IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 14, 2021, accepted March 14, 2021, date of publication March 24, 2021, date of current version April 14, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3068659

A Deep Learning Approach for Robust Detection of Bots in Twitter Using Transformers

DAVID MARTÍN-GUTIÉRREZ[1], (Member, IEEE), GUSTAVO HERNÁNDEZ-PEÑALOZA[1], (Member, IEEE), ALBERTO BELMONTE HERNÁNDEZ[1], (Member, IEEE), ALICIA LOZANO-DIEZ[2], AND FEDERICO ÁLVAREZ[1], (Member, IEEE)

[1]Visual Telecommunication Applications Group, Signals, Systems and Radio Communications (SSR) Department, Universidad Politécnica de Madrid, 28040 Madrid, Spain
[2]AUDIAS–Audio Data Intelligence and Speech, Universidad Autónoma de Madrid, 28049 Madrid, Spain

Corresponding author: David Martín-Gutiérrez (dmz@gatv.ssr.upm.es)

This work was supported by the H2020 European Project: FAke News discovery and propagation from big Data ANalysis and artificial intelliGence Operations (FANDANGO) under Grant 780355.

ABSTRACT During the last decades, the volume of multimedia content posted in social networks has grown exponentially and such information is immediately propagated and consumed by a significant number of users. In this scenario, the disruption of fake news providers and bot accounts for spreading propaganda information as well as sensitive content throughout the network has fostered applied research to automatically measure the reliability of social networks accounts via Artificial Intelligence (AI). In this paper, we present a multilingual approach for addressing the bot identification task in Twitter via Deep learning (DL) approaches to support end-users when checking the credibility of a certain Twitter account. To do so, several experiments were conducted using state-of-the-art Multilingual Language Models to generate an encoding of the text-based features of the user account that are later on concatenated with the rest of the metadata to build a potential input vector on top of a Dense Network denoted as *Bot-DenseNet*. Consequently, this paper assesses the language constraint from previous studies where the encoding of the user account only considered either the metadata information or the metadata information together with some basic semantic text features. Moreover, the *Bot-DenseNet* produces a low-dimensional representation of the user account which can be used for any application within the Information Retrieval (IR) framework.

INDEX TERMS Artificial intelligence, bot detector, deep learning, feature representation, language models, misinformation detection, social media mining, transfer learning, transformers.

I. INTRODUCTION & MOTIVATION

In recent years, social media platforms such as Twitter or Facebook have gained a large level of both popularity and influence among millions of users due to the benefits of publishing, propagating and exchanging large volumes of multimedia content along the network. Therefore, these platforms allow users to establish a digital community as remarked in [22], which has made possible not only to discover and embrace new relationships but to maintain and boost existing ones.

On the other hand, due to both the great influence these platforms have on the lifestyle of people and its evolving as a potential communication tool, they have exponentially promoted its attraction for marketing and commercial purposes by analysing the behaviour and opinion of users in different topics or events such as political elections. Consequently, numerous research studies have been fostered in the social media field with different purposes including sentiment analysis [35], traffic control [48], or consumer behaviour mining [4].

However, the considerable growth of social media platforms has also provoked the desire of altering people's opinion in certain topics by spreading propaganda or bias information. Many of these controlling procedures are carried out by Bots which are widely described in numerous investigations [31], [32], [40] such as automatic systems which are capable of generating and spreading multimedia content

The associate editor coordinating the review of this manuscript and approving it for publication was Weiping Ding.

throughout the network without the supervision of a human being.

Furthermore, with the disruptive growth of Artificial Intelligence (AI) algorithms, the identification of bots or non-reliable sources has become a crucial challenge to be investigated. It raised many studies and publications with the goal of building robust automatic systems to improve the quality of experience of consumers in such platforms by reducing their privacy risks as well as increasing the trustworthiness on the platform itself at the same time.

Therefore, this paper aims to contribute to the state-of-the-art in this field by proposing a novel method for automatically

(i) encoding an input user account as a low-dimensional feature vector independently of its language,
(ii) identifying the input encoding vector as a suspicious bot account with a certain probability throughout a Deep Neural Network (DNN) referred as *Bot-DenseNet*.
(iii) generating a low-dimensional embedding which represents the original input encoding vector of the user account and which can be used for any other purpose related to Information Retrieval (IR).

More specifically, our study is focused on identifying Bot accounts in Twitter by considering three main aspects of the account: its activity level, popularity and profile information. The global set of features can be separated into two modalities including metadata and text-based descriptors, where the latter set is encoded via novel Language Model Embeddings (LME) to mitigate the language constraint suffered in previous studies.

The remainder of this paper is organized as follows: in Section II, previous studies and investigations within the Bot identification framework are described. Section III outlines the main components involved in the proposed multilingual approach including Section III-A to describe the generation of the input encoding vector and Section III-B which summarizes the proposed architecture of the proposed *Bot-DenseNet* model. Afterwards, Section IV presents the distinct experiments conducted as well as the outcomes and breakthroughs reached. Finally, the general conclusions and future work are summarized in Section V.

II. RELATED WORK

Recently, AI techniques including Deep Learning (DL) and Machine Learning (ML) methods have gained popularity and interest in many applied research and industry services related to social media analysis where, sentiment analysis and text classification have been the central focus of these investigations specially for searching engines or recommender systems.

More specifically, the essence of sentiment analysis consists in extracting an aspect term of an input sentence to determine its polarity as positive, neutral and negative as authors remark in [8] and it is generally solved as a multi-class classification problem. Moreover, sentiment analysis has widely been used in numerous studies for both reviews and user opinions analysis in online commercial platforms [16], [47] and user behaviour mining in social media platforms such as Twitter [6], [29], [37], [42].

Furthermore, the continuous growth of the social media platforms such as Twitter or Facebook in the last decade along with the considerable propagation of non-trusted information throughout them, have raised applied research to automatically identify these non-trusted sources which in many cases correspond to non-human or Bot accounts. One of the first studies in this field was proposed by [9] and it was based on a random forest approach to classify bots and non-bots accounts using a manually annotated dataset with around 2000 samples. In 2016, *BotOrNot* was proposed in [12] as a service to automatically detect bots in Twitter using similarities between characteristics of social bots. This model has inspired posterior investigations in this field that even employed this service to automatically annotate data from Twitter.

In [11], authors annotated more than 8000 accounts and proposed a classifier which achieved a considerable level of accuracy for such set of samples. Additionally, [38] presented a model for Twitter bot detection based on a large number of metadata from the account to perform the classification.

More recently, several scientific studies have incorporated more annotated samples to support this research such as [27], [44], [45] including some procedures for achieving better level of accuracy by strategically selecting a subset of training samples that better generalize the problem. In [22], a language-agnostic approach is employed to identify potential features to distinguish between human and bot accounts. The model is then trained and validated using over 8000 samples distributed in an unbalanced fashion and its performance reaches an accuracy of 98%.

Moreover, authors in [32] proposed a 2D Convolutional Network model based on user-generated contents for detecting bots from human accounts including its gender (male, female account) covering both Spanish and English languages. A similar goal is explored by authors in [40], where both Word and Character N-Grams are employed as main features to perform the classification.

A different manner of addressing the problem was recently proposed by authors in [5], where novel *altmetrics* data to investigate social networks are analysed and they are used to train a Graph Convolutional Network (GCN) which reaches over 70% of accuracy in this task. On the other hand, authors in [34] presented a novel one-class classifier to enhance Twitter bot detection without any requiring previous information about them.

Most of the aforementioned approaches were limited due to lack of large volumes of annotated data for this specific task by the time their experiments were conducted. This problem is also remarked in [45] and thus, this paper has considered all the available public datasets in the current days in order to build the system with the most updated, newest and relevant state-of-the-art annotated data from Twitter. Additionally, although many approaches employ both metadata

and text-based features from the user accounts, the text-based features are either extracted at a lexical level or they only cover a limited number of languages such as Spanish or English.

Unlike previous studies, our proposed model encodes all the text-based features of an input user account via novel multilingual Language Models (LM) including transformer models such as the so-called BERT [14] or Contextual string embeddings proposed in [2]. Thus, by concatenating both the metadata set of features along with the output vector provided by these LMs, an input vector of the user account is obtained. Finally, this paper proposes a Dense-based DL model to produce both the final decision of the account and a low-dimensional embedding of the user based on the aforementioned input vector.

III. A MULTILINGUAL APPROACH FOR USER ACCOUNT ENCODING VIA TRANSFORMERS

As it was introduced in previous sections, our system consists in a multilingual approach capable of better identify suspicious Twitter accounts based on a set of features independently of the language of the account. More specifically, the methodology of building the whole system can be distributed in two separate processes: (i) a preprocessing stage where a multilingual input vector of the user account is generated and (ii) a final decision system for identifying whether the account has a normal or abnormal behaviour according to existence patterns in the input vector generated during the first stage.

Moreover, the former process is responsible for retrieving a large collection of annotated Twitter accounts in a binary fashion, where the positive class indicates that the account is Bot whereas the negative class means that it belongs to the human account category. Subsequently, several features were collected from each Twitter account to enhance some relevant aspects including: (i) *Level of activity*, (ii) *Level of popularity*, (iii) *Profile information*.

Finally, this first stage ends by combining all the features to generate an input vector with both textual and metadata information for each Twitter account. Section III-A describes the full process to provide all the details of the implementation.

The latter process, described in Section III-B, is in charged of automatically identifying patterns in the input encoding vector to properly distinguish between bots and human Twitter accounts via Deep Neural Networks (DNNs). Moreover, this process automatically obtains a low-dimensional feature representation of the input vector which can be used for any IR purpose in a more efficient way due to its low-dimensional nature.

A. MULTILINGUAL USER ENCODING VIA TRANSFORMERS

As introduced in Section III, a first process is needed to combine distinct relevant aspects from Twitter accounts to build a solid multilingual encoding representation which can be used as potential inputs for classification purposes throughout DNNs.

In Figure 1, an illustrative block diagram of this process is presented to show the different tools and stages needed to achieve the objectives of this first phase of the system.

More specifically, a first stage to retrieve all the data from the set of users U is performed via Twitter API.[1] Subsequently, each account is represented as a vector considering two modalities: text-based and metadata features. The former set is passed through a multilingual pre-trained LM model to obtain a feature vector representation of the text information including the description, the username of the account as well as the language of the account. The latter set of features is directly passed forward to the final stage which is a concatenation of both modalities into a single feature vector x, which encodes the information of an input user account.

1) DATASET GENERATION

There are several public datasets to address the bot identification problem from a binary classification perspective as the ones presented in Table 1. Moreover, some of these datasets were already used to train and evaluate the so-called *Botometer* (formerly *BotOrNot*) service proposed by [12].

However, as authors described in [7], [28], the generation of bot accounts continuously changes over time and additionally, some of the provided accounts have been already suspended by Twitter. Thus, a preprocessing is needed to improve the usability of this large collection of datasets by removing the identifiers those accounts that were already removed by Twitter. This aspect is also critical since several previous Bot detectors have not been updated with the new tendencies and features that bots accounts may currently have, so that, they are no longer as reliable as they used to be.

On the other hand, due to policy restriction terms from Twitter, the datasets only contain the identifier of the Twitter account but not any significant feature. Consequently, an additional data crawling process via the Twitter API is performed to gather further information about the available accounts which is needed for the analysis. As introduced in Section III, the following information is collected:

- *Popularity features* including total number of both friends and followers,
- *Activity features* including the following fields: creation date, average tweets per day, tweets & favourites counts, account age.
- *Profile information features* including: screen name, description, language, location, verified indicator, default profile indicator.

After crawling and preprocessing the data, the final complete dataset is composed of **37438** Twitter accounts, where **25013** were annotated as human accounts and the remaining **12425** are bots.

2) INPUT USER ENCODING GENERATION

The crucial part of this first stage lies in the generation of an user encoding vector based on the aforementioned collection

[1] https://developer.twitter.com/en/docs/twitter-api

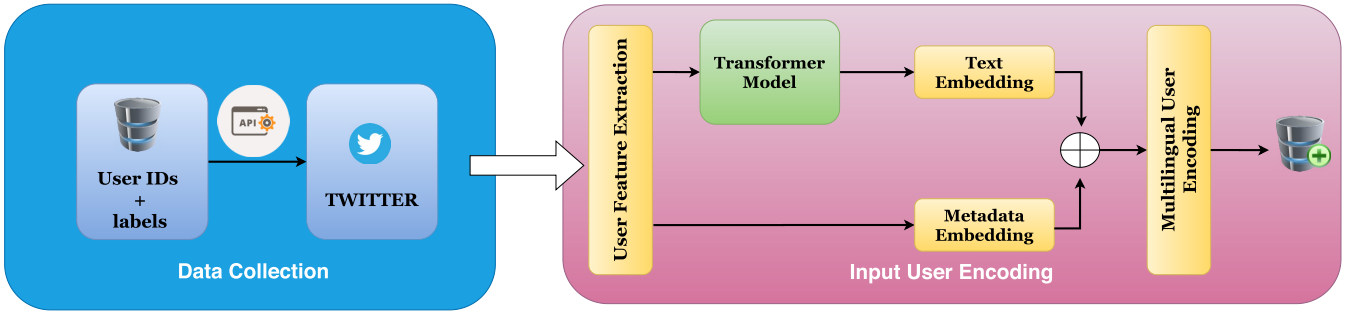


FIGURE 1. A block diagram regarding the process of extracting multilingual encoding representations of user accounts in Twitter.

TABLE 1. List of datasets employed when conducting the experiments of this study.

Dataset	Description	Paper
verified-2019	Verified human accounts Labels and user objects.	[45]
botwiki-2019	Self-identified bots from https://botwiki.org. Labels and user objects.	[45]
midterm-2018	Manually labeled human and bot accounts from 2018 US midterm elections. Labels and user objects.	[45]
cresci-stock-2018	Automated accounts that act in coordinate fashion.Labels and user objects.	[45]
cresci-rtbust-2019	Manually annotated bot and human accounts. Labels and user objects.	[27]
political-bots-2019	Automated political accounts. Labels and user objects.	[44]
botometer-feedback-2019	Botometer feedback accounts manually labeled by K.C. Yang. Labels and user objects.	[44]
vendor-purchased-2019	Fake follower accounts purchased from several companies. Labels and user objects.	[44]
celebrity-2019	Celebrity accounts collected as authentic users. Labels and user objects.	[44]
pronbots-2019	Pronbots shared by Andy Patel (github.com/r0zetta/pronbot2). Labels and user objects.	[44]
gilani-2017	Manually annotated human and bot accounts. Labels and user objects.	[15]
varol-2017	This dataset contains annotation of 2573 Twitter accounts. Annotation and data crawl is completed in April 2016.	[38]
cresci-2017	A dataset of (i) genuine, (ii) traditional, and (iii) social spambot Twitter accounts, annotated by CrowdFlower contributors.	[11]

of features to serve as input of the proposed deep learning model. In previous related studies [7], [12], [13], [20], the proposed solutions had two main constraints: 1) either they were metadata-oriented approaches so that, the text-based features were extracted at a semantic-level in terms of Natural Language Processing, or 2) they employed more advanced NLP procedures based on N-grams or DL solutions but they only supported a limited number of languages when performing the analysis.

However, our proposal addresses the aforementioned constraints by combining relevant metadata features along with powerful models capable of transforming text-based features into vectors independently of the language of the input text.

More specifically, given an input set of Users $U = \{u_1, u_2, \ldots, u_m\}$, a certain user account is represented as $u_i = [u_i^t, u_i^z] \; \forall i = 1, \ldots, m$ where u_i^t indicates its text-based feature vector whereas u_i^z represents the remaining metadata-based vector. Our proposed solution employs a mapping function $f(u)$ to generate a new set of Users $\dot{U} = \{\dot{u}_1, \dot{u}_2, \ldots, \dot{u}_m\}$, where $\dot{u}_i = f(u_i) = g(u_i^t) \,||\, h(u_i^z)$. In this case, we denote $||$ to indicate a concatenation operation between this pair of vectors. The reason behind using a concatenation layer at the end of this process lies in the fact that the system only consider information coming from the same target object: the user account. Other alternatives widely used in Collaborative Recommender Systems such as computing the outer product [19] were not consider for this approach since in those scenarios, the information from the embeddings comes from two different sources: Users and Items, and the goal of the outer product is thus, to catch similarities and discrepancies between this two sets.

3) ENCODING TEXT-BASED FEATURES

Regarding the generation of the text-based vector from an input user account u_i, a certain function $g(u)$ is needed. Different state-of-the-art sentence-level encoders from several NLP frameworks were explored and investigated. In particular, the so-called Flair framework described in [2] was employed to combine state-of-the-art Word Embeddings (WE) and Transformers [39], [43] for extracting robust document embeddings from the text-based features. More precisely, the following main families of embeddings have been employed in this study:

(i) Contextual string embeddings [3] which are trained without any explicit notion of words and therefore, words are modelled as sequences of characters. Moreover, words are contextualized by the surrounding text. The employed model was trained using the so-called JW300 Dataset described in [1]. In this study, both multi-forward and multi-backward embeddings are used. The dimension of their outputs is equal to 2048.
(ii) BERT (Bidirectional Encoder Representations from Transformers) embeddings which were proposed and developed by [14] and are based on a bidirectional

transformer architecture [39], [43]. In this study, the so-called *bert-base-multilingual-cased* has been employed.

(iii) RoBERTa which is an adaptive version of the BERT embedding where the goal is to improve the performance in longer sequences, or when there are vast volumes of data as [41] suggests. In this case, we employed the so-called *roberta-large-mnli* pre-trained model.

Furthermore, several experiments were conducted considering three different solutions. The first approach is based on using one or multiple stacked embeddings similarly to the approach proposed by [25] so that, all the text-based features from the user account are encoded at a sentence-level representation. Subsequently, a document-level representation is computed via a Pooling model, where an average of all the stacked sentence-level embeddings was calculated.

The second approach regards the training of a Long Short-Term Memory (LSTM) recurrent network over all the word embeddings used to generate the sentence-level encoding. Finally, the last approach directly uses an intermediate layer from a pre-trained Transformer model, to produce the document-level embedding.

Moreover, both the multilingual BERT transformer pre-trained model named as *BERT-base-multilingual-cased* and the RoBERTa pre-trained model named as *roberta-large-mnli* have been employed. More specifically, the former generates a 768-dimensional embedded vector whereas the latter produces a 1024 embedded representation. All the details of the aforementioned pre-trained models can be found at the official Hugging Face repository. [2]

4) ENCODING METADATA-BASED FEATURES

On the other hand, all the corresponding metadata features from an input user account are properly preprocessed and encoded throughout function $h(u_i^z)$ to be interpreted by neural networks. In addition, they are concatenated along with the aforementioned text-based features as it is remarked in Section III-A.

In particular, this set of features includes all the information related to both the popularity and the activity of the user account.

B. BOT-DenseNet

Once the set of input user vectors denoted by $\dot{U}$ is obtained, a second process is required to automatically identify an account as bot or human.

To address this goal, we propose a Deep Fully-connected-based neural network named as *Bot-DenseNet*, which is capable of finding robust decision boundaries based on hidden patterns of the input vectors to better recognize bot accounts in Twitter. The details of the model implementation including classical parameters in neural networks such as activation functions, number of neurons in the hidden layers or the selected optimizers to perform the backpropagation and the Gradient Descent algorithm are summarized in Table 2. One of the main differences from previous studies is the incorporation of the so-called *Scaled Exponential Linear Unit* (SELU) similarly to the approach proposed at [23] which obtains better results than the classical ReLU activation function.

TABLE 2. List of parameters involved in the design of the *Bot-DenseNet* model. # indicates the total number.

Parameter name	*Parameter value*
Loss function	Binary Cross Entropy
Optimizer	Adam
Learning rate	0.001
Batch size	256
# Hidden layers	2
# Neurons last hidden layer	256
Intermediate activation function	SELU
Output activation function	Sigmoid

On the other hand, the architecture of the system, presented in Figure 2, is composed of a set of blocks including Dense + Batch Normalization + Activation + Dropout layers in a sequential fashion as usual in general dense-based models.

Furthermore, the inclusion of a Batch Normalization layer within the architecture lies in previous studies such as the one described in [10], [21], where authors proved that the training performance indeed improves when using such normalization since this layer provides many benefits including a faster convergence as it allows to employ higher learning rates during the Gradient descent algorithm.

The aforementioned hyper-parameters were carefully selected after conducting several heuristic experiments in order to design a model with the best performance in terms of F1-score.

IV. EXPERIMENTAL RESULTS

One of the main objectives of this paper relies on analysing, via an ablation study, different input feature vectors based on Transformers as well as additional novel approaches to investigate the performance of the same DNN model.

Moreover, The DL architecture was trained in the classical supervised learning fashion considering a binary classification where the Positive class refers to Bots whereas the Negative corresponds to Human accounts.

Due to the unbalanced constraint of the dataset, two main steps have been followed: (i) the so-called Stratify sampling updated in [26] has been employed to force having samples from both classes during the training and evaluation of the system, (ii) the F1-score metric as the main measurement of the performance of the system since it balances both precision and recall metrics in a single value and provides more realistic information about the capability of the model to detect both Positive and Negative classes than the classical accuracy metric.

Moreover, to mitigate the classical *overfitting* problem, which happens when the neural network is not capable of generalizing properly for unseen samples, two main widely used techniques have been incorporated: (i) *Dropout* firstly proposed in [36], which pursues the aim of deactivating some

[2]https://huggingface.co/transformers/

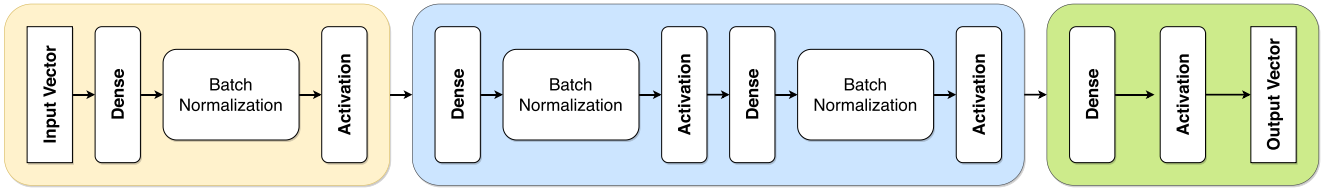


FIGURE 2. A block diagram representation of the proposed architecture *Bot-DenseNet*, where the yellow block indicates the input layer, the blue one regards the hidden layers and finally, the green one indicates the output layer.

TABLE 3. Outcomes of the experiments conducted to train our *Bot-DenseNet* using different input feature vectors which were generated as the concatenation of both a metadata feature vector and a text embedding.

Input feature vector	Configuration metrics					
	F1-Score Training			F1-Score Validation		
	Pooling	LSTM	Transformer	Pooling	LSTM	Transformer
Flair + metadata	0.84	0.75	-	0.73	0.73	-
BERT + metadata	0.88	0.85	0.74	0.72	0.71	0.72
Flair + BERT + metadata	0.85	0.84	-	0.72	0.73	-
RoBERTa + metadata	-	-	0.79	-	-	0.77

neurons randomly at each epoch with a certain probability; (ii) *Early Stopping*, fully described in [46], which attempts to stop the training process whenever the performance on the validation set has no longer improved in a certain number of epochs. Therefore, a hyper-parameter is required to indicate the number of consecutive epochs with no improvements in the loss function or in the metrics considering the validation set, and it is usually named as patience.

A. TRAINING & VALIDATING BOT-DenseNet

The aim of these experiments is to find the most appropriate text embedding to be added on top of the *Bot-DenseNet* along with the remaining metadata feature vector in order to find optimal decision boundaries for downstream tasks such as the one presented in this paper.

The outcomes obtained during the training and validation stages for all the possible input feature vectors are summarized in Table 3. Since the dataset is not balanced, the F1-score metric has a crucial role in the evaluation of the system in order to objective measure the performance of identifying bots in a social network such as Twitter where only a few accounts from the total set of accounts belong indeed to the bot category as it is described in previous studies [7], [27], [34]. In particular, since the model is trained using an Early-Stopping callback to stop the process in the epoch when the loss function in the validation set is no longer decreasing, the F1-score was computed by using both the recall and the precision at this specific epoch.

Moreover, in Table 3, each row indicates the pre-trained LM employed such as Flair, BERT, RoBERTa etc., as well as the approach followed to generate the final input user encoding (via Pooling, bidirectional LSTM or directly the embedding obtained from an intermediate layer of the Transformer model). In particular, the results presented in Table 3 reflect that when combining the text embeddings directly extracted from intermediate layers of the Transformers along with the metadata features, the proposed model *Bot-DenseNet* achieves higher scores in terms of F1-score in both training and validation sets. On the other hand, when using either Pooling or LSTMs to produce the final text embeddings, the F1-score metric in the training phase is higher than in the validation phase which indicates that the model is suffering from overfitting. As a result, this issue may provoke a decrease in the performance of the system when analysing unseen observations in future predictions.

B. SELF-SUPERVISED USER EMBEDDING

Our proposed model is capable of identifying suspicious Twitter accounts based on a robust set of input features. However, as it is well-known in the Deep Learning framework, intermediate layers are usually an adequate embedded representation of the inputs which can be employed in other downstream tasks such as text classification or similarity analysis in a more efficient way due to their low-dimensional nature. Thus, after training our proposed model, it produces a relevant representation of an input Twitter account in a self-supervised fashion since such representation was automatically learned by the intermediate hidden layers. Hence, *Bot-DenseNet* can also be used to encode any user account as a 256-dimensional vector throughout its last intermediate layer. To better visualize the final embeddings obtained by the last layers of the different models, a 2D projected representation of them were computed using the so-called T-SNE algorithm, updated in [30] with a level of perplexity equal to 80.

More specifically, Figure 3 includes the embeddings as a result of training the model using directly the pre-trained Transformers models whereas Figure 4 shows the outcomes when using different combinations of Word Embeddings along with either Pooling or LSTMs on top of them to produce the final text vector.

TABLE 4. An ablation study by comparing the complexity of the model in terms of both the feature vector length, the total number of parameters to be trained as well as the F1-score achieved during the validation phase.

Input feature vector	*Input vector length*	*Trainable Parameters.*	*F1-Score*
Flair + Pooling metadata	4107	21,351,581	0.73
Flair + LSTM + metadata	267	106,781	0.73
BERT + Pooling + metadata	3083	11,226,269	0.72
BERT + LSTM + metadata	267	106,781	0.71
BERT + metadata + Transformer	779	743,069	0.72
Flair + BERT + Pooling + metadata	7179	59,194,037	0.72
Flair + BERT + LSTM + metadata	267	106,781	0.73
RoBERTa + metadata	**779**	**743,069**	**0.77**

Moreover, Figure 3 remarks the potential of Transformers in any downstream NLP tasks such the one presented in this paper where their intermediate layers have been employed to obtain robust representations of the text-based features of the user accounts. As a consequence, they have increased the capability of *Bot-DenseNet* to better distinguish between bots and human accounts in a more efficient manner.

Furthermore, Figure 3 shows that the boundaries determined by both BERT and RoBERTa transformers without any additional step to generate a text encoding (neither Pooling nor LSTMs) are simpler and more adequate than the ones provided by combining different text embeddings which is summarized in Figure 4. This fact is also presented in the F1-score metrics from Table 3, where it is clear that these two solutions are the ones which are not overfitting the data and thus, they are more suitable to be employed as top of our proposed *Bot-DenseNet* model.

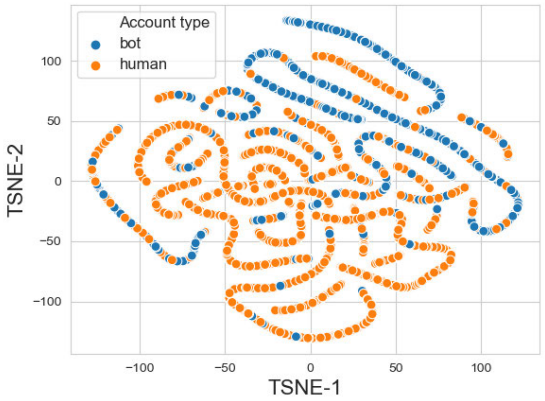
C. DECISION MAKING CRITERIA FOR BOT-DenseNet

After conducting the aforementioned experiments, a final decision to select the best configuration of the model is required. To do so, the following criteria have been considered: (i) The performance of the model in terms of F1-score in both training and validation to provide an objective criteria when making the final decision; (ii) The simplicity of the model in terms of both trainable parameters and the length of the input feature vector as it is remarked in Table 4; (iii) The simplicity of the final decision boundaries to distinguish between Bot and Human accounts which is assessed by observing the low-dimensional embeddings distribution in a TSNE projection. This aspect arises as crucial in order to propose a robust model capable of generalizing in future applications.

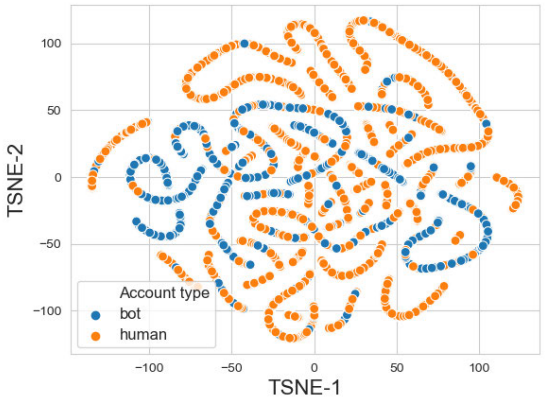
Thus, considering these elements for making the final decision both Table 4 as well as Figure 3 have been analysed to provide an objective decision.

Firstly, regarding performance, the best model is the one that uses the so-called RoBERTa Transfomer on top of it according to the F1-score achieved during the validation phase. Secondly, when it comes to simplicity, it is noticed that the best approaches are the ones that have employed LSTMs during the generation of the input vectors but they have achieved considerable lower results in terms of F1-score and therefore, they were discarded from the further analysis. On the other hand, the family of approaches that have employed a Pooling procedure are the ones with the highest level of complexity in terms of trainable parameters as well the lowest performance based on the F1-score, thus, they were directly discarded from the final decision.

Regarding simplicity in terms of the samples distribution after applying the TSNE method, Figures 4 and 3 shows that the boundaries achieved when using Transformers as part of the input feature vector are more suitable and simpler than the rest of configurations.

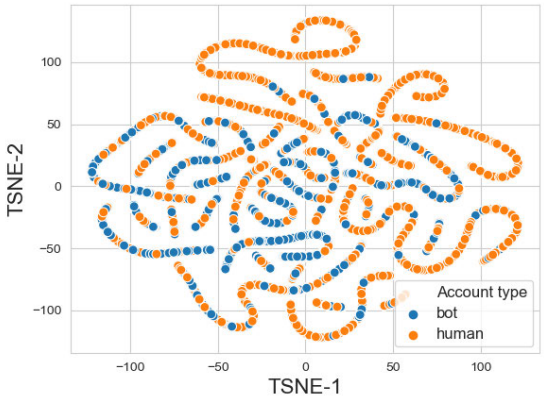


(a) BERT transformer approach for generating the text feature vector.

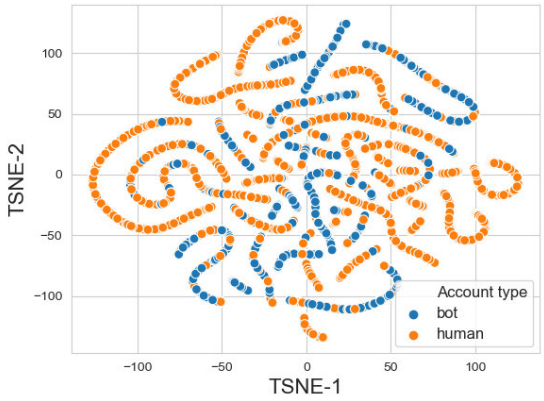


(b) RoBERTa transformer approach for generating the text feature vector.

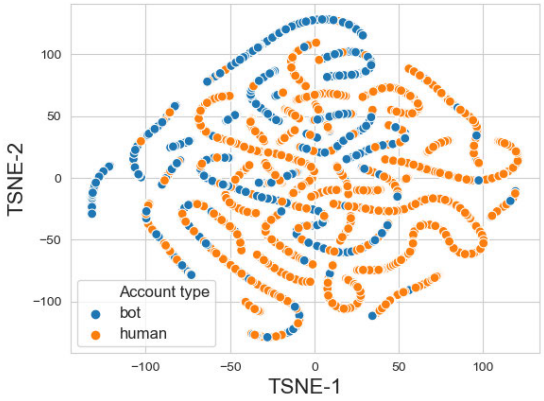
FIGURE 3. 2D projected representations of the embeddings obtained after training the proposed DL model *Bot-DenseNet* using as inputs the pre-trained embeddings from both BERT and RoBERTa models.



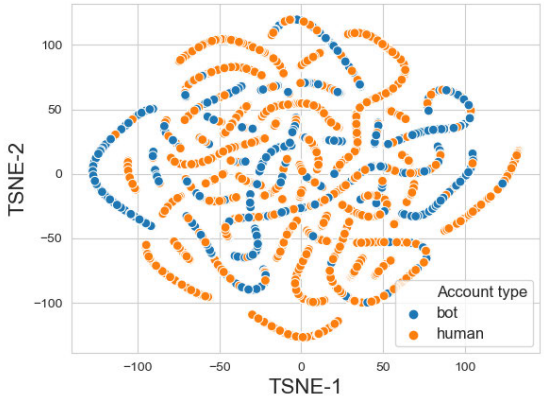
(a) Flair word embeddings combined via a pooling approach for document representation.



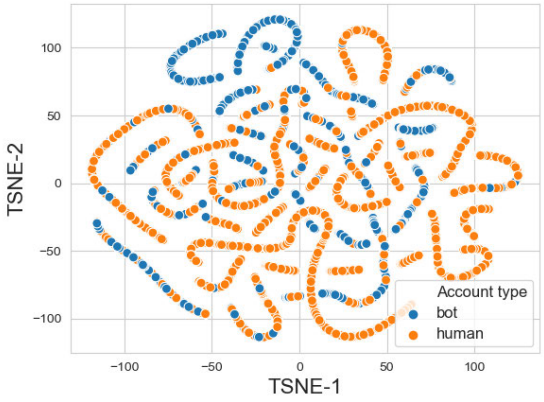
(b) Flair embeddings combined via LSTM approach for document representation.



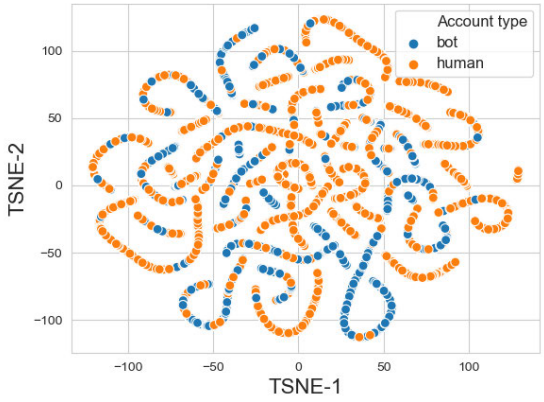
(c) BERT word embeddings combined via pooling approach for document representation.



(d) BERT word embeddings combined via LSTM approach for document representation.



(e) Flair + BERT embeddings and pooling approach for document representation.



(f) Flair + BERT embeddings and LSTM approach for document representation.

FIGURE 4. A set of 2D projected representations of the embeddings obtained in the proposed *Bot-DenseNet* via the so-called TSNE algorithm using a value of perplexity equal to 80.

Consequently, the final model uses the so-called RoBERTa transformer on top of it since it provides a remarkable trade-off between precision and simplicity, which are critical aspects to be considered when implementing Deep Learning models

Finally, Table 5 shows a comparison between our proposed DL architecture in comparison with previous studies in terms of both the performance throughout the F1-score metric, the learning approach as well as the capability of incorporating language-dependent features. This comparison enhances our proposed method in the following aspects: (i) *Language-dependent features* throughout the analysis of text descriptors via Transformers to improve the input feature vector and thus, the robustness of the system when facing non-English languages. (ii) *Hybrid learning*, since it produces embeddings throughout its intermediate hidden layers (unsupervised learning) once the model has been trained using a limited set of annotated data (supervised learning).

TABLE 5. A comparison of previous studies along with the proposed *Bot-Dense* model in terms of F1-Score, learning approach as well as its capability for analysing text features from multiple languages.

Study	*F1-score*	*Learning Method*	*Multilingual*
[45]	0.77	Supervised	No
[34]	>0.8	Supervised	No
[27]	0.87	Unsupervised	No
[38]	>0.7	Supervised	No
Bot-Dense	**0.77**	**Hybrid**	**Yes**

(iii) *Promising scoring metrics* in comparison with latest studies.

All the details of the implementation as well as the code for re-training or testing the proposed system can be found at this **Github** repository. More specifically, a Python library named as *User2Vec* was released to foster further research in this field.

V. CONCLUSION & FUTURE WORK

In this paper, a robust solution for detecting Bots in Twitter accounts has been described. In particular, this study has taken advantage of Transfer learning techniques via powerful state-of-the-art NLP models such as Transformers to extract compact multilingual representations of the text-based features associated with user accounts. By doing so, several constraints presented in previous studies related to process text-based features to improve the input feature vector from multiple languages were mitigated.

Furthermore, by employing the text encodings along with additional metadata on top of a dense-based neural network, a final classifier named as *Bot-DenseNet* has been trained and validated using a large set of samples collected via the Twitter API. More specifically, several experiments were conducted using different combinations of Word Embeddings, document embeddings (Pooling and LSTMs) and Transformers to obtain a single vector regarding the text-based features of the user account. Subsequently, a detailed comparison of the performance of the proposed classifier when using these approaches of Language Models as part of the input has been presented to investigate which input vector provides the best result in terms of performance simplicity in the generation of decision boundaries and feasibility.

In particular, the comparison of these experiments suggested that the *Bot-DenseNet* achieves the most adequate trade-off between performance and feasibility when using the so-called RoBERTa Transformer as part of the input feature vector.

Consequently, this paper provides two main contributions to the scientific community including a DL model for automatically detecting bots as well as a robust manner of representing any Twitter account as a low-dimensional feature vector throughout an intermediate layer of the aforementioned model. Moreover, this compact representation of the Twitter account can be used as a baseline for recommender or search engines, similarity analysis or any other application related with social media mining.

Finally, this study also remarks the outstanding performance of novel Transformers in downstream NLP tasks as the one presented, by providing a more robust input vector which leads the final classifier model to be more capable of extracting relevant low-level features from it. As Future work, the latest Transformers such as the GPT-3 [17] and T5 [33] will be considered for generating the input vector of the proposed DL model in order to compare the performance with the work described on this paper. Moreover, novel approaches such the one described by authors in [24] to automatically generate non-parametric Two-Sample tests based on the so-called Maximum Mean Discrepancy (MMD) [18], will be considered once all the user embeddings are generated, to find discrepancies and similarities between the distributions of both bots and non-bots embeddings.

REFERENCES

[1] Ž. Agić and I. Vulić, "JW300: A wide-coverage parallel corpus for low-resource languages," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*. Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 3204–3210.

[2] A. Akbik, T. Bergmann, D. Blythe, K. Rasul, S. Schweter, and A. Vollgraf, "FLAIR: An easy-to-use framework for state-of-the-art NLP," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Demonstrations*, 2019, pp. 54–59.

[3] A. Akbik, D. Blythe, and R. Vollgraf, "Contextual string embeddings for sequence labeling," in *Proc. 27th Int. Conf. Comput. Linguistics*, 2018, pp. 1638–1649.

[4] A. S. M. Alharbi and E. de Doncker, "Twitter sentiment analysis with a deep neural network: An enhanced approach using user behavioral information," *Cogn. Syst. Res.*, vol. 54, pp. 50–61, May 2019.

[5] N. R. Aljohani, A. Fayoumi, and S.-U. Hassan, "Bot prediction on social networks of Twitter in altmetrics using deep graph convolutional networks," *Soft Comput.*, vol. 24, pp. 11109–11120, Jan. 2020.

[6] M. Arora and V. Kansal, "Character level embedding with deep convolutional neural network for text normalization of unstructured data for Twitter sentiment analysis," *Social Netw. Anal. Mining*, vol. 9, no. 1, p. 12, Dec. 2019.

[7] A. Balestrucci, R. De Nicola, O. Inverso, and C. Trubiani, "Identification of credulous users on Twitter," in *Proc. 34th ACM/SIGAPP Symp. Appl. Comput.*, Apr. 2019, pp. 2096–2103.

[8] A. Bhoi and S. Joshi, "Various approaches to aspect-based sentiment analysis," 2018, *arXiv:1805.01984*. [Online]. Available: http://arxiv.org/abs/1805.01984

[9] Z. Chu, S. Gianvecchio, H. Wang, and S. Jajodia, "Detecting automation of Twitter accounts: Are you a human, bot, or cyborg?" *IEEE Trans. Depend. Sec. Comput.*, vol. 9, no. 6, pp. 811–824, Nov./Dec. 2012.

[10] T. Cooijmans, N. Ballas, C. Laurent, Ç. Gülçehre, and A. Courville, "Recurrent batch normalization," 2016, *arXiv:1603.09025*. [Online]. Available: http://arxiv.org/abs/1603.09025

[11] S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi, and M. Tesconi, "The paradigm-shift of social spambots: Evidence, theories, and tools for the arms race," in *Proc. 26th Int. Conf. World Wide Web Companion*, 2017, pp. 963–972.

[12] C. A. Davis, O. Varol, E. Ferrara, A. Flammini, and F. Menczer, "BotOrNot: A system to evaluate social bots," in *Proc. 25th Int. Conf. Companion World Wide*, 2016, pp. 273–274.

[13] A. Davoudi, A. Z. Klein, A. Sarker, and G. Gonzalez-Hernandez, "Towards automatic bot detection in twitter for health-related tasks," *AMIA Summits Transl. Sci. Proc.*, vol. 2020, p. 136, May 2020.

[14] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*. [Online]. Available: http://arxiv.org/abs/1810.04805

[15] J. Diesner, E. Ferrari, and G. Xu, in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining*. Sydney, NSW, Australia: ACM, Aug. 2017. [Online]. Available: https://dblp.org/rec/bib/conf/asunam/2017, doi: 10.1145/3110025.

[16] C. D. Santos and M. Gatti, "Deep convolutional neural networks for sentiment analysis of short texts," in *Proc. 25th Int. Conf. Comput. Linguistics (COLING)*, 2014, pp. 69–78.

[17] L. Floridi and M. Chiriatti, "GPT-3: Its nature, scope, limits, and consequences," *Minds Mach.*, vol. 30, pp. 681–694, Nov. 2020.

[18] R. Gao, F. Liu, J. Zhang, B. Han, T. Liu, G. Niu, and M. Sugiyama, "Maximum mean discrepancy is aware of adversarial attacks," 2020, *arXiv:2010.11415*. [Online]. Available: http://arxiv.org/abs/2010.11415
[19] X. He, X. Du, X. Wang, F. Tian, J. Tang, and T.-S. Chua, "Outer product-based neural collaborative filtering," 2018, *arXiv:1808.03912*. [Online]. Available: http://arxiv.org/abs/1808.03912
[20] J. Im, E. Chandrasekharan, J. Sargent, P. Lighthammer, T. Denby, A. Bhargava, L. Hemphill, D. Jurgens, and E. Gilbert, "Still out there: Modeling and identifying Russian troll accounts on Twitter," in *Proc. 12th ACM Conf. Web Sci.*, Jul. 2020, pp. 1–10.
[21] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*. [Online]. Available: http://arxiv.org/abs/1502.03167
[22] J. Knauth, "Language-agnostic Twitter-bot detection," in *Proc. Int. Conf. Recent Adv. Natural Lang. Process. (RANLP)*, 2019, pp. 550–558.
[23] Z. Lin, S. Mu, F. Huang, K. A. Mateen, M. Wang, W. Gao, and J. Jia, "A unified matrix-based convolutional neural network for fine-grained image classification of wheat leaf diseases," *IEEE Access*, vol. 7, pp. 11570–11590, 2019.
[24] F. Liu, W. Xu, J. Lu, G. Zhang, A. Gretton, and D. J. Sutherland, "Learning deep Kernels for non-parametric two-sample tests," 2020, *arXiv:2002.09116*. [Online]. Available: http://arxiv.org/abs/2002.09116
[25] Y. Liu, P. Dmitriev, Y. Huang, A. Brooks, and L. Dong, "An evaluation of transfer learning for classifying sales engagement emails at large scale," 2019, *arXiv:1905.01971*. [Online]. Available: http://arxiv.org/abs/1905.01971
[26] P. Lynn, "The advantage and disadvantage of implicitly stratified sampling," *Methods, Data, Analyses*, vol. 13, no. 2, p. 14, 2019.
[27] M. Mazza, S. Cresci, M. Avvenuti, W. Quattrociocchi, and M. Tesconi, "RTbust: Exploiting temporal patterns for botnet detection on Twitter," in *Proc. 10th ACM Conf. Web Sci.*, 2019, pp. 183–192.
[28] A. Minnich, N. Chavoshi, D. Koutra, and A. Mueen, "BotWalk: Efficient adaptive exploration of Twitter bot networks," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining*, Jul. 2017, pp. 467–474.
[29] U. Naseem, I. Razzak, K. Musial, and M. Imran, "Transformer based deep intelligent contextual embedding for Twitter sentiment analysis," *Future Gener. Comput. Syst.*, vol. 113, pp. 58–69, Dec. 2020.
[30] M. Orliński and N. Jankowski, "Fast t-SNE algorithm with forest of balanced LSH trees and hybrid computation of repulsive forces," *Knowl.-Based Syst.*, vol. 206, Oct. 2020, Art. no. 106318.
[31] J. Pizarro, "Using N-grams to detect bots on Twitter," in *Proc. CLEF, Working Notes*, 2019, pp. 1–10.
[32] M. Polignano, M. G. de Pinto, P. Lops, and G. Semeraro, "Identification of bot accounts in Twitter using 2D CNNs on user-generated contents," in *Proc. CLEF, Working Notes*, 2019, pp. 1–11.
[33] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *J. Mach. Learn. Res.*, vol. 21, no. 140, pp. 1–67, 2020.
[34] J. Rodríguez-Ruiz, J. I. Mata-Sánchez, R. Monroy, O. Loyola-González, and A. López-Cuevas, "A one-class classification approach for bot detection on Twitter," *Comput. Secur.*, vol. 91, Apr. 2020, Art. no. 101715.
[35] K. Shuang, H. Guo, Z. Zhang, J. Loo, and S. Su, "A word-building method based on neural network for text classification," *J. Exp. Theor. Artif. Intell.*, vol. 31, no. 3, pp. 455–474, May 2019.
[36] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[37] D. Stojanovski, G. Strezoski, G. Madjarov, and I. Dimitrovski, "Twitter sentiment analysis using deep convolutional neural network," in *Proc. Int. Conf. Hybrid Artif. Intell. Syst.* Springer, 2015, pp. 726–737. [Online]. Available: https://scholar.googleusercontent.com/scholar.bib?q=info:HnIU7VyTzLUJ:scholar.google.com/& output=citation&scisdr=CgXc4k0kELTt-pJoVlM:AAGBfm0AAAAAYGxtTlO0qf0SoEojztYZqYNU1uzAmqAp&sci sig=AAGBfm0AAAAAYGxtTlfXvsJzQ3eCFjVQwVDi0pipTQma&scisf=4&ct=citation&cd=-1&hl=es
[38] O. Varol, E. Ferrara, C. A. Davis, F. Menczer, and A. Flammini, "Online human-bot interactions: Detection, estimation, and characterization," 2017, *arXiv:1703.03107*. [Online]. Available: http://arxiv.org/abs/1703.03107
[39] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
[40] I. Vogel and P. Jiang, "Bot and gender identification in Twitter using word and character N-grams," in *Proc. CLEF, Working Notes*, 2019, pp. 1–9.
[41] B. Wang and C.-C. J. Kuo, "SBERT-WK: A sentence embedding method by dissecting bert-based word models," 2020, *arXiv:2002.06652*. [Online]. Available: http://arxiv.org/abs/2002.06652
[42] L. Wang, J. Niu, and S. Yu, "SentiDiff: Combining textual information and sentiment diffusion patterns for Twitter sentiment analysis," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 10, pp. 2026–2039, Oct. 2020.
[43] T. Wolf *et al.*, "HuggingFace's transformers: State-of-the-art natural language processing," 2019, *arXiv:1910.03771*. [Online]. Available: http://arxiv.org/abs/1910.03771
[44] K. Yang, O. Varol, C. A. Davis, E. Ferrara, A. Flammini, and F. Menczer, "Arming the public with artificial intelligence to counter social bots," *Hum. Behav. Emerg. Technol.*, vol. 1, no. 1, pp. 48–61, Jan. 2019.
[45] K.-C. Yang, O. Varol, P.-M. Hui, and F. Menczer, "Scalable and generalizable social bot detection through data selection," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, 2020, pp. 1096–1103.
[46] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning requires rethinking generalization," 2016, *arXiv:1611.03530*. [Online]. Available: http://arxiv.org/abs/1611.03530
[47] S. Zhang, X. Xu, Y. Pang, and J. Han, "Multi-layer attention based CNN for target-dependent sentiment classification," *Neural Process. Lett.*, vol. 51, no. 3, pp. 2089–2103, Jun. 2020.
[48] J. Zhu, C. Huang, M. Yang, and G. P. Cheong Fung, "Context-based prediction for road traffic state using trajectory pattern mining and recurrent convolutional neural networks," *Inf. Sci.*, vol. 473, pp. 190–201, Jan. 2019.

DAVID MARTÍN-GUTIÉRREZ (Member, IEEE) was born in Madrid, Spain, in 1993. He received the bachelor's degree in audio-visual system engineering from Carlos III University, in 2016, and the master's degree in signal processing and machine learning from Universidad Politécnica de Madrid, in 2019, where he is currently pursuing the Ph.D. degree in multimodal feature learning.

From 2016 to 2017, he was a Software Developer with Everis S.L and later on as a Data Fusion Engineer with Ixion Industry and Aerospace. Since 2017, he has been working as the Research Scientist and the Data Scientist with the Visual Telecommunications Applications Research Group developing artificial intelligent algorithms related to multimedia content in several national and EU projects.

GUSTAVO HERNÁNDEZ-PEÑALOZA (Member, IEEE) received the Telecom Engineering degree from Universidad Santo Tomás, in 2007, the Master of Science degree in telecommunication technologies, system and networks from Universidad Politécnica de Valencia (UPV), in 2009, and the Master of Business Administrator (M.B.A.) and the Ph.D. degree *(cum laude)* from Universidad Politécnica de Madrid (UPM), in 2019. He is currently working as a Postdoctoral Fellow with UPM. From 2010 to 2013, he was an Associate Research Fellow with Universidad de Valencia (UV). His main research interest includes artificial intelligence (AI) applied to healthcare. He has been involved in multiple National and European funded projects in technical and management activities.

ALBERTO BELMONTE HERNÁNDEZ (Member, IEEE) received the degree in telecommunication engineering, the master's degree in communication systems, and the Ph.D. degree *(cum laude)* from Universidad Politécnica de Madrid (UPM), in 2014, 2016, and 2020, respectively. He is currently an Assistant Professor in several subjects. Since 2016, he has been with the Visual Telecommunications Applications Group (GATV), UPM. He is actively working on artificial intelligent applied to multimedia content and sensors for pattern detection, recognition, and fusion. He has been developing technical parts in national and EU projects.

ALICIA LOZANO-DIEZ received the double degree in computer science engineering and mathematics, the master's degree in research and innovation in ICT, and the Ph.D. degree *(cum laude)* from Universidad Autónoma de Madrid (UAM), Spain, in 2012, 2013, and 2018, respectively. She was an Assistant Professor with UAM. Since 2012, she has been with the Audias Research Group, UAM. During the Ph.D. degree, she interned at the Speech Group (Speech@FIT) with the Brno University of Technology (BUT), Czech Republic, and with the SRI International (STAR Lab), USA. Her research interests include deep neural networks (DNN) for automatic language and speaker recognition. In 2019, she got the H2020 Marie Curie funding for the project ''Robust End-To-End SPEAKER Recognition Based on Deep Learning and Attention Models'' and joined the Speech@FIT (BUT) as a Postdoctoral Researcher, and will resume to her position at UAM in 2021.

FEDERICO ÁLVAREZ (Member, IEEE) received the Telecom Engineering degree (Hons.) and the Ph.D. degree *(cum laude)* from Universidad Politécnica de Madrid (UPM), in 2003 and 2009, respectively. He is currently working as an Assistant Professor with UPM. Since 2003, he has been working for the research group with the Visual Telecommunications Applications Group (GATV), UPM. He is the author and a coauthor of more than 60 articles and several books, book chapters, and patents in the field of ICT networks and audiovisual technologies. He has been participating with different managerial and technical responsibilities in several national and EU projects, being coordinator of five EU projects in the last six years. He had participated in national and international standardization for a DVB, CENELEC TC206, and so on. He is also a member of the Program Committee of several scientific conferences.