Can Your Classifier Detect Boundaries? Adaptation of Artificial Text Detection Methods for the Real Or Fake Text Challenge

Anonymous ACL submission

Abstract

Due to the rapid development of text generation models, people increasingly often encounter texts that may start out as written by a human but then continue as AI-generated. Detecting the boundary between human-written and machine-generated parts of such texts is a very challenging problem that has not received much attention in literature. We consider a number of different approaches for artificial text boundary detection, comparing predictors over features of different nature. We show that supervised fine-tuning of the RoBERTa model works well for in-domain detection of a single LLM but fails to generalize in important cross-domain and cross-generator settings, demonstrating a tendency to overfit to spurious features of the data. Then, we adapt perplexity-based approaches and propose novel algorithms based on features extracted from a frozen LLM's embeddings. We show that these approaches outperform the human accuracy level on an extremely hard Real or Fake Text benchmark. Analyzing the robustness of our approaches in cross-domain and cross-model settings, we discover important properties of the data that can hinder the performance of artificial text boundary detection algorithms.

1 Introduction

Artificial text detection (ATD) is a very difficult problem in real life, where machine-generated text may be intertwined with human-written text, lightly edited, or pad out human-generated prompts. However, in literature ATD is usually formulated in a simpler way, with a dataset of text samples labeled as either entirely human-written or entirely machine-written, so the detection problem can be safely treated as binary classification. Moreover, the models for this binary classification are often developed and trained to detect a particular type of generator, e.g., text produced by a specific large language model (LLM) (Uchendu et al., 2023a). This is in stark contrast with how we may encounter artificially created text in real life, where documents partially written by humans and partially generated by LLMs already abound. This setting is much more complex and much less researched.

To approach this problem, in this work we experiment with a lesser known dataset called RoFT (Real Or Fake Text), collected by Dugan et al. (2020). Each text in this dataset consists of ten sentences, where the first several sentences are human-written and the rest are machine-generated starting from this prompt, mainly by models from the GPT family (Radford et al., 2019; Brown et al., 2020). We consider several techniques developed for binary ATD, modifying them for this more complex boundary detection setting; e.g., following Tulchinskii et al. (2023a) we adapt intrinsic dimension estimation which is currently considered to be the best method for binary detection.

Our primary contributions are as follows: (1) we develop new approaches for detecting the boundary between human-written and machine-generated text and show how time series analysis can be adapted for extracting useful information from Transformer representations; our approaches are mostly based on RoBERTa latent representations; (2) we adapt perplexity-based methods for this new setting, discussing the differences in their behavior compared to binary ATD and providing a comprehensive analysis of how perplexity scores react to the machine–human transition in the text; (3) we analyze the robustness of boundary detectors (including the fully-tuned RoBERTa baseline) to domain shift and show how it depends on the properties of the domains; we study the properties of the dataset itself and their effect on the performance of various approaches; (4) we enrich the RoFT dataset with *GPT-3.5-turbo*[1] (ChatGPT) generation sam-

[1] https://platform.openai.com/docs/model-index-for-researchers

ples; we share this new dataset with the community, establish baselines, and analyze the behavior of our detectors on it.

We hope that the present work will encourage further research in both directions: first, boundary detection for texts that are partially human-written and partially generated, and second, analyzing how inner representations of Transformer-based models react to such transitions; the latter direction may also help interpretability research.

The rest of the paper is organized as follows. In Section 2 we survey related work, and Section 3 introduces the methods we have applied for artificial text boundary detection. Section 4 presents a comprehensive evaluation study on the RoFT and RoFT-chatgpt datasets. Section 5 presents a detailed discussion and analysis of our experimental results, and Section 6 concludes the paper.

2 Related Work

Recent surveys by Uchendu et al. (2023a) and Yang et al. (2023) discuss a number of different ATD settings and the main types of detectors, so here we concentrate only on a few specific approaches that are most relevant to our research.

ATD with Topological Data Analysis. In ATD methods, apart from standard approaches we take special inspiration from the results of Tulchinskii et al. (2023a). Their work shows that many artificial text generators, including GPT-2-XL, ChatGPT, and gpt-3.5-davinci, share a common property: the intrinsic dimensions (PHD) of RoBERTa embeddings of the texts created by these generators are typically smaller than those of the texts written by humans. Hence, Tulchinskii et al. (2023a) suggested intrinsic dimensions as high-performing features for ATD, a direction that we explore further in this work. Kushnareva et al. (2021) for the first time introduced topological features of inner representations of a Transformer-based model (BERT in that case) for ATD. Uchendu et al. (2023b) integrated topological features into an ATD pipeline as a "TDA layer". Topological data analysis has also proven to be useful for closely related tasks of fake news detection (Tudoreanu), authorship attribution (Elyasi and Moghadam, 2019), and detection of synthetic speech (Tulchinskii et al., 2023b).

Style Change Detection and Authorship Attribution. ATD can also be considered as a special case of authorship attribution (AA), where the LLM is one author and the human is another, and the task is to determine the authorship of different parts of the text. Jones et al. (2022) show that LLM models can successfully imitate human style and deceive existing popular online AA methods. However, Venkatraman et al. (2023) propose an approach based on the principle of uniform information density that can detect the authorship of LLM. Artificial text boundary detection and AA tasks have a lot in common with the style change detection problem (Zangerle et al., 2021). In this setting, the model is analyzing multi-author documents (e.g., research papers), and the goal is to detect where the author writing it changes. In the challenge by Zangerle et al. (2021), documents were created by compiling answers from *StackExchange* Q&A threads into a single text. In some subtasks, style changes occur only between paragraphs; in others, they may occur at the level of individual sentences. The currently best style change detectors, including the works of Lin et al. (2022), Jiang et al. (2022), Lao et al. (2022), and Iyer and Vosoughi (2020), are based on Transformer-based encoders such as BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), AlBERT (Lan et al., 2020), and ELECTRA (Clark et al., 2020). In this work, we also use RoBERTa as a baseline and a source of embeddings.

RoFT (Real Or Fake Text). Our main dataset, introduced in Dugan et al. (2020) and further developed in Dugan et al. (2023), originates from a website called RoFT[2] developed as a tool to analyze how *humans* detect generated text, including an extended study of whether and how they can explain their choice, when they say that a text sample is machine-generated, and how humans can learn to recognize machine-generated text better. Every user of this website can choose the topic ("Short Stories", "Recipes", "New York Times", or "Presidential Speeches") and start the following "game". The player sees ten sentences one by one. The first sentence is always written by a human, but for each subsequent sentence the player must determine whether it is machine-generated or human-written. If the player believes the text was AI-generated, they should explain why they think so. If they guess machine generation before the true boundary, they earn zero points. Otherwise, they earn $5 - x$ points, where x is the number of sentences after the correct boundary. This is actually a harder problem than just boundary detection since the player does not see the full text, and the

[2] http://www.roft.io/

scoring function is skewed.

As a result of running the game, Dugan et al. (2023) created a dataset also known as RoFT. In this dataset, every sample consists of ten sentences: the first N sentences are human-written, and the other $10 - N$ sentences are generated by some language model that uses the first N sentences as a prompt. Every sample is accompanied by the following information: true value of N; the value $\hat{N}$ predicted by a player; the topic ("Short Stories", "Recipes", "New York Times", or "Presidential Speeches"); the generator model; explanation that the player provided for why they think that the $\hat{N}$th sentence is machine-generated; information about the player.

The original RoFT contains generations from GPT-2 (Radford et al., 2019), GPT-2 XL, GPT-2 finetuned on the "Recipes" domain, GPT-3.5 (davinci), CTRL (Keskar et al., 2019) with control code "nocode" and with control code "Politics", and baseline, where instead of an LLM-generated continuation the passage transitions to a completely different news article selected at random. In most studies, RoFT and similar datasets are used to investigate how humans detect artificially generated texts manually; e.g., Clark et al. (2021) evaluated several methods of improving the human ability to distinguish texts generated by GPT-2 and GPT-3 (i.e., training humans rather than neural networks).

Cutler et al. (2021) provided the first baselines on automatically solving RoFT. For this purpose, they used and compared several shallow classification and regression models based on RoBERTa and SRoBERTa (Reimers and Gurevych, 2019) embeddings collected from the last layer of these models. As a result, they found that logistic regression and random forests worked best when trained on particular domains. Aside from predicting true boundary labels, they also learned to predict human-predicted labels ("human-predicted boundary detection task").

However, the cross-domain and cross-model settings in their research were very limited. In fact, Cutler et al. (2021) call "Out of Domain" (OOD) classifiers that had been trained on *all* available data and then evaluated on a given subset, and they call "In Domain" (ID) classifiers that had been trained and evaluated on the same subset. Besides, they did not analyze all generators and domains within this cross-domain setting. In contrast, in our work we concentrate on cross-domain and cross-model settings and interpretability. We evaluate our boundary detectors on unseen generators (models) and topics (domains). Also, we do this for all models and domains, establishing new baselines for the RoFT dataset. A similar problem was addressed by Zeng et al. (2023) who used a TriBERT-based approach for artificial text boundary detection in student essays in the field of education.

3 Approach

We consider several different approaches, including a multilabel classification framework, as proposed by Cutler et al. (2021), where the label of a text corresponds to the number of the first generated sentence, time series analysis that slides a window over the text tokens, and regression methods that minimize the MSE between true and predicted boundaries. We design our classifiers based on features that have been successfully used in prior works on ATD (Solaiman et al., 2019; Mitchell et al., 2023). We also introduce a new baseline based on sentence lengths. Below, we consider all of these approaches in detail.

RoBERTa classifier. Unlike Cutler et al. (2021) who process each sentence separately, we fine-tune the RoBERTa model (Liu et al., 2019) to represent the entire text sample via the [CLS] vector. This is the only case among our proposed models where we apply full model fine-tuning. For other methods, we use simpler classifiers such as logistic regression or gradient boosting (Friedman, 2001) trained on various features extracted directly from larger trained models (e.g., Transformer-based LLMs), with no updates to the larger model's weights.

Perplexity-based features. We use a single unified model for all data to compute sentence-wise perplexity, making our approach more practical than that of Cutler et al. (2021) who used perplexity scores from the original model used to generate artificial text. Note that using the same model is not only harder in practice but may also be infeasible if the generator is unknown, in particular in cross-model scenarios.

Perplexity from GPT-2. We calculate the log likelihood of each word and each sentence via GPT-2 (Radford et al., 2019). We then train a classifier using these vectors of sentence likelihoods as features. Our underlying hypothesis here is that texts generated by similar models such as GPT-2 or GPT-3.5 might appear more "natural" to these models, as reflected by their likelihood scores. Our findings corroborate this assumption (see Appendix E).

DetectGPT. The DetectGPT framework

(Mitchell et al., 2023) proposes a more nuanced perplexity-based scoring function. It involves perturbing a text passage and comparing log-probabilities between original and altered texts. This score then serves as an input to a classification model, with GPT-2 as the base model and T5-Large (Raffel et al., 2020) generating the perturbations.

Length-Based Baseline. Since we have observed a statistical difference in sentence length distributions between human-written and generated texts (see Fig. 2), we leverage sentence lengths as a simple baseline feature. This baseline allows us to gauge the effectiveness of a classifier in identifying boundaries without semantic understanding.

Perplexity Regression. We use a gradient boosting regressor trained on sentence-wise log likelihood features to predict boundary values; note that this regression-based formulation takes advantage of the task's sequential nature and aims to minimize label discrepancies rather than necessarily predict the exact first artificial sentence.

Topological Time Series (TTS). Inspired by Tulchinskii et al. (2023a), we explore the potential of topological features based on intrinsic dimensionality (ID); we provide an introduction to topological data analysis (TDA), including the definitions of features, in Appendix A. We hypothesize that geometric variations in token sequences can help identify machine-generated text, so we introduce models that process TDA-based features treating them as time series. For every text, we slide a window of $H = 20$ tokens (step size $S = 5$) over RoBERTa token embeddings and find the intrinsic dimension (PHD) of the points within the window, as shown in Fig. 1 (Schweinhart, 2020). The time series are then classified with an SVM with the global alignment kernel (GAK) (Cuturi, 2011); this method is called "PHD + TS ML" in the tables.

Boundary detection via binary classification. In this approach, we train a binary classifier to distinguish between *fake* and *natural* text atop a specific predictor. For the base predictor, we employ intrinsic dimensionality calculated over a sliding window of 20 tokens. The TLE (tight local) intrinsic dimension estimator was chosen due to its robust performance on small data samples (Amsaleg et al., 2019). For the base classifier, we use gradient boosting trees (Friedman, 2000, 2002). To translate probabilities predicted with a binary classifier into a specific boundary that separates real and fake text, we determine the final label by maximizing $\mathbf{y} \mapsto \arg\max_{I \in \mathcal{I}} s_I(\mathbf{y}, \mathbf{x})$, where the score function s_I is defined as $s_I(\mathbf{y}, \mathbf{x}) = -\sum_j^n I_{\mathbf{y}}^j \log \hat{p}(\mathbf{x}^j)$, and $\hat{p}(\mathbf{x})$ are predictions from the base binary classifier. In this context, we use a binary indicator set $I \in \mathbf{1}^n$ consisting of a chunk of k zeros followed by another chunk of $n - k$ ones. This method is called "TLE + TS Binary" in the tables.

4 Experimental evaluation

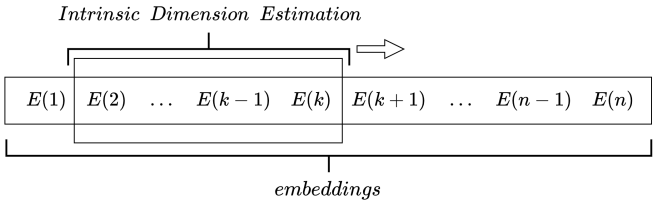
4.1 Dataset preparation and analysis

In all experiments, the task is to detect the exact boundary where a text passage that starts as human-written transits to machine generation. In addition to RoFT (Dugan et al., 2020), we created its new version called **RoFT-chatgpt**, where the same human prompts are continued with the *gpt-3.5-turbo* model. RoFT-chatgpt is supposed to be more challenging for artificial text boundary detection while preserving the basic statistical properties of the original such as the label distribution. During preprocessing, we removed duplicates that were different only in human-predicted labels from both datasets; for *RoFT-chatgpt*, we also removed all samples containing "As an AI language model..." and short samples that were clearly failed generations. As a result, we retained 8943 samples from the original RoFT and 6940 samples from *RoFT-chatgpt*. Preliminary statistical analysis of the datasets revealed that the distributions of sentence lengths in different subdomains vary significantly, as shown in Fig. 2 and Figs. 6, 7 in Appendix D (see also a discussion in Section 5).

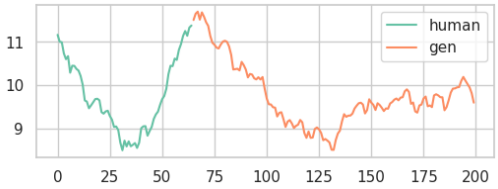
4.2 Results

Table 1 presents the main results of our experiments on artificial text boundary detection on RoFT and RoFT-chatgpt datasets. The simplest baseline in Table 1 is the majority class prediction, which is the last (9th) class of fully human-written texts. On the original RoFT, we also include the human baseline and the best result reported by Cutler et al. (2021), obtained by a classifier built upon concatenated SRoBERTa embeddings of each sentence.

Apart from accuracy (*Acc*), we report two other metrics reflecting the sequential nature of predicted labels: soft accuracy, i.e., the percentage of predictions that differ from the correct label by at most one (*SoftAcc1*), and mean squared error (MSE). SoftAcc1 is interesting because a large part of misclassifications tends to be on the neighboring class (see Fig. 18 in Appendix E), and a slight relaxation of the exact classification seems to be acceptable



(a) Sliding window technique



(b) Sample result

Figure 1: Multilabel time series classification: (a) estimating the intrinsic dimension of token embeddings in a sliding window, (b) a sample resulting series: green – human-written, orange – machine-generated tokens.

Table 1: Boundary detection results. **Bold** shows the best method; underlined, second best, *italic*, values outperforming the human baseline.

Method	RoFT			RoFT-chatgpt		
	Acc	SoftAcc1	MSE	Acc	SoftAcc1	MSE
RoBERTa + SEP	**49.64 %**	**79.71 %**	**02.63**	**54.61 %**	**79.03 %**	**03.06**
RoBERTa	46.47 %	74.86 %	03.00	39.01 %	75.18 %	03.15
Perplexity-based classifiers						
Perplexity + GB	*24.25 %*	*47.23 %*	*11.68*	34.94 %	59.80 %	07.46
Perplexity + LogRegr	*23.75 %*	*42.15 %*	15.80	33.50 %	57.56 %	09.25
DetectGPT + GB	19.79 %	37.40 %	*08.35*	21.69 %	43.52 %	06.87
DetectGPT + LogRegr	19.45 %	33.82 %	*09.03*	15.35 %	41.43 %	07.22
Perplexity-based regression	12.58 %	36.67 %	*06.89*	19.74 %	54.03 %	04.89
Topological Time Series						
PHD + TS ML	*23.50 %*	*46.32 %*	14.14	17.29 %	35.81 %	14.45
TLE + TS Binary	12.58 %	30.41 %	22.23	20.02 %	34.58 %	18.52
Length + GB	14.64 %	33.43 %	16.55	25.72 %	46.18 %	18.99
Majority class	15.26 %	25.43 %	27.58	13.83 %	24.42 %	26.46
SRoBERTa (Cutler et al., 2021)	42 %	–	–		–	
Human Baseline	22.62 %	40.31 %	13.88		–	

in many real-world applications.

As for the results, we first note that on the original RoFT, RoBERTa-based classifiers outperform others by a factor of almost 2x in terms of accuracy metrics and also significantly outperform the previously best reported SRoBERTa model (Cutler et al., 2021). This model also provides the lowest MSE (0.03) among all the methods. We note, however, that our RoBERTa classifier has significantly more trainable parameters than any other method in the table because no other approaches require language model fine-tuning. Second, topological and perplexity features improve over the human baseline. Perplexity-based classifiers are the best in terms of accuracy, while the perplexity regressor provides good MSE values. Recall from Section 2, however, that humans were solving a harder problem with a somewhat different objective.

Third, RoBERTa's accuracy on *RoFT-chatgpt* drops by 6% compared to RoFT, while soft accuracy and MSE are roughly the same. Surprisingly, the opposite holds for perplexity-based methods: on *RoFT-chatgpt* the results are significantly *better* than for data generated by older LLMs. The reason for this might be that we used a GPT-like model (GPT-2) for perplexity calculation, and the GPT-3.5-turbo model that we used to generate fake samples in *RoFT-chatgpt* belongs to the same family. However, note that we used a smaller model (GPT-2) to detect text generated by a larger model (GPT-3.5) and got second-best results among other approaches, despite Mitchell et al. (2023) reporting that smaller models are not capable of detecting text generated by larger models. The baseline length-based classifier also improves its accuracy significantly (by 1.8x) when transferring to *RoFT-chatgpt*. We hypothesize that this kind of a shallow feature emerges in ChatGPT generation and makes the task easier (see also Section 5).

The other perplexity-based approach, DetectGPT, exhibits lower accuracy compared to RoBERTa and perplexity-based classifiers on both datasets. Mitchell et al. (2023) note that DetectGPT can detect whether the sample is generated by

a specific base model, but we use several models in our setup, and text samples may be too short this approach. On the other hand, DetectGPT has quite good MSE values, close to the regression approach designed to optimize this value directly. Nevertheless, we exclude this method from further experiments for the sake of the balance of quality and computational complexity.

4.3 Cross-domain generalization

Supervised ATD methods with fine-tuning such as RoBERTa are more sensitive to spurious correlations in a dataset and often demonstrate poor cross-domain transfer, especially in comparison to topology-based approaches (Tulchinskii et al., 2023a). Table 2 reports the results of cross-domain transfer between four text topics presented in the *RoFT-chatgpt* dataset. We report in-domain and out-of-domain accuracy: the IN column shows results from domains seen during training, while OUT shows results for the unseen domain corresponding to this column. MSE scores are reported in Table 3 in Appendix B. For each model, training was done on three domains, and the resulting model was tested on the fourth, unseen domain; we used 60% of these subsets mixed together, as the training set, 20% as the validation set, and 20% as the test set for in-domain evaluation.

First, note how RoBERTa's performance drops for all subsets with a very significant change from 0.25 to 0.61. The perplexity-based classifier demonstrates excellent cross-domain generalization for *Presidential Speeches* and *Short Stories*, while for the *Recipes* domain TTS classifiers prove to be the most stable. For the *New York Times* subset, we observe a significant generalization gap for all classifiers. These results mean that each type of classifier can handle its own set of spurious features well, and no classifier is universally better than the others. We hypothesize that aggregation of different features can improve the results and leave this for future research. As for classification accuracy, surprisingly, on two subsets out of four the perplexity-based classifier outperforms a fully fine-tuned RoBERTa. Moreover, for *Recipes* the multilabel topological time series method is the best. This happens because TDA-based methods are extremely stable under domain shift on this dataset split despite being significantly worse than others in absolute values. We also note significant differences in the format of sentences in the *Recipes* domain compared to others (see Section 5).

In general, we can conclude that perplexity-based classifiers and RoBERTa perform roughly equally on average in the out-of-domain setting. This is a remarkable property, taking into account the training budget of these classifiers: the former is a simple classifier trained on ten features extracted by a language model with frozen weights, while the latter involves full LM fine-tuning.

Our final observation in this setting is related to the length-based baseline. For in-domain data, average sentence length provides a strong signal, leading to accuracy between 20% to 32% depending on the data split and even outperforming topological methods. But cross-domain generalization fails, which means that we should prefer the classifiers that ignore this feature in order to achieve good generalization (see more details in Section 5).

4.4 Cross-model generalization

The original RoFT dataset contains data generated by different models. Appendix C shows detailed experimental results for cross-generator generalization (Tables 4 and 5). We tune our classifiers on generation results produced by one model and test the performance for all other models. In general, this task is harder for all considered classifiers: there are models for which prediction accuracy drops down to virtually zero values. But we observe an interesting result for the perplexity-based classifier: it achieves good generalization when transferring to very large models such as *GPT3-davinci* and GPT2-XL, while for other models the generalization is poor. We provide possible explanations of such results in the next section.

5 Discussion and analysis of the results

In this section, we present some interesting conclusions that can be made from the data and model performance. First, the length of sentences seems to play an important role, deceiving our classifiers. We connect it with a significant difference between distributions of sentence lengths written by humans and generated by large language models (see Fig. 2 and Figs. 6, 7 in Appendix D). This is supported by our experiments with the length-based classifier, which sometimes outperforms other methods in terms of accuracy (Tables 1 and 2).

As for the data, we note interesting peculiarities in the *Recipes* topic: texts often contain the index of the current step in a recipe ("1.", "2." etc.) as a distinct sentence, which can be easily picked up

Table 2: Accuracy for leave-one-out cross-domain evaluation on *RoFT-chatgpt*. △ and ▽ show relative change from the model's *in-domain* score to the human score; ▲ and ▼ show relative change from the *out-of-domain* score to the *in-domain* score. *Green* highlights improvements, *red* indicates deteriorations.

Predictor	Model	Context	Pres. Speeches IN ↑	Pres. Speeches OUT ↑	Recipes IN ↑	Recipes OUT ↑	New York Times IN ↑	New York Times OUT ↑	Short Stories IN ↑	Short Stories OUT ↑	Avg Δ
Text	RoBERTa SEP	global	57.3△153%	31.4▼45%	43.2△91%	13.1▼70%	53.2△135%	38.1▼28%	54.3△140%	28.6▼47%	−48%
Text	RoBERTa	global	54.2△140%	40.2▼26%	39.7△75%	15.1▼62%	53.3△136%	34.0▼36%	50.3△122%	27.7▼45%	−42%
Perpl.	GB	sentence	36.1△60%	35.3▼02%	36.6△62%	19.4▼47%	36.9△63%	29.7▼20%	32.8△45%	32.9▲00%	−17%
Perpl.	LogRegr	sentence	34.8△54%	32.4▼07%	40.6△80%	20.4▼50%	36.7△62%	28.9▼21%	32.7△44%	35.6▲09%	-17%
Perpl.	Regr. (GB)	sentence	20.9▽08%	22.6▲08%	23.8△05%	14.4▼39%	18.5▽18%	16.1▼13%	21.2▽06%	21.8▲03%	-10%
PHD	TS multilabel	100 tokens	20.3▽10%	13.7▼32%	19.2▽15%	19.5▲02%	20.9▽08%	17.2▼18%	21.2▽06%	17.6▼17%	-16%
TLE	TS Binary	20 tokens	25.6△13%	14.7▼42%	16.5▽27%	16.3▼01%	25.0△11%	17.1▼32%	22.1▽02%	11.1▼50%	-31%
Length	GB	sentence	28.1△24%	11.8▼58%	21.1▽07%	15.5▼26%	30.4△34%	18.4▼39%	32.3△43%	15.8▼51%	-44%
Length	LogRegr	sentence	19.6▽14%	10.8▼45%	17.0▽25%	12.9▼24%	22.2▽02%	09.1▼59%	22.9△01%	09.1▼60%	-47%
Majority		—	15.4▽32%		13.0▽43%		15.9▽30%		17.7▽22%		—
Approximated human		global	22.62		22.62		22.62		22.62		

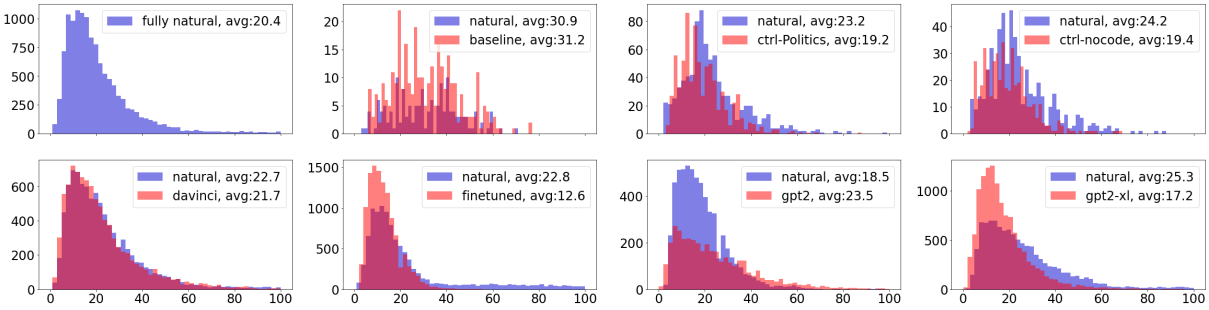


Figure 2: Sentence length distributions in RoBERTA tokens, original RoFT, by model

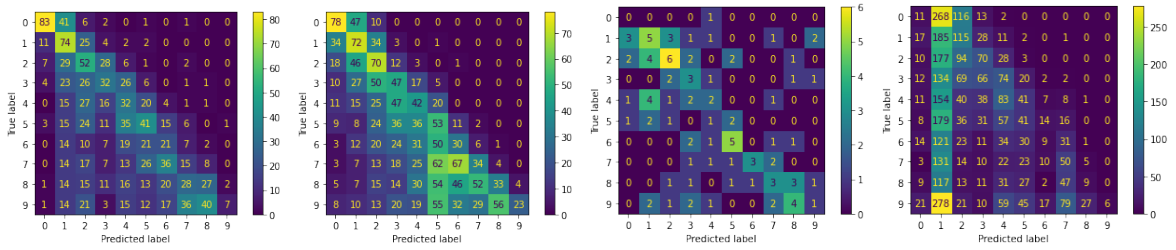


Figure 3: Confusion matrices of RoBERTa predictions on the four domains in the *RoFT-chatgpt* dataset.

as a domain feature, and the first sentence tends to be very long compared to others. This may explain worse performance when *Recipes* is used as the out-of-domain part. The RoBERTa classifier has the largest quality drop here, so we investigated its confusion matrices. Fig. 3 shows the anomaly on the *Recipes* domain: RoBERTa tends to predict the label "1" most of the time, while on other domains its predictions are distributed much more evenly and adequately. The only approach able to handle cross-domain transfer to *Recipes* is topological time series; we suggest that unlike other methods it is able to ignore sentence length variations.

Label distributions (see Figs. 9 and 10 in Appendix D) vary significantly across models. GPT-2 is especially different from natural texts, and the behavior of our models in cross-model transfer to GPT-2 (Appendix C), suggests that the only model stable under the label distribution shift is perplexity-based regression. Although its accuracy numbers are low, it outperforms the human level in terms of MSE even in out-of-domain evaluation.

Distributions of sentence perplexities across different generators are also interesting (see Fig. 4). Perplexity distributions for texts generated by *baseline* or GPT-2 are very different from the distributions of texts generated by other models. This might explain the poor performance of decision tree and linear classifiers when these generators are excluded from the training set: significant dis-

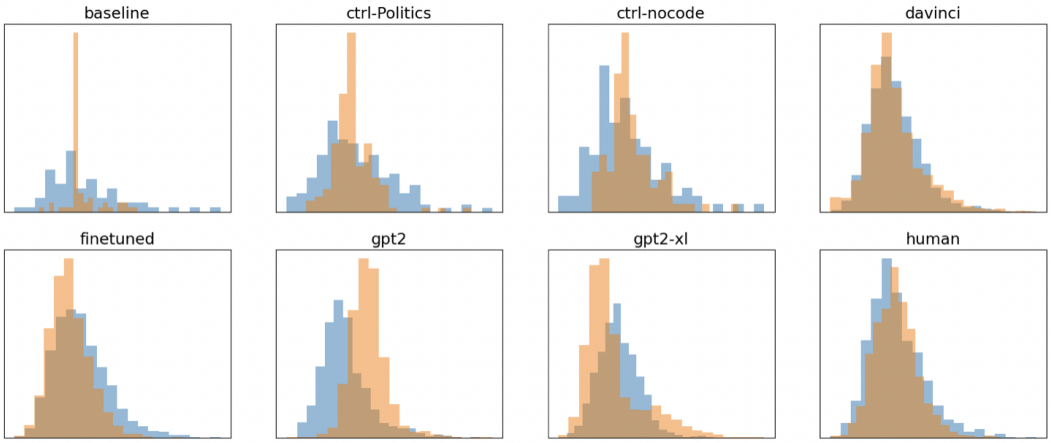


Figure 4: Perplexities of the last sentence in RoFT by model; *blue* — distribution on the in-domain set, i.e., the entire dataset except the speficied generator; *orange* — on the out-of-domain set, i.e., data from the specified generator.

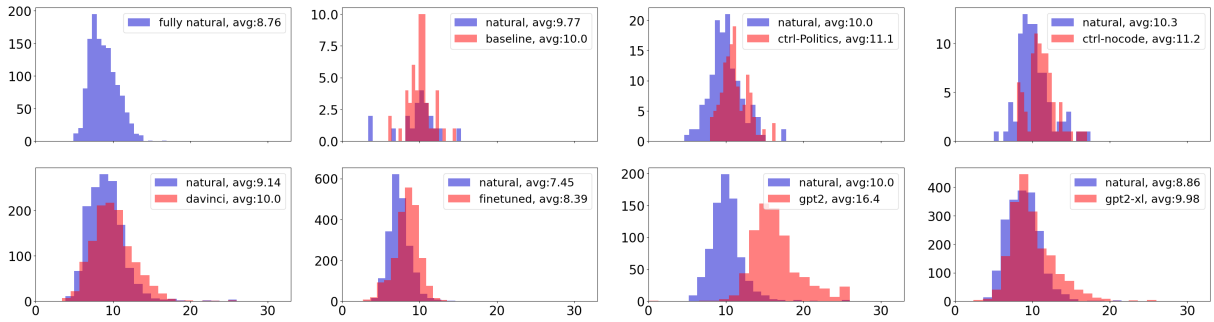


Figure 5: PHD distributions for the real and fake parts of the RoFT dataset by generator models

crepancies between feature distributions across domains may lead to poor cross-domain transfer in these cases. Perplexity-based logistic regression and gradient boosting classifiers over the same set of features show the opposite behavior in GPT2-XL subset transfer: the former fails to generalize, while the latter demonstrates an improvement in the quality metrics (see Appendix C). Moreover, distributions of sentence perplexities can vary highly depending on the generator. We believe that additional normalization and/or better choice of a model are needed to mitigate this.

Finally, we note that geometric properties of the embeddings for both RoFT and *RoFT-chatgpt* show the difference between PHD distributions of real and fake RoFT text for different models (Fig. 5) and topics (Figs. 11, 12 in Appendix D). TLE dimension distributions for sentences in human-written and AI-generated parts of the texts are different as well (Figs. 13, 14, 15, Appendix D).

6 Conclusion

In this work, we address the task of boundary detection between human-written and generated parts in texts that combine both. We believe that this setting is increasingly important in real world applications and is a natural setting for recognizing artificial text since it is often mixed with and prompted by human text.

We have considered the RoFT dataset, presented its modification *RoFT-chatgpt* generated with a more modern LLM, and investigated the performance of features that were useful for artificial text detection in previous works. In particular, we have shown that LLM fine-tuning works reasonably well for this task but tends to overfit to spurious features in the data, which leads to generalization failures in some settings. On the other hand, perplexity-based and topological features provide a signal that can help in these situations. We have demonstrated that perplexity features are the best overall on balance between accuracy, generalization, and training complexity, and proposed new algorithms for boundary detection.

Finally, our analysis has uncovered gaps in current approaches and discovered difficult aspects of the task, which we plan to address in future research.

7 Limitations

The task of detecting the exact boundary between human-written and machine-generated text is extremely hard in cross-domain and cross-model settings, both for short texts such as RoFT (due to lack of information) and longer texts (due to a large space of possibilities). Therefore, it is no wonder that none of suggested methods have achieved a really high quality in this setting, but the results in any case suggest a large room for improvement. Besides, we have to note that all methods we considered were based on Transformers with relatively small context window size (RoBERTa, GPT-2). This fact limits the transferability of the proposed approaches onto longer text samples. Another limitation is that methods relying on RoBERTa fine-tuning and on intrinsic dimension estimation are slower than methods relying on perplexity estimation.

References

Henry Adams, Manuchehr Aminian, Elin Farnell, Michael Kirby, Joshua Mirth, Rachel Neville, Chris Peterson, and Clayton Shonkwiler. 2020. A fractal dimension for measures via persistent homology. In *Topological Data Analysis: The Abel Symposium 2018*, pages 1–31. Springer.

Laurent Amsaleg, Oussama Chelly, Michael E Houle, Ken-Ichi Kawarabayashi, Miloš Radovanović, and Weeris Treeratanajaru. 2019. Intrinsic Dimensionality Estimation within Tight Localities. In *2019 SIAM International conference on Data Mining*, pages 181–189, Calgary (Alberta), Canada. Society for Industrial and Applied Mathematics.

Serguei Barannikov. 1994. The framed morse complex and its invariants. *Advances in Soviet Mathematics*, 21:93–116.

Tolga Birdal, Aaron Lou, Leonidas J Guibas, and Umut Simsekli. 2021. Intrinsic dimension, persistent homology and generalization in neural networks. *Advances in Neural Information Processing Systems*, 34:6776–6789.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Gunnar Carlsson. 2020. Persistent homology and applied homotopy theory. In *Handbook of Homotopy Theory*, pages 297–329. Chapman and Hall/CRC.

Elizabeth Clark, Tal August, Sofia Serrano, Nikita Haduong, Suchin Gururangan, and Noah A. Smith. 2021. All that's 'human' is not gold: Evaluating human evaluation of generated text. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7282–7296, Online. Association for Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

Joseph Cutler, Liam Dugan, Shreya Havaldar, and Adam Stein. 2021. Automatic detection of hybrid human-machine text boundaries.

Marco Cuturi. 2011. Fast global alignment kernels. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 929–936.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Liam Dugan, Daphne Ippolito, Arun Kirubarajan, and Chris Callison-Burch. 2020. RoFT: A tool for evaluating human detection of machine-generated text. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 189–196, Online. Association for Computational Linguistics.

Liam Dugan, Daphne Ippolito, Arun Kirubarajan, Sherry Shi, and Chris Callison-Burch. 2023. Real or fake text? investigating human ability to detect boundaries between human-written and machine-generated text. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'23/IAAI'23/EAAI'23. AAAI Press.

Naiereh Elyasi and Mehdi Hosseini Moghadam. 2019. An introduction to a new text classification and visualization for natural language processing using topological data analysis.

Jerome H. Friedman. 2000. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29:1189–1232.

Jerome H. Friedman. 2001. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232.

Jerome H. Friedman. 2002. Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38(4):367–378. Nonlinear Methods and Data Mining.

Karl Pearson F.R.S. 1901. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572.

Aarish Iyer and Soroush Vosoughi. 2020. Style change detection using bert. In *Conference and Labs of the Evaluation Forum*.

Xinyin Jiang, Haoliang Qi, Zhijie Zhang, and Mingjie Huang. 2022. Style change detection: Method based on pre-trained model and similarity recognition. In *Conference and Labs of the Evaluation Forum*.

Keenan Jones, Jason RC Nurse, and Shujun Li. 2022. Are you robert or roberta? deceiving online authorship attribution models using neural text generators. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 16, pages 429–440.

Nitish Shirish Keskar, Bryan McCann, Lav Varshney, Caiming Xiong, and Richard Socher. 2019. CTRL - A Conditional Transformer Language Model for Controllable Generation. *arXiv preprint arXiv:1909.05858*.

Laida Kushnareva, Daniil Cherniavskii, Vladislav Mikhailov, Ekaterina Artemova, Serguei Barannikov, Alexander Bernstein, Irina Piontkovskaya, Dmitri Piontkovski, and Evgeny Burnaev. 2021. Artificial text detection via examining the topology of attention maps. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 635–649, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A lite BERT for self-supervised learning of language representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Qidi Lao, Li Ma, Wenyin Yang, Zexian Yang, Dong Yuan, Zhenlin Tan, and Langzhang Liang. 2022. Style change detection based on bert and conv1d. In *Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum, Bologna, Italy, September 5th - to - 8th, 2022*, volume 3180 of *CEUR Workshop Proceedings*, pages 2554–2559. CEUR-WS.org.

Tzu-Mi Lin, Chao-Yi Chen, Yu-Wen Tzeng, and Lung-Hao Lee. 2022. Ensemble pre-trained transformer models for writing style change detection. In *Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum, Bologna, Italy, September 5th - to - 8th, 2022*, volume 3180 of *CEUR Workshop Proceedings*, pages 2565–2573. CEUR-WS.org.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. Cite arxiv:1907.11692.

German Magai. 2023. Deep neural networks architectures from the perspective of manifold learning. *arXiv preprint arXiv:2306.03406*.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. 2023. Detectgpt: Zero-shot machine-generated text detection using probability curvature.

Hariharan Narayanan and Sanjoy K. Mitter. 2010. Sample complexity of testing the manifold hypothesis. In *NIPS*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Benjamin Schweinhart. 2020. Fractal dimension and the persistent homology of random geometric complexes. *Advances in Mathematics*, 372:107291.

Primoz Skraba, Gugan Thoppe, and D Yogeshwaran. 2017. Randomly weighted $d-$ complexes: Minimal spanning acycles and persistence diagrams. *arXiv preprint arXiv:1701.00239*.

Irene Solaiman, Miles Brundage, Jack Clark, Amanda Askell, Ariel Herbert-Voss, Jeff Wu, Alec Radford, Gretchen Krueger, Jong Wook Kim, Sarah Kreps, et al. 2019. Release strategies and the social impacts of language models. *arXiv preprint arXiv:1908.09203*.

J Michael Steele. 1988. Growth rates of euclidean minimal spanning trees with power weighted edges. *The Annals of Probability*, 16(4):1767–1787.

M. Eduard Tudoreanu. Exploring the use of topological data analysis to automatically detect data quality faults. *Frontiers in Big Data*, 5.

Eduard Tulchinskii, Kristian Kuznetsov, Laida Kushnareva, Daniil Cherniavskii, Serguei Barannikov, Irina Piontkovskaya, Sergey Nikolenko, and Evgeny Burnaev. 2023a. Intrinsic dimension estimation for robust detection of ai-generated texts.

Eduard Tulchinskii, Kristian Kuznetsov, Laida Kushnareva, Daniil Cherniavskii, Serguei Barannikov, Irina Piontkovskaya, Sergey Nikolenko, and Evgeny Burnaev. 2023b. Topological Data Analysis for Speech Processing. In *Proc. INTERSPEECH 2023*, pages 311–315.

Adaku Uchendu, Thai Le, and Dongwon Lee. 2023a. Attribution and obfuscation of neural text authorship: A data mining perspective. *ACM SIGKDD Explorations Newsletter*, 25(1):1–18.

Adaku Uchendu, Thai Le, and Dongwon Lee. 2023b. Toproberta: Topology-aware authorship attribution of deepfake texts.

Saranya Venkatraman, Adaku Uchendu, and Dongwon Lee. 2023. Gpt-who: An information density-based machine-generated text detector. *arXiv preprint arXiv:2310.06202*.

Xianjun Yang, Liangming Pan, Xuandong Zhao, Haifeng Chen, Linda Petzold, William Yang Wang, and Wei Cheng. 2023. A survey on detection of llms-generated content. *arXiv preprint arXiv:2310.15654*.

Eva Zangerle, Maximilian Mayerl, Martin Potthast, and Benno Stein. 2021. Overview of the style change detection task at pan 2021. In *Conference and Labs of the Evaluation Forum*.

Zijie Zeng, Lele Sha, Yuheng Li, Kaixun Yang, Dragan Gašević, and Guanliang Chen. 2023. Towards automatic boundary detection for human-ai hybrid essay in education. *arXiv preprint arXiv:2307.12267*.

A Intrinsic Dimension Estimation Methods

According to the manifold hypothesis (Narayanan and Mitter, 2010), the data X lies on a low-dimensional submanifold: $X \subseteq M^n \subseteq R^d$, where d is the extrinsic dimension and n is the intrinsic dimension (ID). The geometric and topological properties of the manifold M are of particular interest. There are various methods for estimating the ID that can be divided into global and local methods.

For the tight local intrinsic dimension estimator (TLE) proposed by Amsaleg et al. (2019), we use the neighborhood center point x, a set of neighborhood samples V and a specially defined distance between points in a sufficiently small neighborhood of x:

$$d_{x,r}(q,v) = \frac{r(v-q)\cdot(v-q)}{2(x-q)\cdot(v-q)}, \qquad (1)$$

where r is the radius of the neighbourhood. For every three points x, v, w we can compute

$$M(x,v,w) = \ln \frac{d_{x,r}(v,w)}{r} + \ln \frac{d_{x,r}(2x-v,w)}{r}.$$

If $V_* = V \cup \{x\}$, then the intrinsic dimension can be found by averaging the estimates for all points x, as defined by the following formula:

$$\hat{m}_r(x) = -\left(\frac{1}{|V_*|^2}\sum_{v,w\in V_*,\ v\neq w} M(x,v,w)\right)^{-1} \qquad (2)$$

Applied algebraic topology provides effective tools for analyzing the topological structure of data. The theoretical foundations of topological data analysis (TDA) have been described in detail by, e.g., Barannikov (1994) and Carlsson (2020). TDA allows us to consider a dataset $X \subseteq R^d$ from the topological point of view. In order to move from point clouds X to topological spaces, it is necessary to approximate the data by a simplicial complex R. In our research, we use the Vietoris-Rips complex $R(X;t)$. The method of constructing the complex R is as follows: simplexes are formed by subsets of points from X whose pairwise distances do not exceed t (a scaling parameter). An increasing sequence of simplicial complexes is called a filtration: $\{R_t\}_{(t\geq 0)} = R_{t_0} \subseteq R_{t_1} \subseteq ... \subseteq R_s$.

Homology groups $H_i(R)$ are a topological invariant that expresses the properties of a topological space R. We use $\beta_i(R) = \dim H_i(R)$, which is known as the ith *Betti number*, a topological feature equal to the dimension of the homology group; for $i = 0, 1, 2$ the Betti number corresponds to the number of connectivity components, cycles, and cavities respectively.

Topological features appear and disappear at different values of t, which leads to the next core concept in TDA: the *barcode*. It summarises the dynamics of topological features in the filtration

process. A *bar* is the lifetime of the nth homology feature $I_n = t_n^{\text{birth}} - t_n^{\text{death}}$. A long bar in the barcode means that the data contains a fairly persistent and informative topological feature.

Schweinhart (2020) introduced the persistent homological fractal dimension (PHD) that generalized Steele (1988) for higher dimensions of the homology group and used the topological properties of the point cloud. The PHD has already been proven to be useful in the study of the properties of deep learning models (Birdal et al., 2021; Magai, 2023).

Let us denote the power-weighted sum of N bars for the ith degree of homology as follows:

$$E_\alpha^i(X) = \sum_{i=1}^{N} I_i^\alpha. \quad (3)$$

It is interesting to note that $E_1^0(X_n)$ is equal to the length of the Euclidean minimum spanning tree (MST) of $X_n \subseteq R^d$ (Skraba et al., 2017).

Then the persistent homological fractal dimension (PHD) can be defined as follows:

$$\text{PHD}^i = \frac{\alpha}{1-\beta}, \quad (4)$$

where

$$\beta = \lim_{n\to\infty} \sup \frac{\log(\mathbb{E}(E_\alpha^i(x_1, ..., x_n)))}{\log(n)}, \quad (5)$$

and $x_1, \ldots, x_n$ are sampled independently from X. That is, $\text{PHD}^i(X) = d$ if $E_\alpha^i(x_1, ..., x_n)$ scales as $n^{\frac{d-\alpha}{d}}$ and $\alpha \geq 0$ (we take $i = 0$, $\alpha = 1$). Persistent homological fractal dimension can be estimated by analyzing the asymptotical behavior at $n \to \infty$ of $E_\alpha^i(x_1...x_n)$ for every i. In other words, to calculate PHD we must find a power law that shows how $E_\alpha^i(x_1...x_n)$ scales as n increases. See Adams et al. (2020) and Schweinhart (2020) for more details.

B Cross-domain transfer

Table 3 supplements Table 2 from the original text, reporting the results of cross-domain transfer for our methods on the *RoFT-chatgpt* dataset between four text topics presented in the data. We report in-domain and out-of-domain accuracy: the IN column shows results from domains seen during training, while the OUT column reflects the model's ability to detect artificial texts in the unseen domain corresponding to this column. For each model, training was done on three out of the four domains, and the resulting model was tested on the fourth, unseen domain; we used 60% of these subsets, mixed together, as the training set, 20% as the validation set, and 20% as the test set for in-domain evaluation.

C Cross-model transfer

Tables 4 and 5 show our experimental results on cross-model transfer for all considered text generation models. The artificial text boundary detection models were trained on the parts of the dataset generated by all language models except one, which is held out for cross-model testing, and tested in-domain (ID) on the same parts and out-of-domain (OOD) on the remaining part generated by the held-out model.

D Additional Dataset Analysis

In this section we provide additional statistics and visualizations for the distributions of various features in the data. In particular, we note that on most diagrams, real texts have smaller PHD than fake texts, which is a very different result from the statistics presented by Tulchinskii et al. (2023a), who noted that the PHD of real texts is larger than that of fake texts. We hypothesize that it can be due either to very short lengths of texts in our work compared to the texts considered by Tulchinskii et al. (2023a) or due to differences in the sampling strategy used by Dugan et al. (2020) and Tulchinskii et al. (2023a) when generating texts. Another observation is that the TLE dimension is very different for all generator models in the original RoFT dataset. This may be the reason for the bad generalization performance of intrinsic dimension-based algorithms across domains. For *RoFT-chatgpt* PHD and TLE, real and fake texts are close to each other.

We show dataset statistics in the following figures:

- Figures 2, 6, and 7 show the lengths of texts in tokens produced by the standard RoBERTa tokenizer (the figures have a cutoff of 100 for readability but the datasets do contain a few longer sentences);
- Figure 8 shows the distribution of pretrained (but not fine-tuned) RoBERTa [CLS] embeddings for real and fake parts of text samples from the original RoFT and *RoFT-chatgpt* datasets;

Table 3: Mean squared errors from leave-one-out cross-domain evaluation on *RoFT-chatgpt*. △ and ▽ indicate the relative change from the detection model's *in-domain* score to the human score, while ▲ and ▼ represent the relative change from the *out-of-domain* score to the *in-domain* score. *Green* highlights improvements, *red* indicates deteriorations.

Predictor	**Model**	**Context**	**Pres. Speeches** IN ↓	**Pres. Speeches** OUT ↓	**Recipes** IN ↓	**Recipes** OUT ↓	**New York Times** IN ↓	**New York Times** OUT ↓	**Short Stories** IN ↓	**Short Stories** OUT ↓	**Avg** Δ
Text	RoBERTa SEP	global	02.6▽81%	10.6▲308%	02.6▽81%	18.3▲604%	03.4▽76%	07.9▲132%	02.3▽83%	09.0▲291%	334%
Text	RoBERTa	global	02.3▽84%	07.5▲227%	02.8▽80%	13.5▲389%	02.9▽79%	06.2▲115%	02.6▽81%	05.5▲110%	211%
Perpl.	GB	sentence	07.3▽47%	08.9▲22%	07.0▽49%	14.6▲108%	07.2▽48%	09.4▲31%	08.5▽39%	08.5▼00%	40%
Perpl.	LogReg	sentence	08.2▽41%	11.5▲41%	06.0▽57%	16.8▲180%	08.7▽37%	11.8▲36%	09.6▽31%	09.3▼03%	64%
Perpl.	Regr. (GB)	sentence	04.7▽66%	06.6▲41%	04.8▽65%	09.7▲102%	04.9▽65%	05.7▲17%	05.2▽63%	05.0▼04%	39%
PHD	TS multilabel	100 tokens	12.3▽11%	14.6▲19%	10.7▽23%	11.0▲03%	14.1△01%	16.8▲19%	11.6▽17%	11.6▼00%	10%
TLE	TS Binary	20 tokens	12.3▽12%	15.6▲27%	18.0△29%	15.4▼14%	12.0▽13%	17.6▲47%	17.4△25%	23.9▲38%	24%
Length	GB	sentence	12.8▽08%	15.9▲24%	14.2△02%	17.9▲26%	13.7▽01%	18.4▲34%	12.5▽10%	15.3▲22%	26%
Length	LogReg	sentence	20.1△45%	20.5▲02%	16.7△20%	24.7▲48%	17.2△24%	22.1▲28%	18.5△33%	22.9▲24%	26%
Majority		—	27.5△98%		27.4△97%		27.9△101%		28.0△102%		—
Approximated human**		global	13.88		13.88		13.88		13.88		

Table 4: Original RoFT, cross-model transfer, part 1. The models were trained on all parts of the dataset except one and tested in-domain (ID) on the same parts and out-of-domain (OOD) on the remaining part.

Model	**Metric**	**GPT2-XL** ID	**GPT2-XL** OOD	**GPT2** ID	**GPT2** OOD	**davinci** ID	**davinci** OOD
RoBERTa + SEP	Acc, %	46.38	40.94	45.95	08.78	63.25	19.73
RoBERTa + SEP	SoftAcc1, %	76.85	76.83	76.71	31.22	85.52	47.27
RoBERTa + SEP	MSE	03.90	02.92	04.17	06.77	03.04	07.59
RoBERTa	Acc, %	46.68	32.56	40.30	06.94	57.20	14.48
RoBERTa	SoftAcc1, %	77.30	72.07	75.52	25.31	84.61	40.49
RoBERTa	MSE	03.73	03.10	03.70	07.18	02.94	08.56
Perplexity + GB	Acc, %	23.00	23.43	28.12	04.08	23.75	19.78
Perplexity + GB	SoftAcc1, %	40.35	47.90	47.96	30.61	46.03	42.46
Perplexity + GB	MSE	15.39	10.51	12.19	15.99	11.85	15.38
Perplexity + LogRegr	Acc, %	21.27	08.86	23.67	03.47	21.43	08.31
Perplexity + LogRegr	SoftAcc1, %	33.33	22.14	39.80	27.45	35.35	25.19
Perplexity + LogRegr	MSE	21.52	24.48	16.99	18.98	19.71	22.06
Perplexity + Regr	Acc, %	11.68	15.78	14.56	14.18	13.91	15.30
Perplexity + Regr	SoftAcc1, %	34.84	49.37	39.36	47.86	46.10	44.43
Perplexity + Regr	MSE	07.67	04.62	06.80	06.40	06.73	06.64
PHD + TS ML	Acc, %	31.84	04.02	25.64	04.49	31.38	02.05
PHD + TS ML	SoftAcc1, %	56.08	14.14	44.28	35.85	53.31	13.82
PHD + TS ML	MSE	11.13	28.18	16.37	10.74	11.82	27.03
TLE + TS Binary	Acc, %	14.17	03.15	12.36	07.76	14.48	00.98
TLE + TS Binary	SoftAcc1, %	31.14	13.10	29.19	32.14	34.01	11.58
TLE + TS Binary	MSE	21.58	28.20	21.36	16.35	18.12	30.45
Length + GB	Acc, %	21.70	04.75	18.30	01.22	19.18	06.33
Length + GB	SoftAcc1, %	36.0	09.48	33.20	22.02	36.09	22.07
Length + GB	MSE	23.56	33.32	27.00	19.27	17.35	20.00
Human Baseline	Acc, %	22.48	17.23	22.59	22.53	24.74	14.06
Human Baseline	SoftAcc1, %	41.91	37.03	39.73	48.01	42.44	33.47
Human Baseline	MSE	13.49	14.69	14.29	09.86	14.03	12.91

Table 5: Original RoFT, cross-model transfer, part 2. The models were trained on all parts of the dataset except one and tested in-domain (ID) on the same parts and out-of-domain (OOD) on the remaining part.

Model	**Metric**	**ctrl-Politics**		**ctrl-nocode**		**tuned**		**baseline**	
		ID	OOD	ID	OOD	ID	OOD	ID	OOD
RoBERTa + SEP	Acc, %	49.35	59.12	50.20	60.61	49.55	23.85	51.35	06.35
RoBERTa + SEP	SoftAcc1, %	78.49	89.31	80.10	84.85	81.55	56.96	79.84	15.87
RoBERTa + SEP	MSE	02.86	01.24	02.93	01.87	02.33	06.33	02.51	39.32
RoBERTa	Acc, %	47.46	44.03	46.07	54.55	45.91	20.98	47.29	04.76
RoBERTa	SoftAcc1, %	78.43	85.53	78.01	86.87	80.13	52.88	78.49	15.87
RoBERTa	MSE	02.80	01.21	02.79	01.07	02.37	06.49	02.69	36.00
Perpl. + GB	Acc, %	25.27	10.69	25.32	07.07	30.88	10.15	24.44	06.35
Perpl. + GB	SoftAcc1, %	48.89	27.04	47.71	24.24	53.87	29.20	47.64	14.29
Perpl. + GB	MSE	11.81	20.96	12.27	22.70	11.67	17.86	12.01	39.62
Perpl. + LogRegr	Acc, %	24.08	07.55	21.88	07.07	28.50	08.04	24.77	03.17
Perpl. + LogRegr	SoftAcc1, %	42.23	23.90	38.84	19.19	42.78	22.28	42.45	15.87
Perpl. + LogRegr	MSE	15.70	23.27	16.69	25.38	17.46	24.20	15.72	40.86
Perpl. + Regr	Acc, %	14.80	13.21	13.96	14.14	15.70	11.90	13.62	03.17
Perpl. + Regr	SoftAcc1, %	42.06	40.25	41.89	33.33	43.08	36.03	39.25	14.29
Perpl. + Regr	MSE	06.40	07.78	06.55	08.31	06.81	07.10	06.84	22.86
PHD + TS ML	Acc, %	25.70	08.18	25.18	11.11	21.44	12.58	23.13	03.70
PHD + TS ML	SoftAcc1, %	47.33	32.70	47.30	39.39	35.31	26.37	45.64	05.56
PHD + TS ML	MSE	14.09	11.23	13.62	09.41	18.28	18.43	13.42	52.39
TLE + TS binary	Acc, %	10.98	05.03	11.53	04.04	14.51	06.56	12.39	03.18
TLE + TS binary	SoftAcc1, %	29.08	18.87	28.21	22.22	30.21	17.26	28.21	15.88
TLE + TS binary	MSE	20.36	23.28	20.84	21.82	21.82	26.64	20.59	25.55
Length + GB	Acc, %	17.32	03.77	18.05	0.0	21.64	0.04	15.24	26.98
Length + GB	SoftAcc1, %	33.96	18.86	35.71	15.15	35.80	10.87	32.56	34.92
Length + GB	MSE	22.48	23.87	24.16	26.53	23.59	32.78	16.25	16.0
Human Baseline	Acc, %	22.60	21.6	22.57	23.94	21.46	25.90	22.41	46.15
Human Baseline	SoftAcc1, %	40.61	41.6	40.59	45.07	39.17	44.92	40.46	63.46
Human Baseline	MSE	13.87	10.57	13.87	07.70	13.92	13.48	13.82	11.51

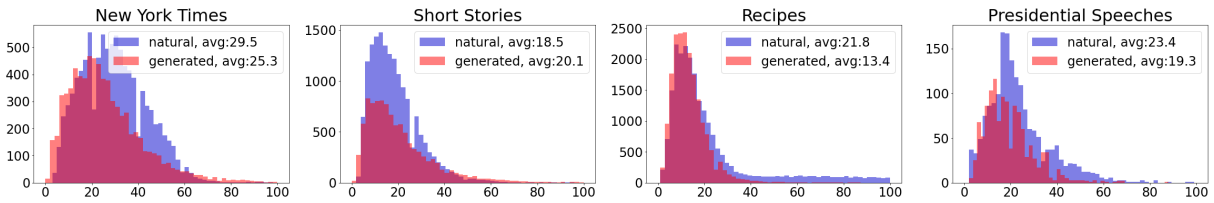


Figure 6: Sentence length distributions in RoBERTA tokens, original RoFT, by topic

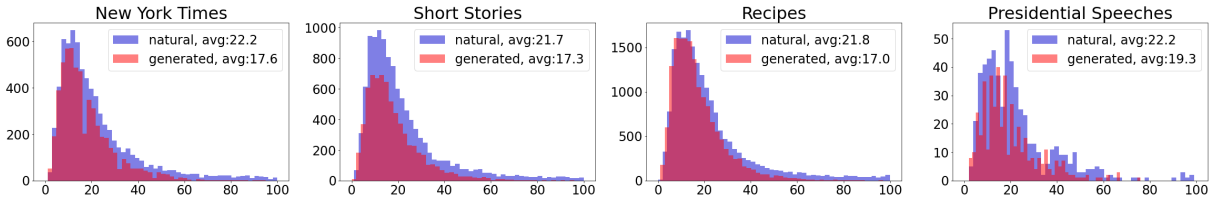


Figure 7: Sentence length distributions in RoBERTA tokens, RoFT-chatgpt, by topic

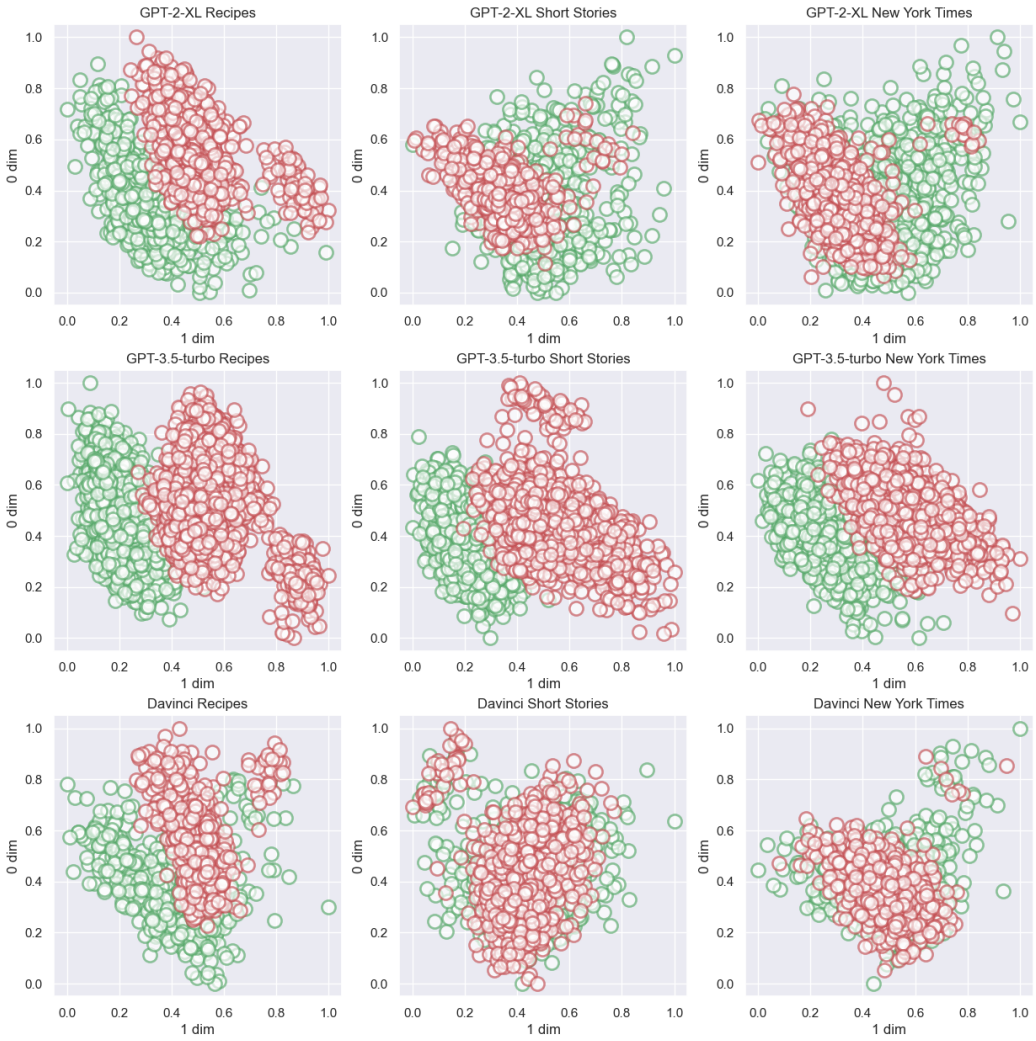


Figure 8: Distribution of pretrained (but not fine-tuned) RoBERTa [CLS] embeddings of real and fake parts of text samples from the original RoFT and *RoFT-chatgpt* datasets. The dimension is reduced to 2D via principal component analysis (F.R.S., 1901).

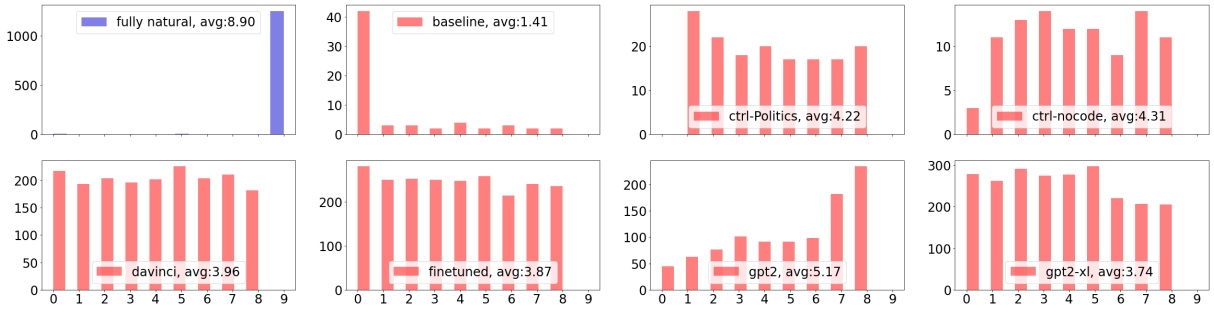


Figure 9: Label distributions for the original RoFT dataset by model

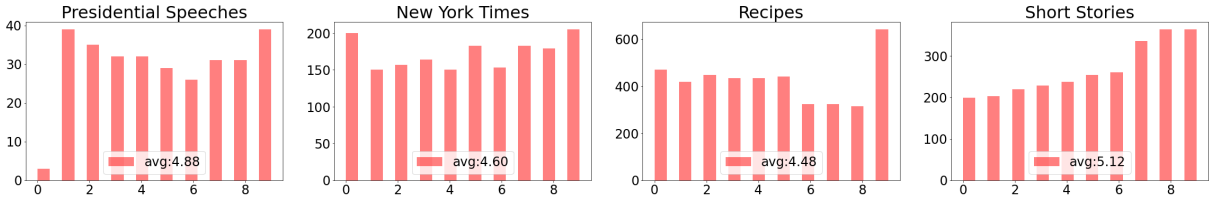


Figure 10: Label distributions for the original RoFT dataset by topic

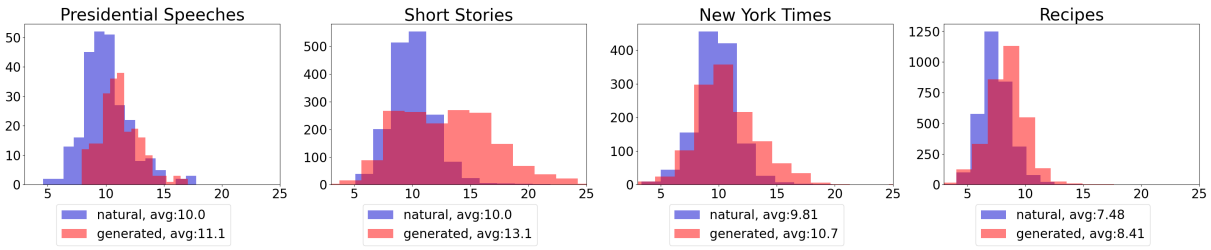


Figure 11: PHD distributions for the real and fake parts of the RoFT dataset by topics

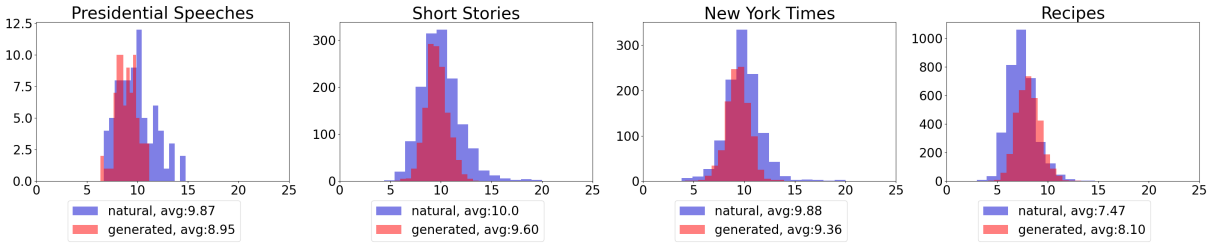


Figure 12: PHD distributions for the real and fake parts of the *RoFT-chatgpt* dataset text by topics

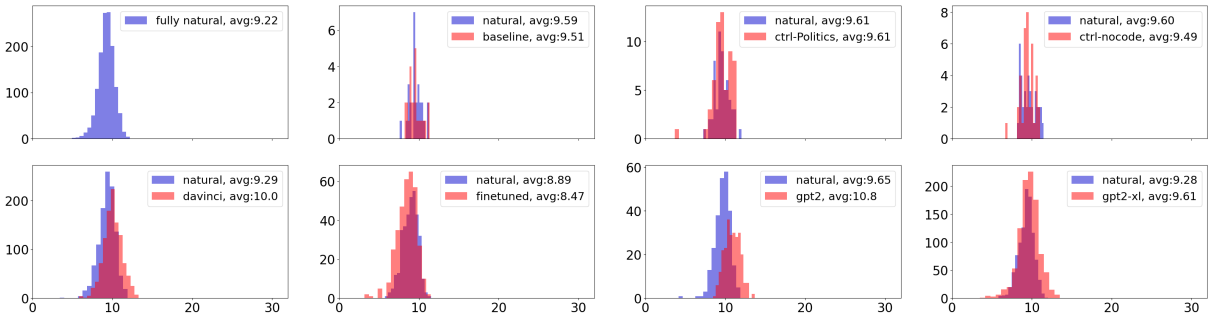


Figure 13: TLE dimension distributions for the sentences in the RoFT dataset by generator models

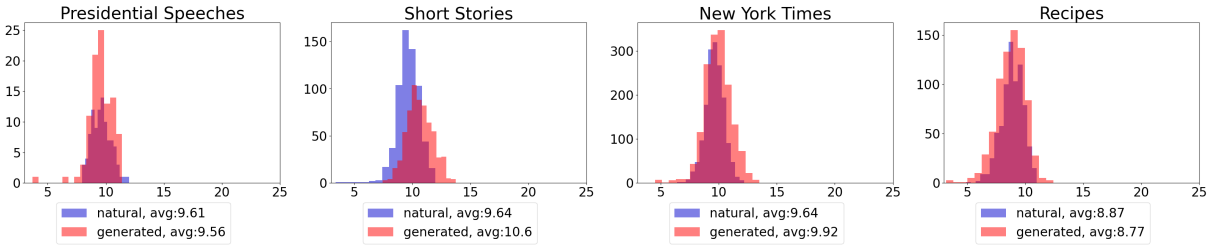


Figure 14: TLE dimension distributions for the sentences in the RoFT dataset by topics

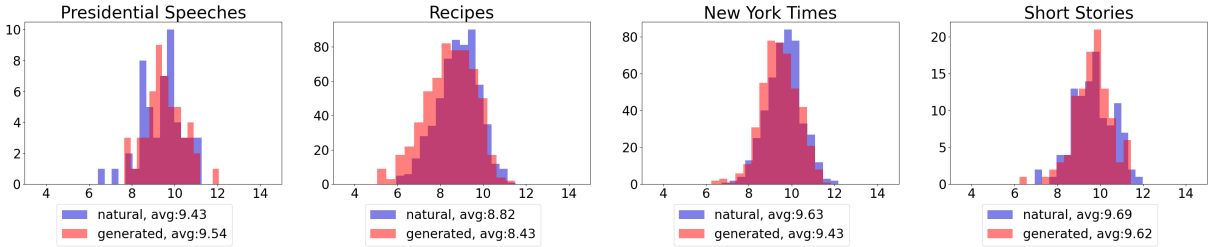


Figure 15: TLE dimension distributions for the sentences in the *RoFT-chatgpt* dataset by topics

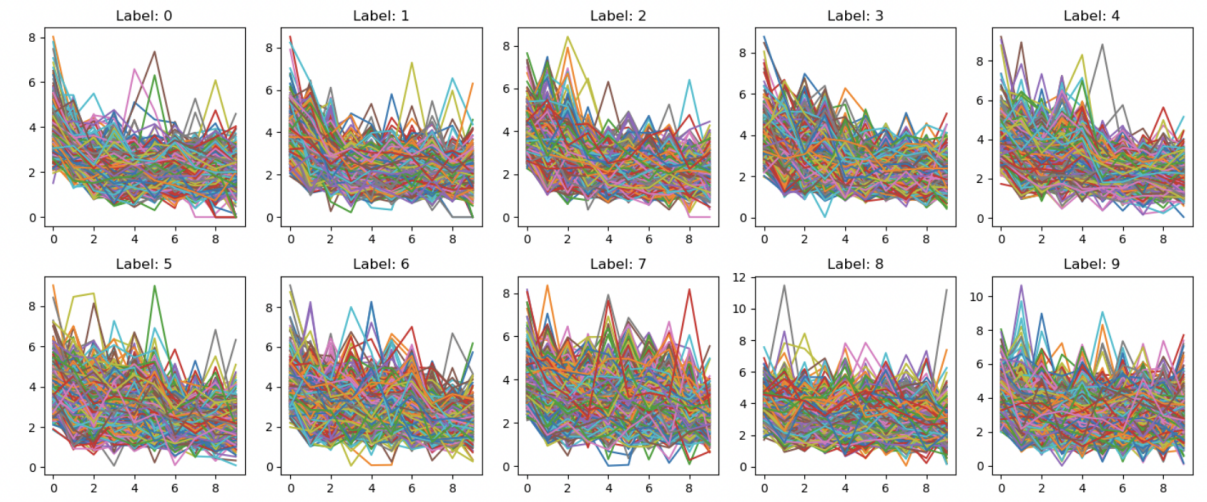
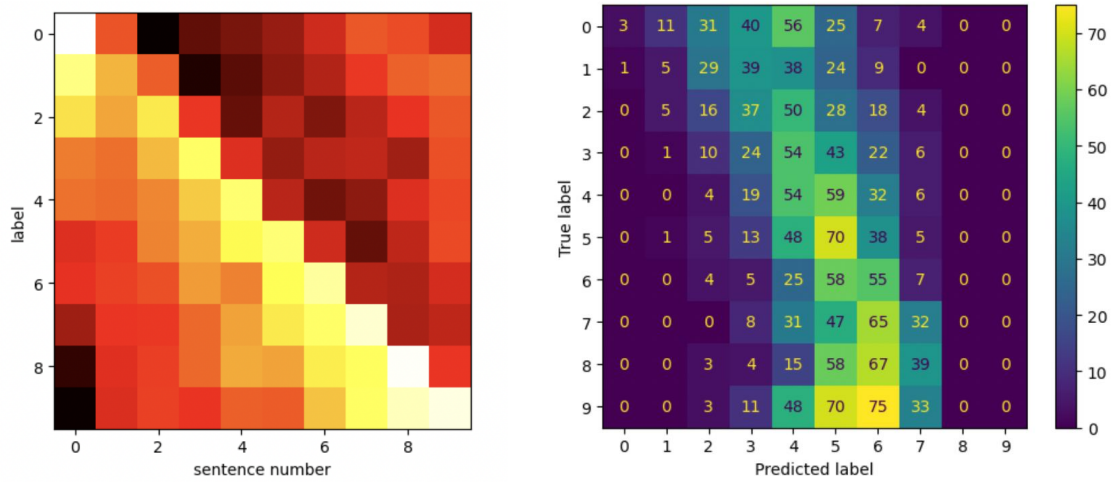


Figure 16: Sentence perplexities in the *RoFT-chatgpt* dataset by label. X axis: sentence index in the text, Y axis: sentence perplexity.



(a) (b)

Figure 17: Analysis of the logistic regression trained on sentence perplexities in the *RoFT-chatgpt* dataset (*Perplexity + LogRegr* in the tables): (a) heatmap of the coefficients; (b) confusion matrix for test set predictions.

- Figure 9 shows the distribution of labels in the original RoFT dataset by generator;
- Figure 10 shows the distribution of labels in the original RoFT dataset by topic; this distribution is identical to the corresponding distribution for the *RoFT-chatgpt* dataset;
- Figure 5 shows the distribution of PH dimensions of real and fake parts of the text by generator;
- Figures 11 and 12 show the distributions of PH dimensions by topic for the original RoFT and *RoFT-chatgpt* respectively;
- Figure 13 shows the distribution of TLE dimensions of different sentences by generator;
- Figures 14 and 15 show the the distributions of TLE dimensions by topic for original RoFT and *RoFT-chatgpt* respectively.

E Detailed experimental results

In this section, we provide additional statistics and visualizations regarding our experimental results. Figure 16 visualizes the changes in perplexities for sentences from the texts in *RoFT-chatgpt* by their labels. We make the following observations.

First, perplexities of the first couple of sentences across all texts are quite high, and the average perplexity of sentences decreases by the end of the text. This is probably due to the fact that for the words of the first sentences the length of the text prefix is not enough for a stable calculation of perplexity.

One solution to mitigate this effect and hence make perplexity-based classifiers more stable might be to generate new prefixes for the text using some generative model (e.g. *gpt-3.5*) and calculate perplexities of original text words using this generated prefix. We leave this idea for further research.

Second, there is no evident spike of perplexity at the start of the fake text. This is an additional indication for the fact that artificial text boundary detection may be a far harder problem than artificial text detection by classifying full texts into real and fake.

Figure 17a visualizes the coefficients of a logistic regression model trained on sentence perplexities from the *RoFT-chatgpt* dataset (*Perplexity + LogRegr* rows in the tables). We can see a distinct pattern in this figure. For the label k, which means that the first fake sentence in the text is the (k+*1*)st, the highest value of the coefficient is the kth one, and the lowest one is often the (k+*2*)nd. This could mean that the model is "searching" for a sudden drop of perplexity at a point where the fake part is starting. This fits together well with the idea that GPT-2 sees text generated by a similar model (GPT-3.5-turbo) as a more "natural" one than real human-produced text. Therefore, perplexity often drops at the point where fake text begins, and logistic regression can pick up this effect and use it as a decision rule.

Finally, Figure 17b visualizes the confusion matrix on the test set of a gradient boosting regressor trained on the original RoFT dataset. We can see that its predictions are highly concentrated around the center labels (3-6), although MSE scores of the gradient boosting regressor on both in-domain and out-of-domain sets are in top-2 among other approaches for almost any test set (Tables 4 and 5). This suggests that further research on the errors of different models on different data subsets is needed.

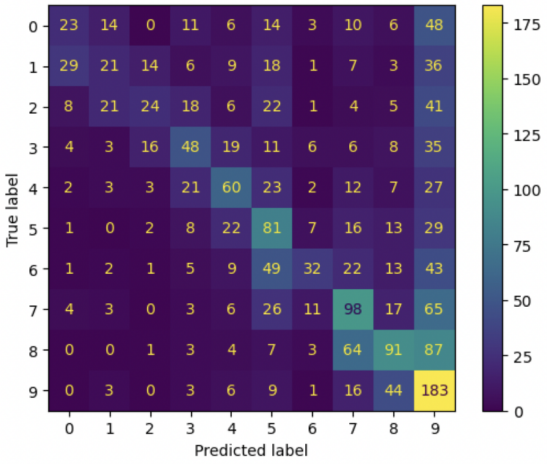


Figure 18: Confusion matrix for the predictions of logistic regression trained on sentence perplexities in the *RoFT-chatgpt* dataset (*Perplexity + LogRegr* in the tables) without the *Short Stories* topic, tested on the *Short Stories* subset.