Stenography GAN: Cracking Stenography with Cycle Generative Adversarial Networks

1st Nibraas Khan
Department of Computer Science
Middle Tennessee State University
Murfreesboro, TN
nak2z@mtmail.mtsu.edu

2nd Ruj Haan
Department of Computer Science
Middle Tennessee State University
Murfreesboro, TN
gm3g@mtmail.mtsu.edu

3rd George Boktor
Department of Computer Science
Middle Tennessee State University
Murfreesboro, TN
gsb3c@mtmail.mtsu.edu

4th Michael McComas
Department of Computer Science
Middle Tennessee State University
Murfreesboro, TN
mrm8m@mtmail.mtsu.edu

5th Ramin Daneshi
Department of Computer Science
Middle Tennessee State University
Murfreesboro, TN
rd3s@mtmail.mtsu.edu

***Abstract*—This document is a model and instructions for LaTeX. This and the IEEEtran.cls file define the components of your paper [title, text, heads, etc.]. *CRITICAL: Do Not Use Symbols, Special Characters, Footnotes, or Math in Paper Title or Abstract.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

Talk about the importance of cyrpto.

The sten specifically.

Sten used in real life.

There have crack to crack cyrpto and steno.

We are also taking on the same task. Our approach is cycle gans.

Before that, we need to about gans in general.

History of cycle gans

Pix2Pix and CNN

Why will this work/or why is this worth exploring

In order to understand its need to compare it to other models, and in this case autoencoders

For maximum accruacy we also introduced. Bayesian optimization which is ...

Start typing here [1].

II. Background

The details of the sten algo. Include some images. Give an example.

We used cycles gans to crack this algo. That etails:

Regular GANs. Include images and the math

Cycle GANs. Explain it a little bit

Our implemention of cycle gans ultalizes pix2pix

Explain what pix2pix is. Include images and rough outlay of the math

Pix2pix emplyed CNN. What they are and the math behind it.

CipherGANs. Explain it a little bit

Autoencoders. Explain it a little bit

Baysian optimzation. Explain it a little but

III. Methods

There are several components to Cycle Generative Adversarial Networks and our implementation of it. Each component works together for the success of our model. There is a certain degree of flexibility with our components, but we will present our implementation of it. We will present the implementation of our CycleGAN and autoencoder and our testing protocols.

A. Cycle Generative Adversarial Network Model Description

Math for the cycle gans

1) Pix2Pix and Convolutional Neural Networks: What is pix2pix and what are CNN

B. Autoencoders Model Description

Math behind the autoencoders

C. Testing Protocols

1) Bayesian Optimization: Math bethind the bayes opt

2) Cycle Generative Adversarial Network Training and Testing: Training CycleGAN: The training and testing data How we actually flow through the network Bayes opt

3) Autoencoder Training and Testing: Training Autoencoders: The training and testing data How we actually flow through the network

D. Various Training Techniques

1) Bit Size Variation: Different training techniques: Bit size

IV. Results

Here are the result for cycle gan: Grab images from the cycle gan algo

Here are the results for autoencoder: Grab images from the autoencoder section

Here are the results for messing with different bit size: Images from bit = 7, 6, 5, 4, 3

Here are the results for cycle gan with bayes opt:

V. Discussion

Summary of the introduction

If our model was successful. Why was it not successful.

Extra things that we can work on.

Our model presentes many fruitful avenues of research

References

[1] L. D. Trang and Z. Mebkhout, "Variétés caractéristiques et variétés polaires," *C. R. Acad. Sc. Paris*, vol. 296, pp. 129–132, 1983.