Reverse Turing Test: Artificially Generated Text Detector

Keith Rebello

03/07/2022

Abstract

With the development of advanced artificial text generation techniques such as GPT-3, It has become increasingly easier for artificial agents to develop texts that are seemingly indistuingishable from humans. Several efforts have been underway over the last 5 years to look at the development of counteractive tools that differentiate between artificial generated texts and human text. This study aims to look at the ability of classifiers such as logistic regression, decision trees, random forest, RNNs and transformers to differentiate between human generated and artficially generated text.

1 Introduction

Modern natural language generation techniques have surpassed the expectations of the common user. Techniques like GPT-3 have demonstrated the capacity of text agents to develop swathes of text-based content that has not existed before. Such artificially generated texts can be concerning to indviduals as in the wrong hands it can be misused to infiltrate social media platforms, generate large swarms of misinformation, and create harmful texts. Therefore a set of forensic techniques is required to be able differentiate between artificially generated text and human generated text. It is especially important to be able to use such tools to detect shorter tweet-style content which can prevent the spread of bots and chatbots over media platforms. This study aims to develop and compare different classification techniques such as logistic regression, random forest, RNNs and transformers to differentiate between human generated and artficially generated short-length text-content. In this study a comparitive analysis will be done by looking at the capabilities of the aforementioned classifiers at detecting text created by different kinds of text generation techniques such as GPT-2, generic RNNs, LSTMs, and Markov Models.

The primary application of such a classifier would be in detecting generated text on social media platforms such as Twitter or Instagram. Another application would be to detect chatbots. As an alternative application, possibly a future scope, such a detector would also be capable of determining the capability of a conversational agent to carry out a human conversation with a human, without the dependancy on human evaluation or other metrics such as BLEU scores.

2 Proposed Project

The project will use classification to solve the given problem. As this is a relatively new problem, (most research developments take place only in 2019) Finding a dataset that perfectly encapsulates this problem statement has been the first challenge. Most datasets available are limited by the scope with most only looking at text from a single text-generation technique or chatbot (WOCHAT, RADNY). Others that incorporate multiple bots have issues in structure for this kind of problem statement. The ConvAI2 dataset which includes data from 5 different bots and their conversations with 789 humans would have been ideal for this problem statement, however the rigid focus of the data collection which insists on human speakers being the first to intiate and end the conversation would afford a propensity for a strong bias in the classifier (a bot could understand this structure and simply always pick the second speaker as the bot, achieving 100% accuracy.) The dataset that will be used is the TweepFake dataset from kaggle [1] which was developed as part of a research study by Fagni et al [2]. The dataset consists of 25,838 samples of tweets from Bots and Humans. It is a balanced dataset with 50% of the tweets coming from human accounts and 50% coming from Bot accounts. There are 9 different kinds of techniques of text-generation (GPT-2, RNN, Markov Models, LSTM, Markov Chains,CharRNN, OpenAI, RNN+Markov) employed by this dataset with two unknown text generation methods also included in the dataset. This problem will simply look to classify a given text as bot (1) or human (0) and hence the 50-50 split is the most important facet of this dataset.

Pre-processing will be one of the challenges with this task. The dataset includes only the raw IDs of the tweets and their users IDs, so retrieval of the text will be the first challenge to overcome. Following the retrieval, cleaning the text by using pre-processing techniques such as stop words, hashtag and emoji processing, lemmatization and stemming, punctuation, will be the next challenge to overcome. Finally embedding the text in an appropriate embedding (this can affect the accuracy of the model greatly) and comparing them accross the different kinds of classifiers will be the last major challenge for this problem statement.

3 Theory

3.1 Random Forest Classifier

3.2 Support Vector Machine

3.3 Logistic Regression

3.4 Long-Short Term Memory

4 Methodology

5 Experimental Results

`https://github.com/keithRebello/Artificially-Generated-Text-Detector`

6 Conclusions

7 Contributions

References

[1] TweepFake - Twitter deep Fake text Dataset.

[2] Tiziano Fagni, Fabrizio Falchi, Margherita Gambini, Antonio Martella, and Maurizio Tesconi. TweepFake: about Detecting Deepfake Tweets. *PLOS ONE*, 16(5):e0251415, May 2021. arXiv: 2008.00036.