 DOI: 10.1109/BigData59044.2023.10386674

One-GPT: A One-Class Deep Fusion Model for Machine-Generated Text Detection

Roberto Corizzo
Department of Computer Science
American University
Washington, DC, USA
rcorizzo@american.edu

Sebastian Leal-Arenas
Department of Linguistics
University of Pittsburgh
Pittsburgh, PA, USA
sal209@pitt.edu

***Abstract*—On the brink of the one-year anniversary since the public release of ChatGPT, scholarly research has directed their interest toward detection methodologies for machine-generated text. Different models have been proposed, including feature-based classification and detection approaches, as well as deep learning architectures, with a small portion of them integrating contextual information to enhance accurate predictions. Moreover, detection approaches explored thus far have focused primarily on English datasets, with limited attention given to the examination of similar methods in other languages. As a result, the applicability and efficacy of these methods in linguistically diverse contexts remains underexplored. In this paper, we present a one-class deep fusion model that considers both contextual text features derived from word embeddings and linguistic features to detect machine-generated texts in English and Spanish. Experimental results indicated that our model outperformed popular baseline one-class learning models in the detection task, presenting higher accuracy scores in the English dataset. Results are discussed in comparison to competing classifiers as well as the language biases found in detection models.**

***Index Terms*—Text classification, one-class learning, natural language processing, data fusion, ChatGPT**

I. Introduction

Chat Generative Pre-trained Transformer, or ChatGPT, has drawn significant attention since its introduction to the general public in November 2022. Its adoption has generated a series of reactions in social media, where uses and misuses are presented. In academic contexts, scholars [1] [2] [3] [4] have mentioned the potential for course assignments to be completed entirely by Artificial Intelligence (AI) systems, leaving instructors in a predicament. This concern stems from the growing capabilities of AI in Natural Language Processing (NLP) and text generation. The written content generated by systems like ChatGPT has the potential to exhibit levels of text production suspiciously similar to those of humans, igniting the need for detection systems.

Classification methods capable of discerning machine-generated from human-generated text have made significant strides within the academic domain. These state-of-the-art techniques can be broadly categorized into two main groups: neural network-based methods and feature-based methods. The former category encompasses end-to-end model architectures [5] [6] and neural network models designed to generate a numerical vector representation of the text, commonly referred to as word embedding [7] [8] [9] [10] [11] [12]. Unlike earlier methods that relied on word-index-based and term-frequency-based approaches, word embedding captures the semantic meaning of the language. The latter class frequently relies on the utilization of linguistic attributes, considering factors such as repetitiveness and emotional semantics [2] [13] [14] [15] [16] [17]. Notwithstanding these advancements, research in this area presents its limitations, such as a limited exploitation of information from multiple modalities, use of generic linguistic features [18], and adoption of fully supervised models [2], which entail the unrealistic assumption that both human and machine-generated texts are available as training data.

While one-class learning approaches represent a viable option to overcome such limitation and have demonstrated success in a number of domains [19] [20] [21] [22] [23], approaches for machine-generated text detection are still scarce, and limited to the adoption of simple one-class baseline models and basic linguistic features [4].

In this paper, we address these gaps by proposing One-GPT, a one-class deep neural-network-based fusion model that jointly harnesses contextual textual features through word embeddings, as well as linguistic features. A fusion branch is responsible for integrating and further refining the contextual and semantically rich information extracted by the text and linguistic features branches, leading to powerful high-level features that provide robust and accurate detections of machine-generated text. Our experimental evaluation with two real-world datasets containing essays written by second language (L2) learners in two languages, English and Spanish, highlights the competitiveness of the proposed model, which achieves a higher level of precision in the identification process when compared to widely accepted one-class benchmark models.

The subsequent sections of the paper are organised as follows: Section II provides an overview of pertinent research. Section III describes the proposed method. Section IV defines the research questions and our experimental setup, followed by a discussion of the results in Section V. Finally, Section VI concludes the paper, summarizing our contributions and outlines prospects for future work.

979-8-3503-2445-7/23/$31.00 ©2023 IEEE

II. Related Work

A. Neural network-based and feature-based approaches

Machine learning-based approaches employed for text analysis can tackle classification and detection tasks. Within the classification domain, these approaches can be classified into two categories: neural network-based and feature-based methodologies. Neural network-based approaches can perform end-to-end classification independently. This is achieved, for example, by adding a softmax output layer to their neural network architecture. Text classification through end-to-end neural networks can be also performed with Bi-LSTM (Bidirectional Long Short-Term Memory) models, which are preferred for sentiment classification [5]. The authors in undertook a deep multi-task learning approach that incorporated novelty detection, emotion recognition, sentiment prediction, and misinformation detection within one architectural framework. The work in [6] used a Character Text Image Classifier (CTIC). This model encompasses elements such as word, character and sentence embeddings as well as images.

Alternatively, neural networks can be used to extract vector embeddings which are subsequently used for classification or detection tasks leveraging external models. Authors in [7] adopted Doc2Vec embeddings of emails along with RF/SVM models to categorise emails as either phishing or legitimate. The authors in [8] harnessed BERT (Bidirectional Encoder Representations from Transformers) embeddings and a CNN classifier to identify spam tweets, amalgamating topic-based features with contextual BERT embeddings. Furthermore, the authors in [9] conducted a comparative analysis of different feature representations, including BERT and Bi-LSTM in conjunction with SVM and Naive Bayes. Their research showed the effectiveness of these results in discerning anti-vaccination tweets during the COVID-19 pandemic.

Feature-based approaches involve the combination of feature extraction approaches and machine-learning classification models, such as Random Forests (RF) and Support Vector Machines (SVM). Numerous studies have demonstrated the efficacy of these methods in various classification tasks across different domains. Despite the effectiveness of word embeddings and vector encodings extracted via neural networks [10] [11] [12], high-level linguistic features tailored to a given classification problem have proven beneficial, as their transparency, explainability, and discriminative capability is a powerful means to represent the data [13]. Methods for distinguishing human and machine-generated text that rely on linguistic features are premised on the existence of certain distinctions between the two forms of text in certain dimensions. Feature-based methods have been applied to a wide range of tasks, including readability assessment [24] [25] and authorship attribution [26].

Regarding feature-based text analysis, scholars have examined aspects such as punctuation distribution and its occurrence in sentences and paragraphs. These measures are effective in fake news detection [13] [14]. Machine-generated texts exhibit repetitive language patterns [15] and rely on frequently used words [16]. Detection of repetitiveness has involved the analysis of lexical repetition through n-gram word overlap [18]. Moreover, machine-generated text lacks emotional nuances and biases found in natural languages. As a result, sentiment-related words receive higher scores when assessing a text's topicality with sentiment analysis models [17]. The extraction of linguistic features offers a viable means of text detection. However, these features are often simplistic and may not uncover complex relationships as effectively as neural network-based methods.

B. Other approaches

Both neural-network-based and feature-based methods face limitations rooted in their need for training data on both classes. Nonetheless, the scarcity of data for the positive class (anomaly class to detect) poses a challenge alongside the potential variability of the class, which is unknown during training.

Thus, one-class learning emerges as a more suitable alternative to fully supervised models. Examples of domains where this learning paradigm was successfully applied include cybersecurity [19], oil and gas stations [20], geo-distributed data in smart grids [21], music streaming data [22], and biology [23]. Despite the differences in approaches, the common characteristic of one-class learning models' task is to detect deviations from the background distribution of the data learned during the training stage. While successful in a wide range of domains and applications, scarce attention has been devoted to machine-generated text content. The work in [4] assesses the efficacy of one-class models on essay data, leveraging baseline approaches with simple sets of linguistic features.

Recently, detection approaches tailored for machine-generated text have been categorized as white box and black box [27]. White box detectors do not require training an external machine learning classifier [28]. However, a significant drawback lies in the need to implement specialised language models to access the token probability output. This direction is frequently unattainable due to the closed-source nature of recently available language models. It is also impractical for non-expert users owing to high technical and computational requirements, thereby constituting a substantial barrier. On the other hand, Black-box approaches generally entail training a classifier using data without requiring any knowledge or access to specific language models. As a result, these approaches present the potential to detect texts generated by different language models. To this end, the adoption of linguistic features can be particularly effective to detect machine-generated text as shown in [29], where n-gram analysis is conducted. However, existing approaches tailored for machine-generated text are still rather simplistic and underdeveloped compared to other domains, where more sophisticated models including deep learning-based architectures have surpassed simple shallow models. Thus, our effort in the present paper is to propose a one-class fusion model that combines the expressiveness of linguistic features with the modeling power of neural network-based approaches via data fusion.

III. METHOD

Our proposed fusion model consists of three main parts: the Text branch, the Linguistic Features branch, and the Fusion branch that jointly processes Text and Linguistic Features together through an autoencoder-like model architecture.

A. Text branch

This branch deals with the extraction of vector embeddings from text through a Doc2Vec model. Doc2Vec [30] is a popular document embedding technique extending Word2Vec [31]. It yields a numerical feature vector of a document, regardless of its length. Differently than Word2Vec, which extracts word-level semantic vector representations, Doc2Vec extracts document-level embeddings, by simultaneously learning a distributed representation for both words and documents. It leverages the Distributed Bag of Words (PV-DBOW) and the Distributed Memory Model of Paragraph Vectors (PV-DM) approaches to extract effective vector embedding representations from textual documents.

The PV-DBOW model tries to predict individual words in a document with the exclusive use of the document vector. Its objective function can be formalized as:

$$\max_{\theta} \sum_{t=1}^{T} \log P(C_t|d,\theta),$$

where T is the number of training sentences, C_t is the set of words in sentence t, d is a document vector, and θ represents the model parameters.

The PV-DM model, instead, tries to maximize the likelihood of predicting the next word in a sentence given the context of the document vector and a set of surrounding words. Its objective function can be formalized as:

$$\max_{\theta} \sum_{t=1}^{T} \sum_{w \in C_t} \log P(w|d,\theta),$$

In our work, we adopt the PV-DM model, since it takes into account the order of words in the document, allowing it to capture semantic and contextual information. In contrast, PV-DBOW treats each document as a bag of words and ignores word order, which may be sufficient for document retrieval applications, but leads to a loss of sequential and semantic information that make it unreliable for more complex detection tasks.

In our workflow, Doc2Vec is trained with human written texts and used to extract a contextual vector embedding representation of size emb for any given text at inference time. This vector representation is fed, alongside with the linguistic feature representation to a fusion layer and it is further processed by the fusion branch.

B. Linguistic Features branch

Machine-generated and human-generated texts exhibit distinct linguistic features for differentiation [25]. This branch is responsible for the extraction and exploitation of different linguistic features, which we categorised into groups: Text, Repetitiveness, Emotional Semantics, Readability, and Part-Of-Speech (POS).

Text: These features pertain to quantifiable data characteristics primarily found in text, such as distinct type of punctuation marks, number of: Oxford commas [4] [2], paragraphs [32], full stops and commas [32], and average sentence length [33]. Previous work [32] observed the use of more punctuation marks in human text.

Repetitiveness: It has been stated that machine-generated text often relies heavily on frequent words, leading to repetition and limited narrative diversity [34]. To quantify these traits, we extracted unique n-grams, counted their occurrences, and calculated the unigram, bi-gram and tri-gram overlap ratio [35]. Additionally, we assessed word frequency in our datasets by comparing it to the 5K and 10K most used words in their respective languages [18]. This helps gauge dataset vocabulary alignment with general language use.

Emotional Semantics: Human-generated text is characterised by greater emotional variation, complexity and subjectivity compared to machine-generated text [36] [37]. Two Emotional Semantic features, Polarity and Subjectivity, are extracted using TextBlob, an open-source, lexicon-driven Python library [38]. Polarity indicates sentiment ranging from -1,0 (negative) to 1.0 (positive). Subjectivity can be very objective (0.0) or very subjective (1.0). For Sentiment (ES), we adopted a Naive Bayes model trained on 800,000 Spanish reviews. For Sentiment (Multi-language), we leveraged the *bert-base-multilingual-uncased-sentiment* model trained on star reviews with multiple languages. Negative reviews (-1.0), neutral (0.0) and positive reviews (1.0) are present. In Sentiment Score, continuous scores 0.0 (negative) and 1.0 (positive) were normalised.

Readability: Assesses vocabulary compleity in text [24] using 13 different indicators, with four being exclusive to Spanish. Each indicator possesses its own unique formula and interpretation. For a detailed account of each index, refer to [4]. Employing various readability indices yields valuable insights into the complexity of each text.

Part-of-Speech (POS): The distribution of different parts of speech can be quantified to account for the lack of syntactic and lexical diversity. These word classes are instrumental in sentence analysis and comprehension. POS has been employed to indicate the relative frequency of certain word types in a text [18] [24]. Due to the intricacies of languages regarding word order, we employed SpaCy [1] to parse and tag the data.

Table I presents a comprehensive list of the features analyzed in this paper with examples. A graphical view of the feature extraction process for document embeddings and linguistic features is presented in Figure 2.

C. Fusion branch

The outputs of the Text (emb) and the linguistic (49) branches are fused, resulting in a new representation of size ($emb + 64$). After fusion, a dropout regularization layer and 5

[1] https://spacy.io/

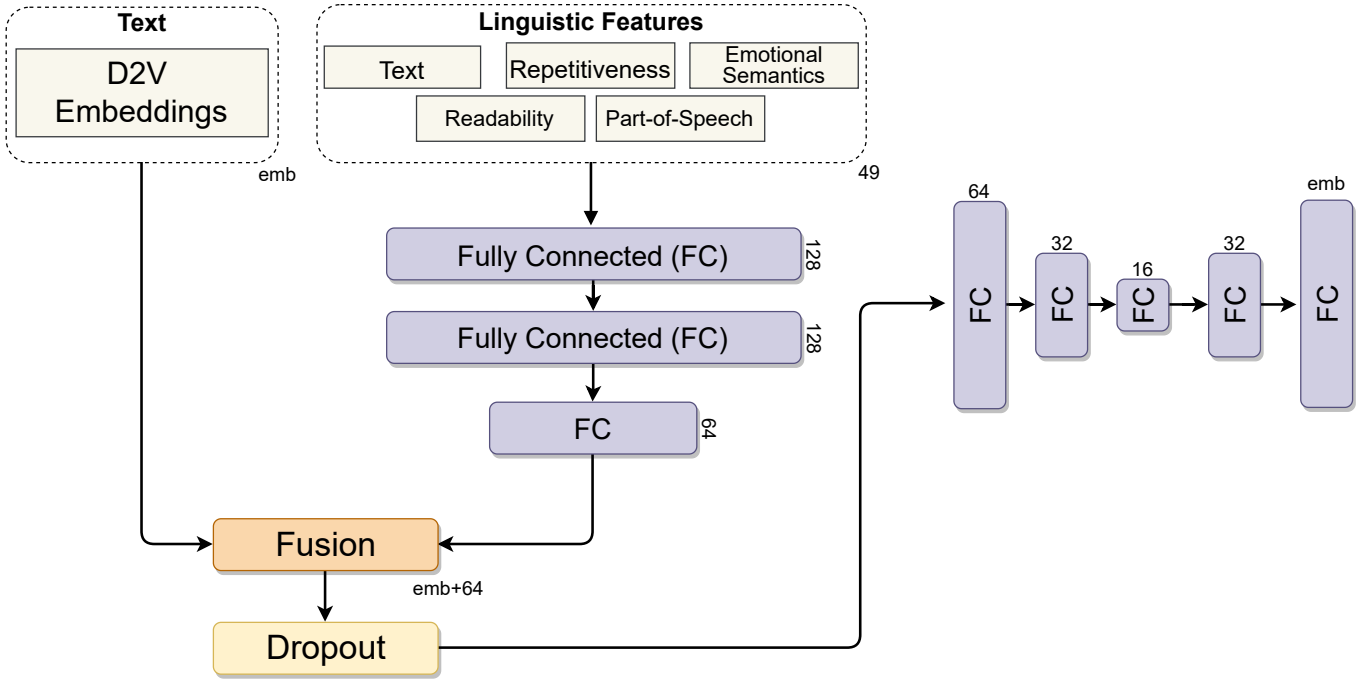

Fig. 1. Proposed one-class deep fusion model architecture for machine-generated text detection. The model aims to effectively integrate the contribution of contextual features in document embeddings with linguistic features (Text, Repetitiveness, Emotional Semantics, Readability, and Part-of-Speech). Subsequent to their integration, a Dropout regularization layer and an autoencoder-like reconstruction-based model branch made of several Fully Connected (FC) layers further processes this representation, facilitating the extraction of hidden semantic and contextually rich information that allows the model to accurately detect machine-generated text.

fully connected layers ($64-32-16-32-emb$) are responsible for integrating and further processing the contextual and semantically rich information extracted by the two independent branches. The goal of this part of the model is to extract powerful, high-level features that are discriminant for the text detection task.

The structure of the fusion branch is an autoencoder-like model architecture, which learns to reconstruct document vector embeddings, given the fused deep representation extracted by the single branches. By doing so, we learn a compressed representation (encoding) of both data views and reconstruct the same from this embedded representation. This branch is trained adopting a reconstruction loss, and the process can be formalized as:

$$AE(\theta) = \frac{1}{N}\sum_{i=1}^{N}\left\|X^{(i)} - \hat{X}^{(i)}\right\|_2^2,$$

where N is the number of human-written documents, $X^{(i)}$ is the input data for the i-th text document, $\hat{X}^{(i)}$ is the reconstructed data for the i-th text document, and θ represents model weights optimized via stochastic gradient descent.

Once the model is trained in an end-to-end fashion using exclusively human written texts, we compute the reconstruction error distribution of training data. We leverage the reconstruction error distribution of training data and adopt a threshold-based approach to obtain the final prediction label $pred(d_i)$:

$$pred(d_i) = \begin{cases} 1, & \text{if } e_{d_i} > \sigma \\ 0, & \text{otherwise,} \end{cases}$$

where e_i represents the reconstruction error of an unlabelled text document i, and σ is the 99^{th} percentile of the reconstruction error distribution of training data. The rationale is that the model should be able to accurately reconstruct human written texts, while it should yield a higher reconstruction error for machine-generated texts, allowing us to detect them as out-of-distribution samples.

IV. Experiments

This study addressed the following research questions:

- **RQ1**: Does the use of linguistic features provide accurate detection of human vs. machine-generated texts using one-class learning models?
- **RQ2:** Can our deep fusion one-class learning model outperform baseline one-class learning models in the detection of machine-generated text with exclusive exploitation of human-written data for model training?
- **RQ3:** To what extent does language affect detection accuracy of one-class learning models?

To answer these RQs, specific datasets, setup, and metrics[2] were used.

[2]Code and datasets are available at: https://github.com/rcorizzo/one-gpt

TABLE I
LINGUISTIC FEATURES IN THE STUDY: TYPES, NAMES, AND EXAMPLES (WHERE APPLICABLE).
AN ASTERISK SYMBOL $(*)$ DENOTES THAT THE FEATURE VALUE IS NORMALIZED BY DOCUMENT LENGTH.

Type	**Feature**	**Example**
Text	Distinct types of punctuation marks +	. , ; ...
	Number of Oxford commas	*" ... me gusta ir a la playa, al cine, y a la piscina"*
	Number of paragraphs	/
	Number of full stops $*$	/
	Number of commas $*$	/
	Average sentence length $*$	/
Repetitiveness	Unigram Overlap	*just ... just ...*
	Bi-gram Overlap	*just like ... just like...*
	Tri-gram Overlap	*just like I ... just like I ...*
	Matches in the 5K most common words of the language $*$	*be, a, in* (EN) *... yo, estar, bueno* (ES) ...
	Matches in the 10K most common words of the language $*$	*cigarrete, lobby, punch* (EN) *... fuente, intranquilo, desobediente* (ES) ...
Emotional Semantics	Polarity	$[-1.0, 1.0]$
	Subjectivity	$[0.0, 1.0]$
	Sentiment (ES)	$[-1.0, 1.0]$
	Sentiment (Multi-language)	$[-1.0, 1.0]$
	Sentiment Score (Multi-language)	$[0.0, 1.0]$
Readability	Flesch Reading Ease Score	[0, 100]
	Flesch Kincaid Grade Score	[-3.40, no limit]
	Smog Index Score	[1, 240]
	Coleman Liau Index Score	[1, 11+]
	Automated Readability Index	[1, 14]
	Dale-Chall Readability Score	[0, 10]
	Difficult Words Score	Varies
	Linsear Write Formula Score	[0, 11+]
	Gunning Fog score	[6-17]
	Fernández-Huerta Score (SPAN)	[0, 100]
	Szigriszt-Pazos Score (SPAN)	[0, 100]
	Gutiérrez de Polini Score (SPAN)	[0, 100]
	Crawford Score (SPAN)	[6-17]
Part-of-Speech (POS)	ADJ (Adjective) *	*scary, subjective*
	ADP (Adverbial Phrase) *	*very slowly, quite easily*
	ADV (Adverb) *	*respectfully, thankfully*
	AUX (Auxiliary) *	*"... that **has** been discussed before..."*
	CCONJ (Coordinating Conjunction) *	*or, and, but*
	DET (Determiner) *	*the, a, an*
	NOUN *	*animals, jail*
	NUM *	*four, millions*
	PRON (Pronoun) *	*I, you, him, hers*
	PROPN (Proper Noun) *	*America, Europe*
	PUNCT (Punctuation) *	. ,
	SCONJ (Subordinating Conjunction) *	*although, because, whereas*
	SYM (Symbol) *	*$*
	VERB *	*to teach, to present, to decide*
	SPACE (Number of spaces) *	*count of number of spaces*

A. Datasets

We conducted our analyses using two datasets comprising short essays authored by L2 English and L2 Spanish speakers, precisely:

1) **English:** We analyzed a subset of the Uppsala Student English Corpus (USE)[3], containing essays written by learners of English at three different proficiency levels. We focused on essays in the 'a2' folder due to the size of the dataset and its suitability. Our analysis involved 335 essays.
2) **Spanish:** We utilized the UC Davis Corpus of Written Spanish, L2 and Heritage Speakers (COWSL2HS[4]), comprising short essays obtained from students in university-level Spanish courses. In this study, we leveraged 350 essays for analysis.

We supplemented the two datasets with an equal number of machine-generated tests. We instructed ChatGPT as follows: "Write an 800-word essay on [topic]." The same instruction was used to generate Spanish data. Topics were matched with those present in the human-written dataset. Thus, the English dataset comprises 670 essays, evenly split between human-generated (335) and GPT-generated (335) essays. Similarly, the Spanish dataset consists of 700 essays, with an equal division of human-generated (350) and GPT-generated (350) essays. Excerpts can be found in II.

B. Setup

Our fusion model is trained using the Adam optimizer. The values of the hyperparameters are: *batch size = 32, learning*

[3]https://ota.bodleian.ox.ac.uk/repository/xmlui/handle/20.500.12024/2457
[4]https://github.com/ucdaviscl/cowsl2h

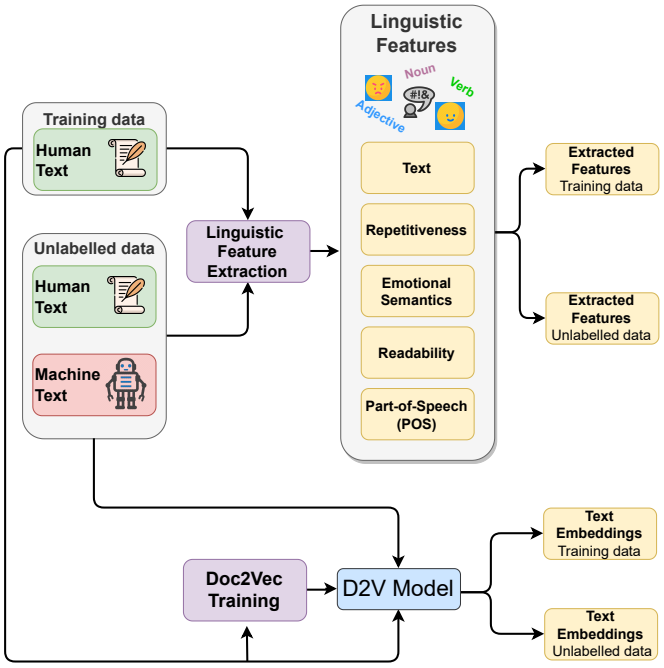

Fig. 2. Feature extraction workflow of the proposed method. Linguistic feature extraction is performed to extract text, repetitiveness, emotional semantics, readability, and Part-Of-Speech (POS) numerical features from data. In parallel, a Doc2Vec model is trained with human written texts and used to extract textual vector embeddings. Both representations are fed to our deep fusion model.

rate = 1e-3, epochs = 400. For the Doc2Vec branch: *min count = 1, iters = 200*, and we consider different values for the dimensionality of the embedding vector, namely, *35, 50,* and *100*, reporting the results obtained with the best configuration. The model is implemented in Python and TensorFlow 2.

We adopt the following baselines in the realm of one-class-learning approaches:

- **One-Class Support Machines (OCSVM)** [39]: Similarly to Support Vector Machines, it identifies a hyperplane to separate data instances from two classes. However, the hyperplane is used to cover all background data samples (human texts). After the training phase, the OCSVM hyperplane can detect a new data point (textual document) as belonging to the background data seen during training (human) or not (machine-generated), based on its geometric location within the decision boundary.
- **Local Outlier Factor (LOF)** [40]: It measures the deviation of local density of data points (textual documents) with respect to their neighbors. It returns an anomaly score based on the ratio between the local density of data points and the average local density of their nearest neighbors. An anomaly score lower than 1 for a data point is representative of a higher density than neighboring data points (inlier), whereas a value greater than 1 is representative of a lower density than neighboring data points (outlier).

TABLE II
EXCERPTS FROM SHORT ESSAYS IN OUR DATASETS. THE FIRST TWO PARAGRAPHS WERE TAKEN FROM THE ENGLISH DATASET (TOP), AND THE LAST TWO FROM THE SPANISH DATASET (BOTTOM).

Topic: We should not exploit animals	
Human	**Generated**
Today most of us accept science. Science of today, especially Darwins Evolutionary Theory teaches us that the human race has developed from animals and that human beings basically are animals among other animals. Even though we do agree with this aspect of science we tend to consider man as the crown of creation with a natural right to treat animals as we please and use them for our own purposes as food, tools and experiment instruments. ...	Exploiting animals, whether for food, clothing, entertainment, or other purposes, is a controversial issue that has been the subject of much debate in recent years. While some argue that animals are a natural resource to be used for human benefit, others believe that we have a moral obligation not to harm or exploit them. ...
Topic: Un lugar que no te gusta (a place you do not like)	
Human	**Generated**
No me gusta el dentista. Es un lugar aburrido y no está divertido. Necesita ir al dentista una o dos veces al año y sentarse cuando ellos limpiar sus dientes. Que no me gusta el dentista? Bueno. La primer razón por la que no me gusta al dentista es ellos siempre te intentar a hablar cuando limpian sus dientes. Dicen "Como estás" y "De dónde eres" y no puede responder porque tiene agua y pasta dental in su boca! ...	Un lugar desagradable que recuerdo es el hospital donde tuve que pasar una semana hace unos años debido a una enfermedad grave. Desde el momento en que llegué al hospital, tuve una sensación de incomodidad y ansiedad. El olor a medicamentos y a limpieza química era fuerte y desagradable, y el ambiente en general era frío y sin vida. ...

- **Isolation Forest** [41]: It learns a set of tree-based models and calculates an isolation score for every data point (text document). The average distance from the tree's root to the leaf associated with the data point, corresponding to the number of splits required to reach the data point, is used to compute the returned anomaly score. Shorter paths in the tree are indicative that the data point being analyzed is an anomaly (machine-generated texts) with respect to the data observed during training (human texts). An anomaly score close to 1 signifies that a data point has a high chance to be an anomaly (machine-generated text), whereas scores smaller than 0.5 are more likely attributed to normal data points (human texts).
- **Angle-Base Outlier Detection (ABOD) [42], [43]:** is a popular method for one-class-learning based on the variance of weighted cosine scores computed between each data point (text document) and its neighbors, which is considered as the anomaly score. To reduce the frequency of false positives, ABOD effectively identifies relationships in high-dimensional spaces between each data point and its neighbors.

- **Histogram-based Outlier Score (HBOS) [44]:** An intuitive statistical approach for one-class learning that assumes feature independence. HBOS generates a histogram for each data feature in two different modalities: *i)* static bin sizes and a preset bin width, and *ii)* dynamic bins with a close-to-equal number of bins. Subsequently, for each data point, it multiplies the inverse height of its bins, assessing the density of all features. This behavior is inspired by the Naive Bayes approach to classification, which is known for multiplying conditional probabilities. Even if feature independence neglects feature relationships, it facilitates a quick convergence of the method.

C. Metrics

We adopt common performance metrics used in machine learning-based classification problems, such as Precision (P), Recall (R), and F-Measure ($F1$), defined as:

$$P = \frac{T_p}{T_p + F_p}; \quad R = \frac{T_p}{T_p + F_n}; \quad F1 = 2 \times \frac{P \times R}{P + R},$$

where T_p is the number of true positives, F_p is the number of false positives, T_n is the number of true negatives, and F_n is the number of false negatives. Since our evaluation involves testing sets with balanced classes, metric values are the same for their Micro, Macro, and Weighted variants.

V. Results and discussion

In this section, we describe our experimental results. Results are extracted via 5-fold stratified Cross Validation for both datasets. Results for the L2 English dataset are reported in Table III, and results for the L2 Spanish dataset are reported in Table IV.

RQ1: We analyzed the efficacy of specific sets of linguistic features in the detection task. In the **English dataset**, the Text setting's F1-Score spans from 0.3324 (ABOD) to 0.6694 (OneClassSVM). The exploitation of **Repetitiveness features results in the highest performance** among the different sets, achieving F1-Scores from 0.4651 (ABOD) to 0.9027 (HBOS). In the Emotional Semantics setting, F1-Score extends from 0.3657 (HBOS) to 0.6436 (OneClassSVM), making this the lowest-performing linguistic feature. Readability features' F1-Scores range from 0.3569 (OneClassSVM) to 0.8942 (IsolationForest), which we identified as the second highest-performing linguistic feature. Finally, the POS setting yields F1-Scores from 0.3800 (HBOS) to 0.7223 (OneClassSVM). Overall, in the English dataset, linguistic features provided satisfactory detection capabilities.

In the **Spanish dataset**, the Text setting's F1-Score ranges from 0.3132 (IsolationForest) to 0.5917 (OneClassSVM), which is close to the highest score achieved in the Readability setting. In the Repetitiveness setting, F1-Score extends from 0.3301 (ABOD) to 0.5199 (OneClassSVM). In Emotional Semantics, the performance, in terms of F1-Score, ranges from 0.3496 (HBOS) to 0.4881 (OneClassSVM), resulting in the lowest-performing setting across all linguistic features. **Readability** features yield F1-Scores from 0.3301 (HBOS) to 0.5946 (LocalOutlierFactor), making this the **highest-performing setting** among the different sets. Finally, the POS setting provides F1-Score ranging from 0.3247 (IsolationForest) to 0.4968 (OneClassSVM). Overall, linguistic features were substantially ineffective in the detection task in Spanish, as noted by the weak performance of all baseline models. The fact that Repetitiveness is the highest-performing feature in English, and the third-best in Spanish is indicative of one of the main characteristics of machine-generated text [45]. English texts displayed repetitive terms and phrases.

RQ2: The proposed one-class deep fusion model is compared with one-class learning baseline models. Results for the English dataset highlight that **One-GPT outperforms all one-class learning model baselines** by achieving F1-Score values of 0.9478 in the English dataset and 0.6357 in the Spanish dataset. For the English dataset, the most competitive baseline is HBOS using Repetitiveness features (F1-Score: 0.9027). In the Spanish dataset, the second-best method is OneClassSVM using Text features (F1-Score: 0.5917). These findings illustrate the superiority of our model architecture, which allowed our method to extract salient and discriminating features for the detection task, leading to a more robust performance when compared to simpler approaches. **Confusion matrices** in Figures 3 and 4 show that the difference between models regarding the number of misclassified texts can be particularly substantial. The second diagonal in the English dataset highlights 35 incorrectly classified text documents for the proposed one-class fusion model, and 65 for HBOS, the most competitive baseline. In the Spanish dataset, we observe a larger number of misclassified texts: 255 incorrectly classified text documents for the proposed one-class fusion model. However, this number is still relatively lower than the most competitive baseline, OneClassSVM, which misclassified 277 text documents.

RQ3: The results of the One-class learning model are compared for each language. Significant disparities in model performance are evident when comparing English and Spanish. F1-Score comparisons are as follows: Text (0.6694 vs. 0.5917), Repetitiveness (0.9417 vs. 0.5199), Emotional Semantics (0.6436 vs. 0.4881), Readability (0.8942 vs. 0.5946), POS (0.7223 vs. 0.4968). **The highest detection accuracy is found in the English dataset**, with readability being the most optimal linguistic feature for both languages. Emotional Semantics is the linguistic feature with the lowest performing scores for both languages. We attribute lower F1-Scores in Spanish to two main reasons: First, English syntax (i.e. word order) presents a relatively fixed pattern compared to that of Spanish [46]. While both languages can be described as having an SVO (Subject Verb Object) order, Spanish is less rigid, presenting alternative VSO or OVS configurations. It is possible that our Doc2Vec PV-DM model could not create the appropriate connections between words due to their different places within sentences. Second, writers' proficiency levels in the Spanish dataset were more diverse than in the English dataset. This difference in proficiency challenges the capabil-

TABLE III
STRATIFIED 5-FOLD CROSS VALIDATION RESULTS WITH BOTH ANALYZED DATASETS IN TERMS OF PRECISION, RECALL, AND F1-SCORE.

English Dataset			
Setting = Text	Precision	Recall	F1-Score
OneClassSVM	0.7607	0.6915	**0.6694**
LocalOutlierFactor	0.5169	0.5052	0.4084
IsolationForest	0.6465	0.6095	0.5836
ABOD	0.3312	0.4933	0.3324
HBOS	0.5978	0.5067	0.3587
Setting = Repetitiveness	Precision	Recall	F1-Score
OneClassSVM	0.6141	0.6095	0.6054
LocalOutlierFactor	0.6764	0.6051	0.5611
IsolationForest	0.9473	0.9419	0.9417
ABOD	0.7268	0.5618	0.4651
HBOS	0.9105	0.9031	**0.9027**
Setting = Emotional Semantics	Precision	Recall	F1-Score
OneClassSVM	0.6699	0.6528	**0.6436**
LocalOutlierFactor	0.6085	0.5648	0.5168
IsolationForest	0.6516	0.6110	0.5835
ABOD	0.6968	0.5350	0.4155
HBOS	0.5674	0.5067	0.3657
Setting = Readability	Precision	Recall	F1-Score
OneClassSVM	0.7526	0.5112	0.3569
LocalOutlierFactor	0.8166	0.7675	0.7582
IsolationForest	0.8943	0.8942	**0.8942**
ABOD	0.7581	0.6006	0.5291
HBOS	0.8072	0.7273	0.7085
Setting = POS	Precision	Recall	F1-Score
OneClassSVM	0.8154	0.7392	**0.7223**
LocalOutlierFactor	0.7457	0.6080	0.5446
IsolationForest	0.6678	0.5887	0.5342
ABOD	0.7326	0.5693	0.4782
HBOS	0.5932	0.5127	0.3800
One-GPT (Proposed)	0.9478	0.9478	**0.9478**

TABLE IV
STRATIFIED 5-FOLD CROSS VALIDATION RESULTS WITH BOTH ANALYZED DATASETS IN TERMS OF PRECISION, RECALL, AND F1-SCORE.

Spanish Dataset			
Setting = Text	Precision	Recall	F1-Score
OneClassSVM	0.6190	0.6043	**0.5917**
LocalOutlierFactor	0.3641	0.4600	0.3443
IsolationForest	0.2511	0.4514	0.3132
ABOD	0.2489	0.4957	0.3314
HBOS	0.2475	0.4900	0.3289
Setting = Repetitiveness	Precision	Recall	F1-Score
OneClassSVM	0.5200	0.5200	**0.5199**
LocalOutlierFactor	0.5369	0.5171	0.4426
IsolationForest	0.3971	0.4886	0.3424
ABOD	0.2482	0.4929	0.3301
HBOS	0.2486	0.4943	0.3308
Setting = Emotional Semantics	Precision	Recall	F1-Score
OneClassSVM	0.4898	0.4900	**0.4881**
LocalOutlierFactor	0.5305	0.5186	0.4663
IsolationForest	0.4859	0.4900	0.4499
ABOD	0.5947	0.5100	0.3689
HBOS	0.4830	0.4986	0.3496
Setting = Readability	Precision	Recall	F1-Score
OneClassSVM	0.7514	0.5057	0.3459
LocalOutlierFactor	0.7134	0.6329	**0.5946**
IsolationForest	0.6904	0.6186	0.5788
ABOD	0.2486	0.4943	0.3308
HBOS	0.2482	0.4929	0.3301
Setting = POS	Precision	Recall	F1-Score
OneClassSVM	0.4971	0.4971	**0.4968**
LocalOutlierFactor	0.5200	0.5029	0.3673
IsolationForest	0.2701	0.4757	0.3247
ABOD	0.2478	0.4914	0.3295
HBOS	0.2482	0.4929	0.3301
One-GPT (Proposed)	0.6357	0.6357	**0.6357**

ities of models, for accurate patterns are more challenging to identify.

VI. CONCLUSIONS AND FUTURE WORK

Machine-generated text detection is a topic of growing significance in the wake of the widespread availability of AI tools, such as ChatGPT, to the general public. In this paper, we presented a one-class deep fusion model that employs a combination of a one-class model, textual, and linguistic features to accurately detect machine-generated text. While satisfactory, our findings show a notable bias in detection models, with English language data being more accurately classified. The practical implication of this research is to provide a tool to assist researchers and practitioners in deeper analyses of text data, considering a diversified perspective brought by the integration of linguistic features and neural vector embeddings. We advocate that detection models should not be used as decisive, punitive proof of the use of generative language models. Our model's capabilities could be fruitfully used in a variety of analytical tasks including fake news detection and content moderation on social media platforms. Future research could explore detection models in different languages, with native and non-native data, to ensure the fair and effective application of AI detection systems. Additionally, incorporating novel linguistic features could enhance existing models and drive the development of innovative algorithms, which can be suited to specific goals. It is crucial to approach the utilization of AI detection tools with a cautious mindset. Language is an ever-changing construct; thus, critical evaluation, continuous monitoring, and efforts to mitigate biases are essential components to balance the potential benefits and risks of detection methods in written communication.

ACKNOWLEDGMENTS

We acknowledge the support of American University and of NVIDIA through the donation of a Titan V GPU.

REFERENCES

[1] A. G. Bleumink and A. Shikhule, "Keeping ai honest in education: Identifying gpt-generated text," *Edukado AI Research*, pp. 1–5, 2023.

[2] R. Corizzo and S. Leal-Arenas, "A deep fusion model for human $vs.$ machine-generated essay classification," in *2023 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2023, pp. 1–10.

[3] D. R. Cotton, P. A. Cotton, and J. R. Shipway, "Chatting and cheating: Ensuring academic integrity in the era of chatgpt," *Innovations in Education and Teaching International*, pp. 1–12, 2023.

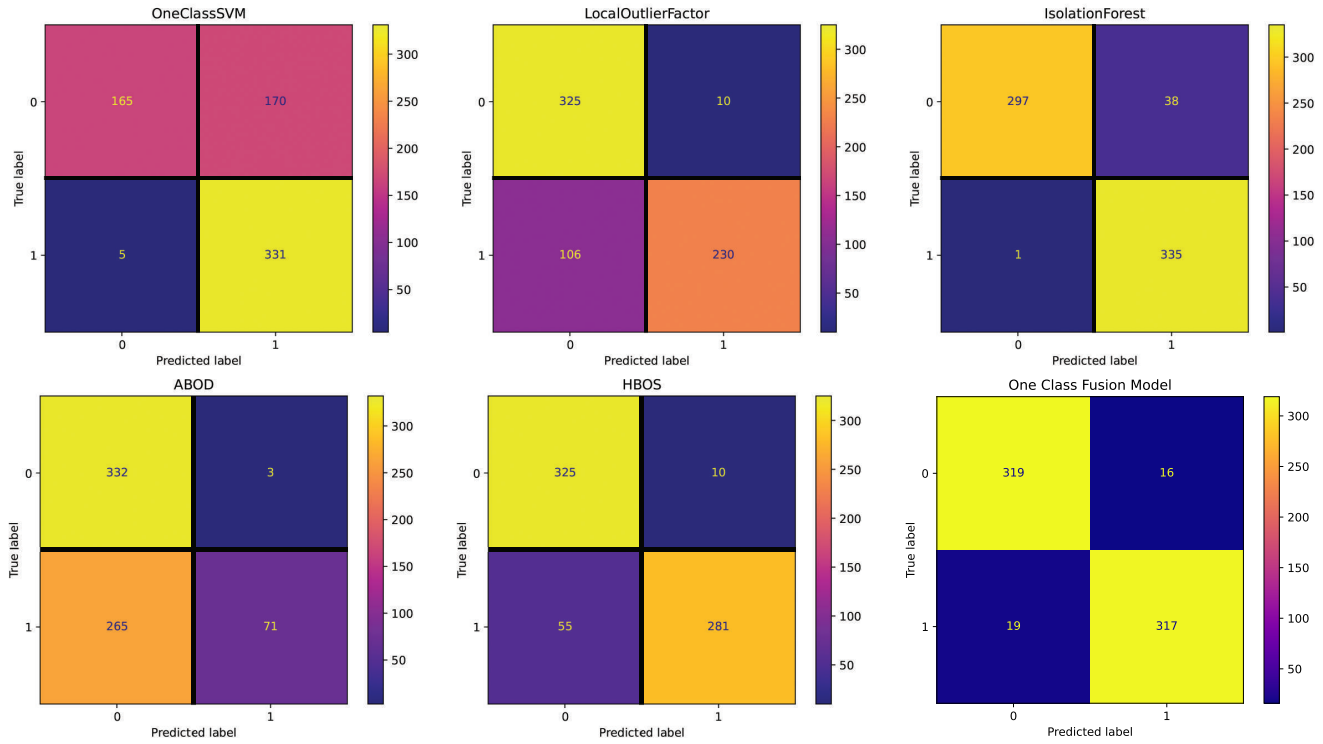

Fig. 3. Confusion matrices reporting the number of correctly (main diagonal) and incorrectly (secondary diagonal) predicted text documents for all methods (English dataset).

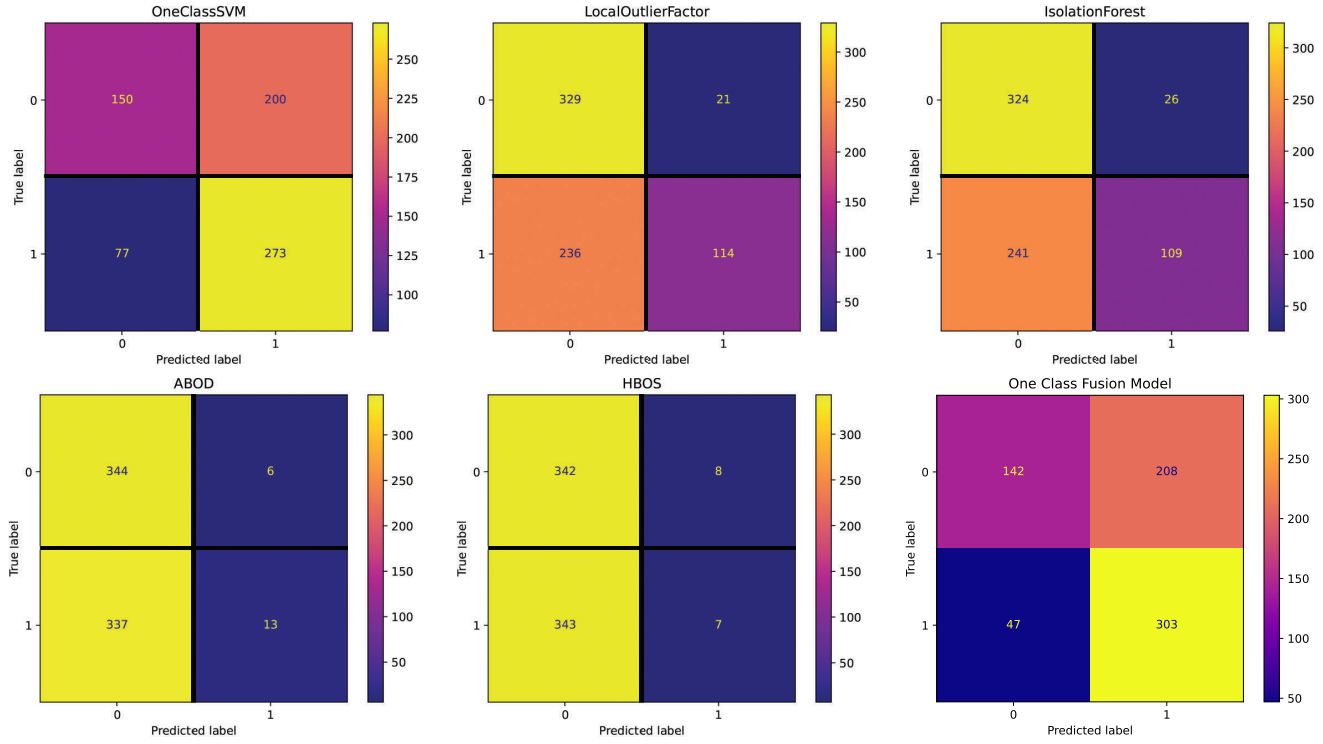

Fig. 4. Confusion matrices reporting the number of correctly (main diagonal) and incorrectly (secondary diagonal) predicted text documents for all methods (Spanish dataset).

[4] R. Corizzo and S. Leal-Arenas, "One-class learning for ai-generated essay detection," *Applied Sciences*, vol. 13, no. 13, p. 7901, 2023.

[5] M. Arbane, R. Benlamri, Y. Brik, and A. D. Alahmar, "Social media-based covid-19 sentiment classification model using bi-lstm," *Expert Systems with Applications*, vol. 212, p. 118710, 2023.

[6] N. Prasad, S. Saha, and P. Bhattacharyya, "A multimodal classification of noisy hate speech using character level embedding and attention," in *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2021, pp. 1–8.

[7] L. F. Gutierrez, F. Abri, M. Armstrong, A. S. Namin, and K. S. Jones, "Email embeddings for phishing detection," in *2020 ieee international conference on big data (big data)*. IEEE, 2020, pp. 2087–2092.

[8] S. Ouni, F. Fkih, and M. N. Omri, "Bert-and cnn-based tobeat approach for unwelcome tweets detection," *Social Network Analysis and Mining*, vol. 12, no. 1, p. 144, 2022.

[9] Q. G. To, K. G. To, V.-A. N. Huynh, N. T. Nguyen, D. T. Ngo, S. J. Alley, A. N. Tran, A. N. Tran, N. T. Pham, T. X. Bui *et al.*, "Applying machine learning to identify anti-vaccination tweets during the covid-19 pandemic," *International journal of environmental research and public health*, vol. 18, no. 8, p. 4069, 2021.

[10] R. Zellers, A. Holtzman, H. Rashkin, Y. Bisk, A. Farhadi, F. Roesner, and Y. Choi, "Defending against neural fake news," *Advances in neural information processing systems*, vol. 32, 2019.

[11] R. Corizzo, E. Zdravevski, M. Russell, A. Vagliano, and N. Japkowicz, "Feature extraction based on word embedding models for intrusion detection in network traffic," *Journal of Surveillance, Security and Safety*, vol. 1, no. 2, pp. 140–150, 2020.

[12] A. Bakhtin, S. Gross, M. Ott, Y. Deng, M. Ranzato, and A. Szlam, "Real or fake? learning to discriminate machine from human generated text," *arXiv preprint arXiv:1906.03351*, 2019.

[13] V. L. Rubin, N. Conroy, Y. Chen, and S. Cornwell, "Fake news or truth? using satirical cues to detect potentially misleading news," in *Proceedings of the second workshop on computational approaches to deception detection*, 2016, pp. 7–17.

[14] V. Pérez-Rosas, B. Kleinberg, A. Lefevre, and R. Mihalcea, "Automatic detection of fake news," *arXiv preprint arXiv:1708.07104*, 2017.

[15] A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi, "The curious case of neural text degeneration," in *International Conference on Learning Representations*.

[16] D. Ippolito, D. Duckworth, C. Callison-Burch, and D. Eck, "Automatic detection of generated text is easiest when humans are fooled," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 1808–1822.

[17] S. Gehrmann, S. Harvard, H. Strobelt, and A. M. Rush, "Gltr: Statistical detection and visualization of generated text," *ACL 2019*, p. 111, 2019.

[18] L. Fröhling and A. Zubiaga, "Feature-based detection of automated language models: tackling gpt-2, gpt-3 and grover," *PeerJ Computer Science*, vol. 7, p. e443, 2021.

[19] K. Faber, R. Corizzo, B. Sniezynski, and N. Japkowicz, "Active lifelong anomaly detection with experience replay," in *2022 IEEE 9th International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, 2022, pp. 1–10.

[20] Y. Lian, Y. Geng, and T. Tian, "Anomaly detection method for multivariate time series data of oil and gas stations based on digital twin and mtad-gan," *Applied Sciences*, vol. 13, no. 3, p. 1891, 2023.

[21] R. Corizzo, M. Ceci, G. Pio, P. Mignone, and N. Japkowicz, "Spatially-aware autoencoders for detecting contextual anomalies in geo-distributed data," in *Discovery Science: 24th International Conference, DS 2021, Halifax, NS, Canada, October 11–13, 2021, Proceedings 24*. Springer, 2021, pp. 461–471.

[22] J. Herskind Sejr, T. Christiansen, N. Dvinge, D. Hougesen, P. Schneider-Kamp, and A. Zimek, "Outlier detection with explanations on music streaming data: A case study with danmark music group ltd." *Applied Sciences*, vol. 11, no. 5, p. 2270, 2021.

[23] J. Kaufmann, K. Asalone, R. Corizzo, C. Saldanha, J. Bracht, and N. Japkowicz, "One-class ensembles for rare genomic sequences identification," in *Discovery Science: 23rd International Conference, DS 2020, Thessaloniki, Greece, October 19–21, 2020, Proceedings 23*. Springer, 2020, pp. 340–354.

[24] L. Feng, M. Jansche, M. Huenerfauth, and N. Elhadad, "A comparison of features for automatic readability assessment," 2010.

[25] S. Argamon-Engelson, M. Koppel, and G. Avneri, "Style-based text categorization: What newspaper am i reading," in *Proc. of the AAAI Workshop on Text Categorization*, 1998, pp. 1–4.

[26] M. Koppel, S. Argamon, and A. R. Shimoni, "Automatically categorizing written texts by author gender," *Literary and linguistic computing*, vol. 17, no. 4, pp. 401–412, 2002.

[27] R. Tang, Y.-N. Chuang, and X. Hu, "The science of detecting llm-generated texts," *arXiv preprint arXiv:2303.07205*, 2023.

[28] E. Mitchell, Y. Lee, A. Khazatsky, C. D. Manning, and C. Finn, "Detectgpt: Zero-shot machine-generated text detection using probability curvature," *arXiv preprint arXiv:2301.11305*, 2023.

[29] X. Yang, W. Cheng, L. Petzold, W. Y. Wang, and H. Chen, "Dna-gpt: Divergent n-gram analysis for training-free detection of gpt-generated text," *arXiv preprint arXiv:2305.17359*, 2023.

[30] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *International conference on machine learning*, 2014, pp. 1188–1196.

[31] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[32] R. Baly, G. Karadzhov, D. Alexandrov, J. Glass, and P. Nakov, "Predicting factuality of reporting and bias of news media sources," *arXiv preprint arXiv:1810.01765*, 2018.

[33] B. D. Horne and S. Adali, "This just in: Fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news," in *Eleventh international AAAI conference on web and social media*, 2017.

[34] A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi, "The curious case of neural text degeneration," *arXiv preprint arXiv:1904.09751*, 2019.

[35] Z. Boukouvalas, C. Mallinson, E. Crothers, N. Japkowicz, A. Piplai, S. Mittal, A. Joshi, and T. Adalı, "Independent component analysis for trustworthy cyberspace during high impact events: an application to covid-19," *arXiv preprint arXiv:2006.01284*, 2020.

[36] W. Wei-Ning, Y. Ying-Lin, and J. Sheng-Ming, "Image retrieval by emotional semantics: A study of emotional space and feature extraction," in *2006 IEEE International Conference on Systems, Man and Cybernetics*, vol. 4. IEEE, 2006, pp. 3534–3539.

[37] S. Vosoughi, D. Roy, and S. Aral, "The spread of true and false news online," *science*, vol. 359, no. 6380, pp. 1146–1151, 2018.

[38] S. Loria *et al.*, "textblob documentation," *Release 0.15*, vol. 2, no. 8, 2018.

[39] B. Schölkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, and J. C. Platt, "Support vector method for novelty detection," in *Advances in neural information processing systems*, 2000, pp. 582–588.

[40] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "Lof: identifying density-based local outliers," in *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, 2000, pp. 93–104.

[41] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *2008 Eighth IEEE International Conference on Data Mining*. IEEE, 2008, pp. 413–422.

[42] H. Kriegel, M. Schubert, and A. Zimek, "Angle-based outlier detection in high-dimensional data," pp. 444–452, 2008.

[43] N. Pham and R. Pagh, "A near-linear time approximation algorithm for angle-based outlier detection in high-dimensional data," pp. 877–885, 2012.

[44] M. Goldstein and A. D. H.-b. O. Score, "A fast unsupervised anomaly detection algorithm," *KI-2012: Poster and Demo Track*, pp. 59–63.

[45] T. Zhu, "From textual experiments to experimental texts: Expressive repetition in" artificial intelligence literature"," *arXiv preprint arXiv:2201.02303*, 2022.

[46] A. K. Tickoo, *On preposing and word order rigidity*. University of Pennsylvania, 1990.