaccard similarity of the sets is a convenient metric that considers the length of different settings.

$$Jaccard(Set\_1, Set\_2) = \frac{|Set\_1 \cap Set\_2|}{|Set\_1 \cup Set\_2|} \quad (1)$$

The intersection of two sets points out the common nodes between two of them, and the union of the sets means to sum the number of the audience of each profile, but remove the ones that are repeated (the intersection).

3.4. Content Processing

This step investigates the contents (tweets) shared by profiles. The similarity of tweets is calculated in two aspects: the number of the same tweets (each tweet is treated separately) and the similarity of the content (in this case, tweets are treated as an extensive document). The same tweets are counted by checking the contents on two timelines, one by one. Moreover, the tweets on the timeline are added together as one document, and these documents' similarity is measured. Two different text feature extraction strategies were applied to turn the words into the vector of numbers to calculate the similarity in content: TF-IDF and DistilBERT. The text needs to be preprocessed before using text vectorization (text feature extraction). Each of these methods requires a particular preprocessing approach.

It is worth mentioning that this component aims to calculate how similar the tweets coming from the two profiles are. Users can post these tweets in different languages. To unify these tweets, they need to be turned into the same language, English in this case. Google translate API [45] covers a very vast range of different languages. Therefore, before performing any further analysis, the languages of the tweets were unified into English. Furthermore, it is necessary to check the dictation of the words because due to the limitation of the number of characters possible to post as a tweet, which is 280 characters, users usually abbreviate the words to add more information to the tweet. Hence, returning these abbreviations to the original terms is necessary. After preparing the text of the tweets, two mentioned feature extraction methods were applied. The following subsections give more information about each process.

3.4.1. Text Feature Extraction Using DistilBERT

Language models are deep neural network models that are context-sensitive and comprehend the language and probability of appearance of the words in sequence. The quality of the performance of NLP tasks highly depends on how extensive the network is and the data on which it is trained [46].

As language models are deep neural network models that deal with sequential data (i.e., words in sentences), deep neural models are commonly used to implement language models. Recurrent neural networks (RNNs) and long short-term memories (LSTMs) are famous examples. The problem with RNNs and LSTMs, apart from the complexity of the network, long training time, and computation expenses, is that the memory of these networks is limited, meaning that the longer the text, the more information is lost from the beginning of the text. LSTMs are an improved version of RNNs that can selectively remember the past by employing a gating mechanism. Bi-LSTM is a version of LSTM that can move through the sequence in both ways [47]. However, the data must still be passed through the network sequentially, which is a considerable disadvantage. To solve all these challenges, Transformers, which are attention-based models [48], have appeared.

The Transformer consists of two key elements, the encoder and decoder. The encoder learns what grammar is, what context is, and what language is [49]. It contains a self-attention mechanism and a feed-forward neural network. Self-attention is an attention mechanism correlating different forms of a single sequence to estimate the design of the series. The decoder is a word embedding concatenated with a context vector made by the encoder. BERT is the stacked encoders. It is used in many tasks such as neural machine translation, question answering, sentiment analysis, and text summarization. Pre-training BERT can explain these tasks to learn the language and fine-tune it to learn particular tasks. The training of BERT has two phases. The first phase is pre-training; the model understands the language and context. The second phase is fine-tuning; the model learns the language, but does not know how to solve this problem.

Pre-training aims to make BERT learn what a language is and what context is. BERT learns language by training on two unsupervised tasks simultaneously. They are masked language modeling (MLM) and next sentence prediction (NSP). For MLM, BERT takes in a sentence with random words filled with masks. The goal is to output these masks' tokens. It helps BERT understand a bi-directional context in a sentence for predicting the subsequent sentences because BERT takes two sentences and decides if the second sentence supports the first. This kind of binary classification problem helps BERT understand context over different sentences themselves and use both of these together [50].

In the fine-tuning phase, BERT is trained on particular NLP tasks by training both MLM and NSP to reduce the merged loss function of the two strategies. Rather than LSTM, which has to hang on to an enormous amount of memory, BERT can selectively look at the relevant things, and the system learns where to look and where to pay attention [51].

DistilBERT is a smaller, quicker, and more affordable version of BERT [18], which includes 40% of the size of the original BERT, but maintaining 97% of its language comprehension capability and being 60% quicker. DistilBERT transforms the input sentence into a fixed-size vector with 768 values. The significant advantage of using DistilBert for feature extraction is that all the words in the sentence are vectorized simultaneously in parallel, but: Will it have a notable impact on the results compared to the less complex methods? The primary motivation is to try the process explained below and compare the results.

3.4.2. Text Feature Extraction Using TF-IDF

TF-IDF stands for term frequency-inverse document frequency. It is a well-liked algorithm for converting text into a meaningful representation of numbers to adapt machine learning algorithms for prediction. The count vectorizer provides the frequency count for the word index, and TF-IDF considers the overall word weight document [52]. TF-IDF is used to extract features from Twitter users' content, which is a document created by merging the tweets posted on the timeline of each profile. Hence, two documents are compared with each other, but the vocabulary set is constructed from both of these documents. The documents with similar content will have similar vectors. The formulas below represent the process of calculating the vector for each word using TF-IDF [53].

$$TF\text{-}IDF = TF * IDF \tag{2}$$

$$TF(t,d) = \frac{f(t,d)}{\sum_k f(w_k,d)} \tag{3}$$

$$IDF(t,d) = log(\frac{N}{1+df_t}) = log(\frac{2}{3}) \tag{4}$$

where:

$f(t,d)$ represents the number of occurrence of term t in document d;

N is the total number of documents in the corpus, equal to two in this research, as each user's tweets are saved in a document and the comparison is made between two users;

df_t is the number of documents containing the term t; in this case, the value of df_t is equal to two.

TF-IDF performs slightly better than DistilBERT. Both possible models transform the text into the related vector, and the similarity metrics were compared as an experiment. TF-IDF is a straightforward model with the benefits of a simple model, meaning that it is fast, and DistilBERT, explained in detail, has a better understanding of the context of the language.

After applying this vectorization method, the distance between these vectors is calculated using the cosine similarity algorithm. *Cosine similarity* is a measure used to assess the similarity of documents, regardless of their size. Mathematically, it measures the cosine of the angle between two vectors projected in multidimensional space. Cosine similarity is helpful because two similar documents can be separated by a Euclidean distance (due

to the size of the document), but oriented closer to each other. The smaller the angle, the greater the cosine similarity is [54].

4. Case Study and Results

To conclude all that has been discussed in this paper, a hybrid model is presented considering a novel, more comprehensive range of features through methods such as cognitive user character computing, building a network of the users' audience, along with obtaining a contextual understanding of the posted tweets.

These three aspects together cover a broader understanding of the users. Moreover, they make the measurement of user similarity a less complex task. Nevertheless, calculations regarding users on social media platforms continue to be a very challenging task. Further analyses and applications are required, ranging from finding duplicate profiles and spammers, classifying users based on their characters and tendencies, marketing to public health concerns, and even cybersecurity.

The evaluation of the proposed solutions and methods for solving social media-related problems, especially on Twitter, is a sophisticated task. Due to the policy changes of Twitter over the last few years, there are very many research articles, such as [55], that have addressed the same problems, but proposed solutions under different policies and restrictions. Furthermore, the way in which the authors have grouped the data makes them lose the sense of living in different time zones because when a query is performed, @user_X can be a user from Japan and @User_Y from the United States of America. Twitter does not reveal the geo-location data. Therefore, both of these queries are matched with the timezone of the person making the query; it does not make sense to compare a user's behavior at night with another one in the early morning. However, suppose the user's activity ratios are considered time series. In that case, the DTW algorithm is more flexible when working with repeated patterns and identifies the difference in time zones.

The search for solutions under different constraints has led to the creation of different solutions to the same problem, by improving the technologies and algorithms and dealing with various rules. For instance, in [56], published in 2013, which focused on validating the model proposed by the authors, they used and published a dataset of users and the respective identity of each user. This was possible in 2013, but since 2016, the policies have been changed, and they are much more restricted, forcing us to design solutions considering other points of view.

A balanced dataset was created using 100 U.S. politicians and Senators and 100 top singers as a case study. The dataset was created by comparing all the possible *tuple selections of two Twitter users*. Therefore, the size of the final dataset was 19,900, containing the comparison values calculated by this model and labeled manually, considering the *singers* who are similar to each other and the *politicians* who are alike. They are similar; however, they are not identical. Furthermore, politicians and singers are different.

To create this dataset, the timelines of each user in the list of the selected singers and politicians were extracted using official Twitter APIs. Then, for each user's timeline, consisting of the most recent 3200 tweets, the set of the audience of the user, primary and advanced calculated features, and the tweets' texts were extracted in parallel. In the next step, two by two, the similarity measurements of the timelines of these profiles were calculated, and then, this dataset was labeled manually.

Figure 4 represents a comparison of the activity level of the three selected profiles. As shown, @User_1, who is Joe Biden, tends to post original tweets more than retweeting others; his posts have a higher user engagement by having higher retweet and favorite ratios, and he tends to post neutral facts. On the other hand, @User_3, Hillary Clinton, a female politician, retweets more. On the contrary, @User_2, Jennifer Lopez, a singer, tends to post positive content, mostly her ideas, rather than facts. She has a steady behavior in posting tweets and retweets, and her fans also have a constant engagement compared to the other two profiles. Furthermore, both politicians have a higher user engagement on the weekends.

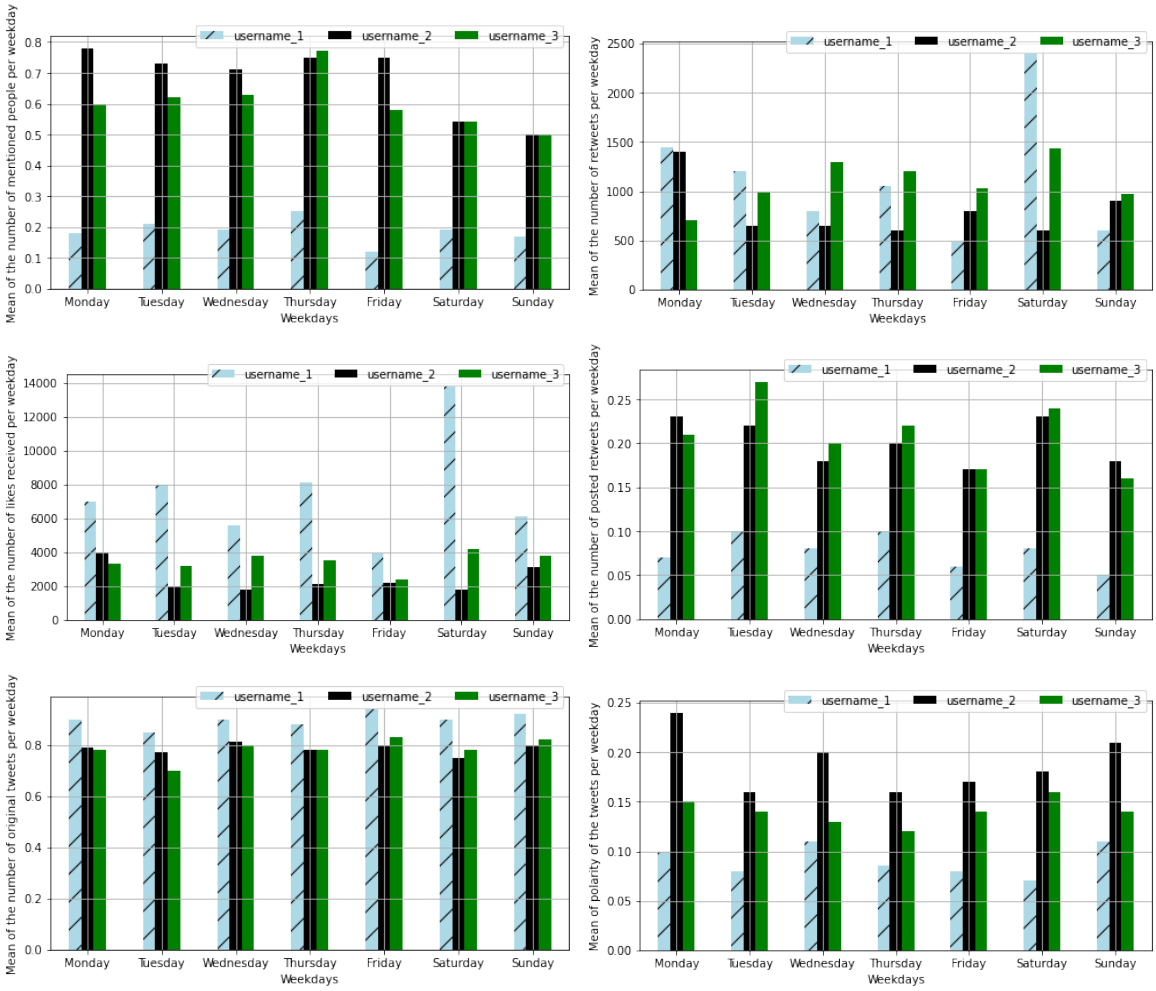

Figure 4. Mean of behavioral features by weekday for Joe Biden, Jennifer Lopez, and Hillary Clinton.

The dataset was divided into training and test datasets with the ratios of 75% and 25% using a stratified train–test split to select an even-handed number of samples from each category (i.e., similar, not similar) to keep the training and test sets balanced and fair.

Figure 5 represents the distribution of the feature values after applying the t-distributed stochastic neighbor embedding (t-SNE) [57] dimension reduction on the training dataset. Table 5 shows the configuration of the t-SNE model.

Table 5. The hyperparameters of the t-SNE model.

Parameter	Value	Description
n_components	2	Dimension of the embedded space.
perplexity	10	The perplexity is related to the number of nearest neighbors used in other manifold learning algorithms.
n_iter	5000	Maximum number of iterations for the optimization. Should be at least 250.
random_state	42	Determines the random number generator. Pass an int for reproducible results across multiple function calls.
n_jobs	−1	The number of parallel jobs to run for neighbors' search.

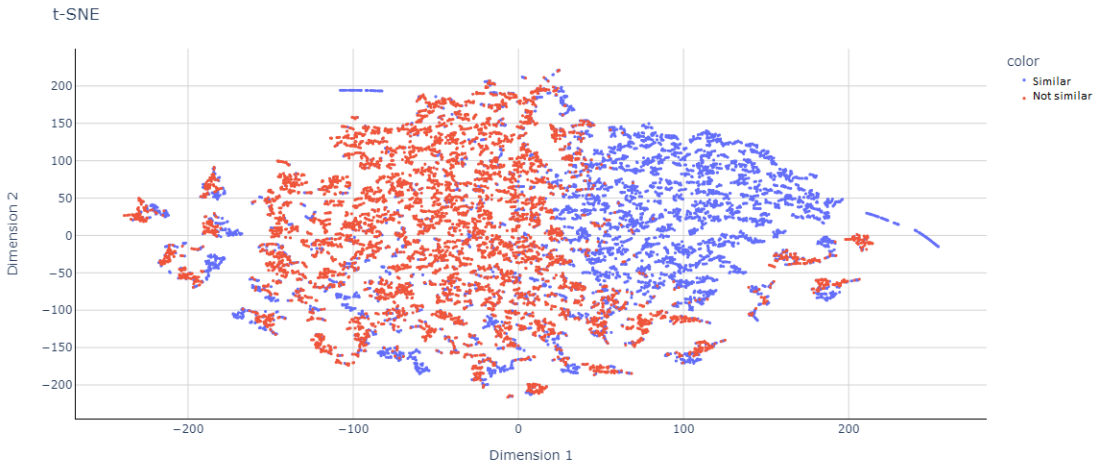

Figure 5. Distribution of the training dataset.

As demonstrated, the similar users are close to each other on the right side of the plot, and no similar users are grouped on the left side. However, there are some overlaps. As discussed in Figure 4, even though Jennifer Lopez and Hillary Clinton are different in the category, they have a similar behavior. Compared to Hillary Clinton and Joe Biden, who are both politicians and therefore are similar, one of the reasons to be investigated in the future is the bias associated with behavior and gender.

These training and test sets were then used to train the models and to measure the proposed model's performance. Different classification models were trained using the training dataset and tested on the test dataset. Table 6 represents some of the models' hyperparameters [58,59] used for randomized search. The best model was achieved by fine-tuning these hyperparameters.

Table 6. The hyperparameters tuned for optimization of each model.

Model	Hyperparameters	Description
SVM	C	Regularization parameter.
	Kernel	The type of kernel has been used in the algorithm.
	gamma	Kernel coefficient
	verbose	If true, it makes the result verbose.
	max_iter	Maximum iteration.
Random Forest classifier	n_estimators	The number of trees in the forest.
	max_depth	The maximum depth of the trees.
	min_samples_split	The least amount of samples for opening a node.
	bootstrap	If it is true, the bootstrap is used to create the trees.
	n_jobs	The number of parallel searches in the neighbors.
	verbose	Control verbosity when it is being trained and predicting.

After the best models were found, they were tested on the test dataset. Table 7 shows the results and the performance of each model.

The results showed that the random forest classification had a slightly better performance than the other models and determined whether the users were similar or not with an accuracy of 0.97. This paper proposes a hybrid model for measuring the distance between Twitter profiles. The result of the case study indicates that the proposed system is a convenient distance metric for comparing Twitter users. However, the trained classifier does not aim to detect the similarity of all types of users, but seeks to represent how well the distance metric is performing. To classify the similarity level of Twitter profiles with supervised classification models, another dataset is needed with a greater variety of users

or clustering methods based on the proposed distance metric, which will be investigated in the future. Moreover, the users with very small distance metric values were suspected to be duplicate profiles so that the proposed distancing metric can catch duplicate profiles.

Table 7. The evaluation metrics of the classification models; trained on the proposed dataset by employing TF-IDF and DistilBERT for text feature extraction.

Model	Classes	Precision	Recall	F1-Score
SVM + TF-IDF	Not similar	0.94	0.94	0.95
	Similar	0.96	0.94	0.95
Random Forest classifier + TF-IDF	Not similar	0.96	0.98	0.97
	Similar	0.98	0.96	0.97
SVM + DistilBERT	Not similar	0.95	0.94	0.96
	Similar	0.95	0.94	0.95
Random Forest classifier + DistilBERT	Not similar	0.90	0.96	0.93
	Similar	0.88	0.92	0.90

5. Conclusions and Future Work

Many studies have been carried out with the aim of quantifying the behavior of users who interact with each other on social media platforms. However, the most optimal method of comparing these profiles remains undetermined. It is important to identify ways in which social media profiles can be characterized and how these characteristics may be comparatively analyzed. In this paper, a method was proposed to calculate the distance between two profiles. The extent of similarity between the profiles was measured in three different ways: the behavioral ratios, the graph of the audience, and the contents they post. After extracting the recent 3200 tweets from the timeline of each profile, using official Twitter APIs, the data were preprocessed in three ways regarding each aspect of the similarity measurement.

First, the behavioral ratios were calculated using DTW, which calculates the distance between time series features. This measurement enables us to understand better the difference between the behavioral ratios of activity habits, such as the number of posts in a day, retweets posted by the user, and/or when the user is more active during the day. Moreover, there is extra information about the ratios of user engagement in a given user profile because the number of likes and retweets of the profile's audience per day was considered. The next step involved extracting the user's audience by defining the relationship between the profiles, in terms of replies, retweets, quotes, and mentions. At this point, a network of the user's audience was built, showing the interaction that takes place between its members. Using the Jaccard similarity, the two selected user sets were calculated. The results showed that the users in the same sector category had more similarities in the audience graph. Finally, the number of the same tweets was calculated in the content similarity measurement. Then, based on the content, all the tweets were unified into the same language, English. The text was preprocessed by employing NLP techniques: tokenization and lemmatization. Then, two different vectorization methods were applied, TF-IDF and DistilBERT, to turn the words into their respective vectors. The similarity between the two vectors was calculated by using the cosine similarity. The distribution of the calculated similarities presented a similar pattern; therefore, the simplicity of performing the vectorization in a semi-real-time manner was due to the character limitation on posting a Tweet; hence, TF-IDF was chosen and implemented in the model.

Future lines of research will focus on gender and its effects on the similarity between the users of Twitter. Furthermore, creating a generic model that can combine the information from different data sources from social media platforms using the deepint.net platform [60] is planned. The abilities provided by deepint make it a perfect choice for implementing the proposed model.

Author Contributions: Conceptualization, N.S. (Niloufar Shoeibi); Data curation, N.S. (Niloufar Shoeibi) and Z.A.; Formal analysis, N.S. (Niloufar Shoeibi); Funding acquisition, J.M.C.; Investigation, N.S. (Niloufar Shoeibi), N.S. (Nastaran Shoeibi), P.C. and J.M.C.; Methodology, N.S. (Niloufar Shoeibi), P.C. and J.M.C.; Project administration, P.C. and J.M.C.; Resources, P.C. and J.M.C.; Software, Z.A.; Supervision, P.C. and J.M.C.; Validation, N.S. (Niloufar Shoeibi), N.S. (Nastaran Shoeibi), P.C. and J.M.C.; Visualization, N.S. (Nastaran Shoeibi) and J.M.C.; Writing—original draft, N.S. (Niloufar Shoeibi), N.S. (Nastaran Shoeibi) and Z.A.; Writing—review & editing, N.S. (Niloufar Shoeibi), P.C. and J.M.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research has been supported by the project "Intelligent and sustainable mobility supported by multi-agent systems and edge computing (InEDGEMobility): Towards Sustainable Intelligent Mobility: Blockchain-based framework for IoT Security", Reference: RTI2018-095390-BC32, financed by the Spanish Ministry of Science, Innovation and Universities (MCIU), the State Research Agency (AEI) and the European Regional Development Fund (FEDER).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Due to the restricted policies of Twitter data publication, these data are confidential. The steps of data extraction from this platform using the official Twitter API have been clearly explained.

Acknowledgments: This research work has been carried out thanks to the project "Intelligent and sustainable mobility supported by multi-agent systems and edge computing (InEDGEMobility): Towards Sustainable Intelligent Mobility: Blockchain-based framework for IoT Security", RTI2018-095390-BC32 Spanish Ministry of Science, Innovation and Universities (MCIU), the State Research Agency (AEI) and the European Regional Development Fund (FEDER).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Allman-Farinelli, M.; Nour, M. Exploring the role of social support and social media for lifestyle interventions to prevent weight gain with young adults: Focus group findings. *J. Hum. Nutr. Diet.* **2021**, *34*, 178–187. [CrossRef] [PubMed]
2. Thelwall, M. Word association thematic analysis: A social media text exploration strategy. *Synth. Lect. Inf. Concepts Retr. Serv.* **2021**, *13*, 1–111. [CrossRef]
3. Osorio-Arjona, J.; Horak, J.; Svoboda, R.; García-Ruíz, Y. Social media semantic perceptions on Madrid Metro system: Using Twitter data to link complaints to space. *Sustain. Cities Soc.* **2021**, *64*, 102530. [CrossRef]
4. Alamsyah, A.; Rahardjo, B.; Kuspriyanto. Social network analysis taxonomy based on graph representation. *arXiv* **2021**, arXiv:2102.08888.
5. Li, Z.; Zhang, Q.; Du, X.; Ma, Y.; Wang, S. Social media rumor refutation effectiveness: Evaluation, modelling and enhancement. *Inf. Process. Manag.* **2021**, *58*, 102420. [CrossRef]
6. Choudhary, A.; Arora, A. Linguistic feature based learning model for fake news detection and classification. *Expert Syst. Appl.* **2021**, *169*, 114171. [CrossRef]
7. Derhab, A.; Alawwad, R.; Dehwah, K.; Tariq, N.; Khan, F.A.; Al-Muhtadi, J. Tweet-based Bot Detection using Big Data Analytics. *IEEE Access* **2021**, *9*, 65988–66005. [CrossRef]
8. Ayo, F.E.; Folorunso, O.; Ibharalu, F.T.; Osinuga, I.A.; Abayomi-Alli, A. A probabilistic clustering model for hate speech classification in twitter. *Expert Syst. Appl.* **2021**, *173*, 114762. [CrossRef]
9. Albalawi, R.; Yeap, T.H.; Benyoucef, M. Using topic modeling methods for short-text data: A comparative analysis. *Front. Artif. Intell.* **2020**, *3*, 42. [CrossRef]
10. Dhiman, A.; Toshniwal, D. An Approximate Model for Event Detection From Twitter Data. *IEEE Access* **2020**, *8*, 122168–122184. [CrossRef]
11. Wu, W.; Chow, K.P.; Mai, Y.; Zhang, J. Public Opinion Monitoring for Proactive Crime Detection Using Named Entity Recognition. In Proceedings of the IFIP International Conference on Digital Forensics, New Delhi, India, 6–8 January 2020; pp. 203–214.
12. Shoeibi, N.; Shoeibi, N.; Hernández, G.; Chamoso, P.; Corchado, J.M. AI-Crime Hunter: An AI Mixture of Experts for Crime Discovery on Twitter. *Electronics* **2021**, *10*, 3081. [CrossRef]
13. Martyniuk, H.; Kozlovskiy, V.; Lazarenko, S.; Balanyuk, Y. Data Mining Technics and Cyber Hygiene Behaviors in Social Media. *South Fla. J. Dev.* **2021**, *2*, 2503–2515. [CrossRef]
14. Sushama, C.; Kumar, M.S.; Neelima, P. Privacy and security issues in the future: A social media. *Mater. Today Proc.* **2021**. [CrossRef]
15. Marmo, R. Social media mining. In *Encyclopedia of Organizational Knowledge, Administration, and Technology*; IGI Global: Hershey, PA, USA, 2021; pp. 2153–2165.

16. Luo, Y. Using tweets to understand how COVID-19–Related health beliefs are affected in the age of social media: Twitter data analysis study. *J. Med. Internet Res.* **2021**, *23*, e26302.
17. Ge, L.; Chen, S. Exact Dynamic Time Warping calculation for weak sparse time series. *Appl. Soft Comput.* **2020**, *96*, 106631. [CrossRef]
18. Sanh, V.; Debut, L.; Chaumond, J.; Wolf, T. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. *arXiv* **2019**, arXiv:1910.01108.
19. Roberts, A.; Raffel, C.; Shazeer, N. How Much Knowledge Can You Pack Into the Parameters of a Language Model? *arXiv* **2020**, arXiv:2002.08910.
20. Xiao, J.; Zhou, Z. Research Progress of RNN Language Model. In Proceedings of the 2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), Dalian, China, 27–29 June 2020; pp. 1285–1288.
21. Zhao, J.; Huang, F.; Lv, J.; Duan, Y.; Qin, Z.; Li, G.; Tian, G. Do rnn and lstm have long memory? In Proceedings of the International Conference on Machine Learning, PMLR, Virtual, 13–18 July 2020; pp. 11365–11375.
22. Akhtar, N.; Ahamad, M.V. Graph tools for social network analysis. In *Research Anthology on Digital Transformation, Organizational Change, and the Impact of Remote Work*; IGI Global: Hershey, PA, USA, 2021; pp. 485–500.
23. Chatterjee, M.; Sowmya, P. Detection of Fake and Cloned Profiles in Online Social Networks. In Proceedings of the Proceedings 2019: Conference on Technologies for Future Cities (CTFC), Maharashtra, India, 8–9 January 2019.
24. Choumane, A.; Yassin, F. Characterizing and Detecting Similar Twitter Users. In Proceedings of the 2021 3rd IEEE Middle East and North Africa COMMunications Conference (MENACOMM), Virtual, 3–5 December 2021; pp. 25–30.
25. Kim, M.; Kim, D. A Suggestion on the LDA-Based Topic Modeling Technique Based on ElasticSearch for Indexing Academic Research Results. *Appl. Sci.* **2022**, *12*, 3118. [CrossRef]
26. Vajjhala, N.R.; Rakshit, S.; Oshogbunu, M.; Salisu, S. Novel user preference recommender system based on Twitter profile analysis. In *Soft Computing Techniques and Applications*; Springer: Singapore, 2021; pp. 85–93.
27. Dahiya, S.; Kumar, G.; Yadav, A. A Contextual Framework to Find Similarity Between Users on Twitter. In *Proceedings of the Second Doctoral Symposium on Computational Intelligence*; Springer: Singapore, 2022; pp. 793–805.
28. Chen, G.; Shi, X.; Chen, M.; Zhou, L. Text similarity semantic calculation based on deep reinforcement learning. *Int. J. Secur. Netw.* **2020**, *15*, 59–66. [CrossRef]
29. Chandrasekaran, D.; Mago, V. Evolution of Semantic Similarity—A Survey. *arXiv* **2020**, arXiv:2004.13820.
30. Park, K.; Hong, J.S.; Kim, W. A methodology combining cosine similarity with classifier for text classification. *Appl. Artif. Intell.* **2020**, *34*, 396–411. [CrossRef]
31. Sowmya, P.; Chatterjee, M. Detection of Fake and Clone accounts in Twitter using Classification and Distance Measure Algorithms. In Proceedings of the 2020 International Conference on Communication and Signal Processing (ICCSP), Chennai, India, 28–30 July 2020; pp. 67–70.
32. Punkamol, D.; Marukatat, R. Detection of Account Cloning in Online Social Networks. In Proceedings of the 2020 8th International Electrical Engineering Congress (iEECON), Chiangmai, Thailand, 4–6 March 2020; pp. 1–4.
33. Guven, Z.A.; Unalir, M.O. Natural language based analysis of SQuAD: An analytical approach for BERT. *Expert Syst. Appl.* **2022**, *195*, 116592. [CrossRef]
34. Peinelt, N.; Nguyen, D.; Liakata, M. tBERT: Topic models and BERT joining forces for semantic similarity detection. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Virtual, 5–10 July 2020; pp. 7047–7055.
35. Dogra, V.; Singh, A.; Verma, S.; Kavita; Jhanjhi, N.; Talib, M. Analyzing DistilBERT for Sentiment Classification of Banking Financial News. In *Intelligent Computing and Innovation on Data Science*; Springer: Singapore, 2021; pp. 501–510.
36. Dogra, V.; Verma, S.; Singh, A.; Kavita; Talib, N.; Humayun, M. Banking news-events representation and classification with a novel hybrid model using DistilBERT and rule-based features. *Turk. J. Comput. Math. Educ.* **2021**, *12*, 3039–3054.
37. Vogel, I.; Meghana, M. Profiling Hate Speech Spreaders on Twitter: SVM vs. Bi-LSTM. In Proceedings of the CLEF, Bucharest, Romania, 21–24 September 2021.
38. Haustein, S. Scholarly twitter metrics. In *Springer Handbook of Science and Technology Indicators*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 729–760.
39. Twitter API Documentation | Docs | Twitter Developer. Available online: https://developer.twitter.com/en/docs/twitter-api (accessed on 1 April 2022)
40. Rate Limits | Docs | Twitter Developer. Available online: https://developer.twitter.com/en/docs/twitter-api/v1/rate-limits (accessed on 1 April 2022).
41. Herbert, C.; El Bolock, A.; Abdennadher, S. How do you feel during the COVID-19 pandemic? A survey using psychological and linguistic self-report measures, and machine learning to investigate mental health, subjective experience, personality, and behaviour during the COVID-19 pandemic among university students. *BMC Psychol.* **2021**, *9*, 1–23.
42. Lahreche, A.; Boucheham, B. A fast and accurate similarity measure for long time series classification based on local extrema and dynamic time warping. *Expert Syst. Appl.* **2021**, *168*, 114374. [CrossRef]
43. Berndt, D.J.; Clifford, J. Using dynamic time warping to find patterns in time series. In Proceedings of the KDD Workshop, Seattle, WA, USA, 31 July 1994; Volume 10, pp. 359–370.
44. Gosliga, J.; Gardner, P.; Bull, L.; Dervilis, N.; Worden, K. Foundations of Population-based SHM, Part II: Heterogeneous populations–Graphs, networks, and communities. *Mech. Syst. Signal Process.* **2021**, *148*, 107144. [CrossRef]

45. Vollmer, S. Google Translate. In *Figures of Interpretation*; Multilingual Matters: Bristol, UK, 2021; pp. 72–75.
46. Wang, C.; Li, M.; Smola, A.J. Language models with Transformers. *arXiv* **2019**, arXiv:1904.09408.
47. Shaikh, S.; Daudpota, S.M.; Imran, A.S.; Kastrati, Z. Towards Improved Classification Accuracy on Highly Imbalanced Text Dataset Using Deep Neural Language Models. *Appl. Sci.* **2021**, *11*, 869. [CrossRef]
48. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in neural information processing systems, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
49. Xiong, R.; Yang, Y.; He, D.; Zheng, K.; Zheng, S.; Xing, C.; Zhang, H.; Lan, Y.; Wang, L.; Liu, T. On Layer Normalization in the Transformer Architecture. In Proceedings of the 37th International Conference on Machine Learning, Virtual, 13–18 July 2020; Volume 119, pp. 10524–10533.
50. Nozza, D.; Bianchi, F.; Hovy, D. What the [mask]? making sense of language-specific BERT models. *arXiv* **2020**, arXiv:2003.02912.
51. Le, N.Q.K.; Ho, Q.T.; Nguyen, T.T.D.; Ou, Y.Y. A Transformer architecture based on BERT and 2D convolutional neural network to identify DNA enhancers from sequence information. *Briefi. Bioinform.* **2021**, *22*, bbab005. [CrossRef] [PubMed]
52. Subba, B.; Gupta, P. A tfidfvectorizer and singular value decomposition based host intrusion detection system framework for detecting anomalous system processes. *Comput. Secur.* **2021**, *100*, 102084. [CrossRef]
53. Qiu, Y.; Yang, B. Research on Micro-blog Text Presentation Model Based on Word2vec and TF-IDF. In Proceedings of the 2021 IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC), Dalian, China, 14–16 April 2021; pp. 47–51.
54. Aljuaid, H.; Iftikhar, R.; Ahmad, S.; Asif, M.; Afzal, M.T. Important citation identification using sentiment analysis of In-text citations. *Telemat. Inform.* **2021**, *56*, 101492. [CrossRef]
55. Johansson, F.; Kaati, L.; Shrestha, A. Detecting multiple aliases in social media. In Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2013), Niagara, ON, Canada, 25–28 August 2013; pp. 1004–1011.
56. Goel, A.; Sharma, A.; Wang, D.; Yin, Z. Discovering similar users on twitter. In Proceedings of the 11th Workshop on Mining and Learning with Graphs, Chicago, IL, USA, 11 August 2013.
57. Van der Maaten, L.; Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.
58. Agrawal, T. Hyperparameter Optimization Using Scikit-Learn. In *Hyperparameter Optimization in Machine Learning*; Apress: Berkeley, CA, USA, 2021; pp. 31–51.
59. Yang, L.; Shami, A. On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing* **2020**, *415*, 295–316. [CrossRef]
60. Corchado, J.M.; Chamoso, P.; Hernández, G.; Gutierrez, A.S.R.; Camacho, A.R.; González-Briones, A.; Pinto-Santos, F.; Goyenechea, E.; Garcia-Retuerta, D.; Alonso-Miguel, M.; et al. Deepint. net: A Rapid Deployment Platform for Smart Territories. *Sensors* **2021**, *21*, 236. [CrossRef]