emoji generation based on textual input using generative adversarial networks

DARSHIL AJAY PAREKH

Final Thesis Report

JUNE 2024

DEDICATION

Firstly, I would like to dedicate the master’s thesis project to my parents who have inspired, supported and always encouraged me during the journey. TODO

ACKNOWLEDGEMENTS

I want to thank and express my gratitude to all those who have contributed to the development of this master’s thesis. TODO

ABSTRACT

With the increase in popularity and its wide variety of uses, emojis have become so necessary and important that sending or responding to a message and commenting or reacting to a post on social media without emoji seems to be impossible. Emoji represents emotions, culture, age, gender, actions, gestures, expressions, sports, foods, places and objects.

Despite the importance the number of emojis are countable and finite in quantity. In this research we will be creating a generative model to generate new unique emojis using short text for more complex and distinction emojis

We will be using the state-of-the-art generative models which consist of mainly two-part generator and discriminator. The generator and discriminator will work one after the other. First the generator will be trained to generate new emojis and after couple of rounds the discriminator will be trained to detect is the emoji is fake or real. This is followed by generator and discriminator rounds. This process continues until the discriminator is not able to detect is the emoji is fake.

This will help the community to generate wide variety of emojis based on the requirement rather than choosing from a finite set of emojis.

This research will help to generate high quality and diverse emojis. It will also help the tech community with the comparison of the different GAN models with different evaluation metrics.

**TABLE OF CONTENTS**

[DEDICATION 2](#_Toc178411173)

[ACKNOWLEDGEMENTS 3](#_Toc178411174)

[ABSTRACT 4](#_Toc178411175)

[LIST OF FIGURES 9](#_Toc178411176)

[LIST OF TABLES 10](#_Toc178411177)

[LIST OF ABBREVIATIONS 11](#_Toc178411178)

[CHAPTER 1 12](#_Toc178411179)

[INTRODUCTION 12](#_Toc178411180)

[1.1 Background of the Study 12](#_Toc178411181)

[1.2 Problem Statement 14](#_Toc178411182)

[1.3 Aim and Objectives 14](#_Toc178411183)

[1.4 Research Questions 15](#_Toc178411184)

[1.5 Scope of the Study 15](#_Toc178411185)

[1.6 Significance of the Study 15](#_Toc178411186)

[1.6.1 Contributions 15](#_Toc178411187)

[1.6.2 Beneficiaries 15](#_Toc178411188)

[1.7 Structure of the study 16](#_Toc178411189)

[CHAPTER 2 17](#_Toc178411190)

[LITERATURE REVIEW 17](#_Toc178411191)

[2.1 Introduction 17](#_Toc178411192)

[2.2 GAN and Applications 17](#_Toc178411193)

[2.2.1 What is Generative Adversarial Networks? 17](#_Toc178411194)

[2.2.2 Basic GAN Architecture and Learning 18](#_Toc178411195)

[2.2.3 Loss function of GAN 19](#_Toc178411196)

[2.2.3 Application of GAN 19](#_Toc178411197)

[2.3 Research on evolution of GAN 20](#_Toc178411198)

[2.3.1 Conditional Generative Adversarial Networks (cGAN) 20](#_Toc178411199)

[2.3.2 Wasserstein Generative Adversarial Networks (WGAN) 21](#_Toc178411200)

[2.3.3 Deep Convolution Generative Adversarial Networks (DCGAN) 22](#_Toc178411201)

[2.3.4 Stacked Generative Adversarial Networks (StackGAN) 23](#_Toc178411202)

[2.3.5 Recurrent Generative Adversarial Networks (RGAN) and Recurrent Conditional Generative Adversarial Networks (RCGAN) 24](#_Toc178411203)

[2.3 Research on Emoji and its Importance 25](#_Toc178411204)

[2.3 Research on Emoji Dataset 25](#_Toc178411205)

[2.4 Research on Emoji Generation 26](#_Toc178411206)

[2.5 Research on Word Embeddings: 27](#_Toc178411207)

[2.5.1 Key Concepts: 27](#_Toc178411208)

[2.5.2 Popular Techniques: 27](#_Toc178411209)

[2.5.2.1 Word2Vec 27](#_Toc178411210)

[2.5.2.2 GloVe (Global Vectors for Word Representation) 28](#_Toc178411211)

[2.5.2.3 FastText 29](#_Toc178411212)

[2.5.2.4 ELMo (Embeddings from Language Models) 29](#_Toc178411213)

[2.5.2.5 BERT (Bidirectional Encoder Representations from Transformers) 30](#_Toc178411214)

[2.7 Discussion 32](#_Toc178411215)

[2.8 Summary 32](#_Toc178411216)

[CHAPTER 3 33](#_Toc178411217)

[RESEARCH METHODOLOGY 33](#_Toc178411218)

[3.1 Research Methodology 33](#_Toc178411219)

[3.1.1 Introduction 33](#_Toc178411220)

[3.1.2 Dataset description 33](#_Toc178411221)

[3.1.3 Data preprocessing 34](#_Toc178411222)

[3.1.4 Transformation 34](#_Toc178411223)

[3.1.5 Models 34](#_Toc178411224)

[3.1.6 High Level Flow Diagram 35](#_Toc178411225)

[3.1.7 Evaluation metrics 35](#_Toc178411226)

[CHAPTER 4 36](#_Toc178411227)

[ANALYSIS 36](#_Toc178411228)

[4.1 Introduction 36](#_Toc178411229)

[4.2 Dataset Description 36](#_Toc178411230)

[4.3 Data Preparation 36](#_Toc178411231)

[4.3.1 Elimination of Variables 36](#_Toc178411232)

[4.3.2 Transformation into Categorical Variables 36](#_Toc178411233)

[4.3.3 Identification of missing values 36](#_Toc178411234)

[4.3.4 Univariate analysis 36](#_Toc178411235)

[4.3.5 Treatment of missing values 36](#_Toc178411236)

[4.3.6 Splitting of original dataset 36](#_Toc178411237)

[4.4 Exploratory Data Analysis (Bivariate analysis) 36](#_Toc178411238)

[4.4.1 Chi-square test 36](#_Toc178411239)

[4.5 Data Visualization 36](#_Toc178411240)

[4.9 Summary 36](#_Toc178411241)

[CHAPTER 5 37](#_Toc178411242)

[RESULTS AND DISCUSSIONS 37](#_Toc178411243)

[5.1 Introduction 37](#_Toc178411244)

[5.2 Interpretation of Visualizations 37](#_Toc178411245)

[5.3 Evaluation of Sampling Methods and Results 37](#_Toc178411246)

[5.4 Testing on Validation Dataset 37](#_Toc178411247)

[5.6 Summary 37](#_Toc178411248)

[CHAPTER 6 38](#_Toc178411249)

[CONCLUSIONS AND RECOMMENDATIONS 38](#_Toc178411250)

[6.1 Introduction 38](#_Toc178411251)

[6.2 Discussion and Conclusion 38](#_Toc178411252)

[6.3 Contribution to knowledge 38](#_Toc178411253)

[6.4 Future Recommendations 38](#_Toc178411254)

[REFERENCES 40](#_Toc178411255)

[APPENDIX A: RESEARCH PROPOSAL 42](#_Toc178411256)

[Abstract 44](#_Toc178411257)

[LIST OF FIGURES 46](#_Toc178411258)

[LIST OF TABLES 47](#_Toc178411259)

[LIST OF ABBREVIATIONS 48](#_Toc178411260)

[1. Background 49](#_Toc178411261)

[2.1 Problem Statement 51](#_Toc178411262)

[2.2 Related Work 52](#_Toc178411263)

[2.2.1 Research on Emoji and its Importance 52](#_Toc178411264)

[2.2.3 Research on Word Embeddings: 53](#_Toc178411265)

[3. Research Questions 55](#_Toc178411266)

[3.1 Research Questions 55](#_Toc178411267)

[4. Aim and Objectives 55](#_Toc178411268)

[5. Significance of the Study 56](#_Toc178411269)

[5.1 Contributions 56](#_Toc178411270)

[5.2 Beneficiaries 56](#_Toc178411271)

[6. Scope of the Study 56](#_Toc178411272)

[7. Research Methodology 57](#_Toc178411273)

[7.1 Introduction 57](#_Toc178411274)

[7.2 Dataset description 57](#_Toc178411275)

[7.3 Data preprocessing 57](#_Toc178411276)

[7.4 Transformation 58](#_Toc178411277)

[7.5 Models 58](#_Toc178411278)

[7.6 High Level Flow Diagram 59](#_Toc178411279)

[7.7 Evaluation metrics 59](#_Toc178411280)

[8. Requirements or Resources 60](#_Toc178411281)

[8.1 Hardware Requirement 60](#_Toc178411282)

[8.2 Software, Web Services and Libraries Requirement 60](#_Toc178411283)

[9. Research Plan 61](#_Toc178411284)

[References 62](#_Toc178411285)

LIST OF FIGURES

[Figure 1 Sample emojis used frequently in informal communication. 10](#_Toc177205080)

[Figure 2 Emoji used in a feedback form 11](#_Toc177205081)

[Figure 3 Dataset sample 18](#_Toc177205082)

[Figure 4 High Level Flow Diagram 20](#_Toc177205083)

[Figure 5 Research Plan 22](#_Toc177205084)

LIST OF TABLES

[Table 1 Related works on Methodologies 15](#_Toc177205071)

[Table 2 State-of-the-art model to be used for emoji generation 20](#_Toc177205072)

LIST OF ABBREVIATIONS

GAN Generative Adversarial Network

VQGAN Vector Quantized Generative Adversarial Network

DCGAN Deep Convolution Generative Adversarial Network

CLIP Contrastive Language-Image Pre-Training

AI Artificial Intelligence

ML Machine Learning

RGB Red Green Blue

RGBA Red Green Blue Alpha

CHAPTER 1

INTRODUCTION

1.1 Background of the Study

Emoji is a visual representation of human interaction based on Human emotions, living beings, objects and symbols. These are in widespread use across the internet such as in textual messaging application, social media platforms which are known as informal modes of communication. Below is an image about the most frequently used emojis during informal modes of communication.

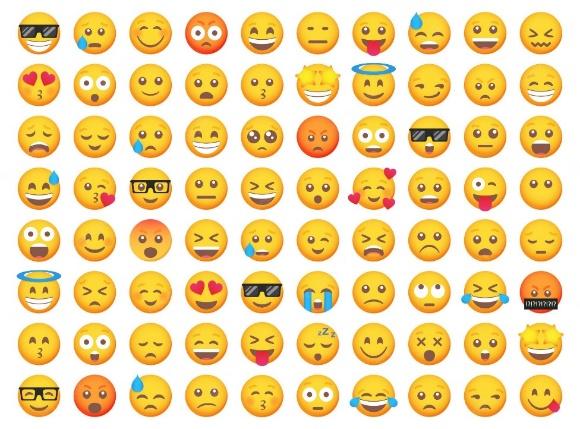


Figure *1* Sample emojis used frequently in informal communication.

According to (Kennison et al., 2024) emojis represent faces with human expressions, objects, animals, and actions.

Emojis are often used to in replacement of words, thus acting as a part of communication and language (Provine et al., 2007). As per the research (Urumutta Hewage et al., 2021) the emojis are static images which are often preserved as human like emotion and expressions. Further, as per (Erle et al., 2022) the communication has greater emotional intensity when we communicated to people using emojis rather than without emojis.

Other application of emojis other than informal communication includes

* Rating/Feedback Form : The feedback forms using emoji instead of stars or numbers. The research has found that the chances of the user giving feedback is higher when emojis are used.
* Providing a Reaction : These days most online applications or websites provide emojis to react to a post, blog, article or tweet.
  + Thumbs up and down emojis are used to like and dislike.
  + Facial emojis are used to express feeling like surprise, wow or shocked.

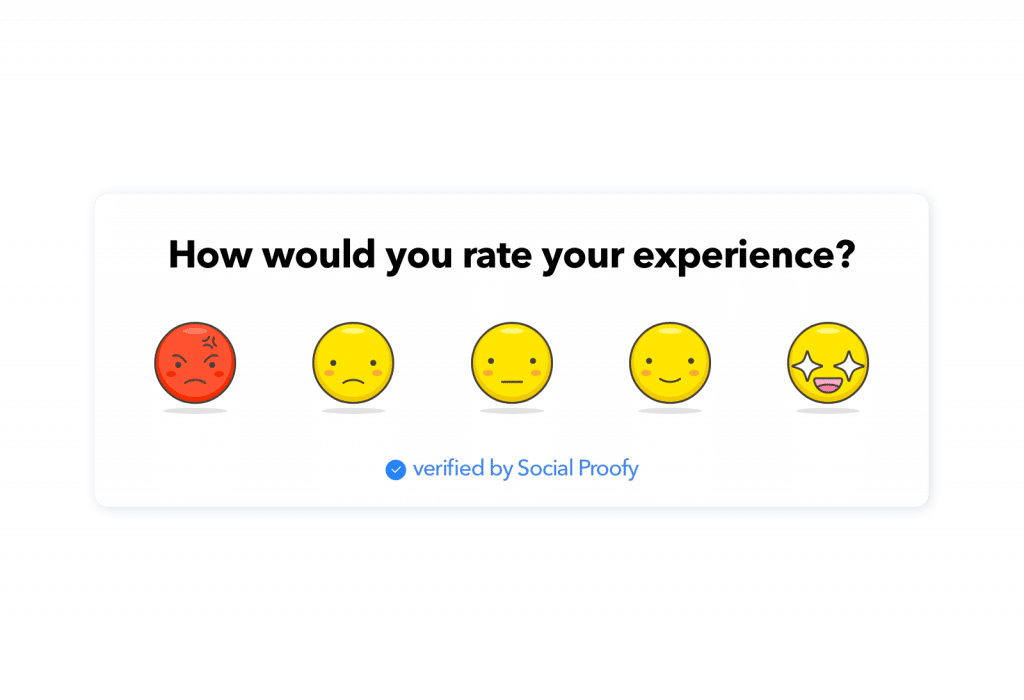


Figure *2* Emoji used in a feedback form

As per the emojipedia.org analysis of over 2.934 billion global tweets between 2017 to 2021, the percentage of tweets with at least one emoji has increased from 13.5% to 21.5%. With the increase of the use of emoji, the demand for newer and customised emoji is increasing. The creation of emoji is a currently is a tedious process of submitting an emoji to Unicode Technical Committee, which takes the final decision for the inclusion, this takes 18-24 months for the complete process Using the state-of-the-art models for generating images using text, we will be able to create new emojis based on the user input.

Prior research has focused on generating facial emojis and emotion based facial emojis. With the state-of-the-art model and architecture, we will be generating not only facial emojis but objects also.

Prior research used techniques like unsupervised learning , Deep Convolution based Generative Adversarial Networks and Conditional Generative Adversarial Networks.

One of the greatest debates on emojis includes whether emoji can be used “as punctuation” or ”after punctuation” which was included in “The Great Emoji Debate”.

In this research we will be creating a state-of-the-art model to generate emojis using emoji content and descriptive information of the emoji and comparing the model with prior models.

We will be looking at the various generative adversarial models :

* VQGAN-CLIP
* TextControl-GAN
* StyleGAN-T

1.2 Problem Statement

Currently the process of creating new emojis is a tedious and time-consuming task which takes a couple of days or months to create and years to get approval from the Unicode Technical Committee. Currently there are only 3,664 emojis as per Unicode standard and the need for need and diverse emoji is increasing. Due to this the current library faces limitations in capturing all the human emotion and experiences. This can lead to miscommunication and inadequate representation of culture and identities.

As per the current trend the use of emoji has been steadily increasing in informal conversation and social media. The need for new emojis to be generated quickly based on the text from the user is high. This issue mostly affects the Gen Z generation as they are the ones who use them mostly during the social media platforms, Messaging apps and content creation. It is important as it can improve the quality of the conversation with improved emotions and objects. This solution will have a huge impact on the conversation by improving the quality of the conversation with new and better emojis which can express emotions and objects accurately.

1.3 Aim and Objectives

The aim of this research is to propose a generative model to generate new emojis is to create a new and diverse design based on textual input from the user and compare it with the baseline and previous models which are visually appealing and useful.

Below are the list of research objectives which will be accomplished in this research:

* To create a generative model that can create new high quality and realistic emojis.
* To generate emoji based on the platform of use.
* To compare between the state-of-the-art generative models.
* To evaluate the performance of the emojis generated using the generative models.
* To finetune the various parameters used in the generative model.

1.4 Research Questions

Generate new realistic emoji in real time based on the textual input from the user.

1.5 Scope of the Study

The scope of a study on generating new emojis are as follows:

* Type of architecture : Generative Adversarial Networks
* Data : <https://huggingface.co/datasets/ChengAoShen/emoji_with_text>
* Input : Short textual input
* Output : 64x64 emoji
* Evaluation techniques :
  + Inception Score (Shmelkov et al., 2018)
  + Fréchet Inception Distance (Shmelkov et al., 2018) (Lee et al., 2023)
  + Learned Perceptual Image Patch Similarity (Lee et al., 2023)
  + Structural Similarity Index (Mittal et al., 2020)

1.6 Significance of the Study

1.6.1 Contributions

* Offer users a wider range of options to accurately express their emotions and experiences using emoji,
* Reflects the variety of individuals and communities.
* Contribute to the development and application of GANs
* Valuable tool for individuals and communities to create custom emojis

1.6.2 Beneficiaries

* Users of social media platforms.
* Content creators like article and blog writers.
* The AI and ML industry.
* Emoji Designers.
* Unicode and emoji organisations like emojipedia.

1.7 Structure of the study

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

GAN or Generative Adversarial Networks belong to generative model family. This model helps to generate new samples from the same distribution from similar sample dataset. Over the last decade, there is significant progress in GAN models. In this chapter we will study the various types of research on GAN and its applications, importance of emoji, various datasets, types and advantages of word embeddings and comparison of various metrics used to test different models.

2.2 GAN and Applications

2.2.1 What is Generative Adversarial Networks?

In 2014 Ian Goodfellow and his colleagues developed the basic GAN model. It was based on zero sum and min max game theory. Here's a simplified explanation of GANs:

* The Forger (Generator): A computer program that starts by making very bad drawings   
  of the emoji
* The Detective (Discriminator): Another computer program that gets shown both real emoji and the Forger's fakes. The Detective tries to tell them apart.

Each time the Detective catches a fake, the Forger learns from its mistakes and tries to make even better fakes. This makes the Detective have to get smarter too.

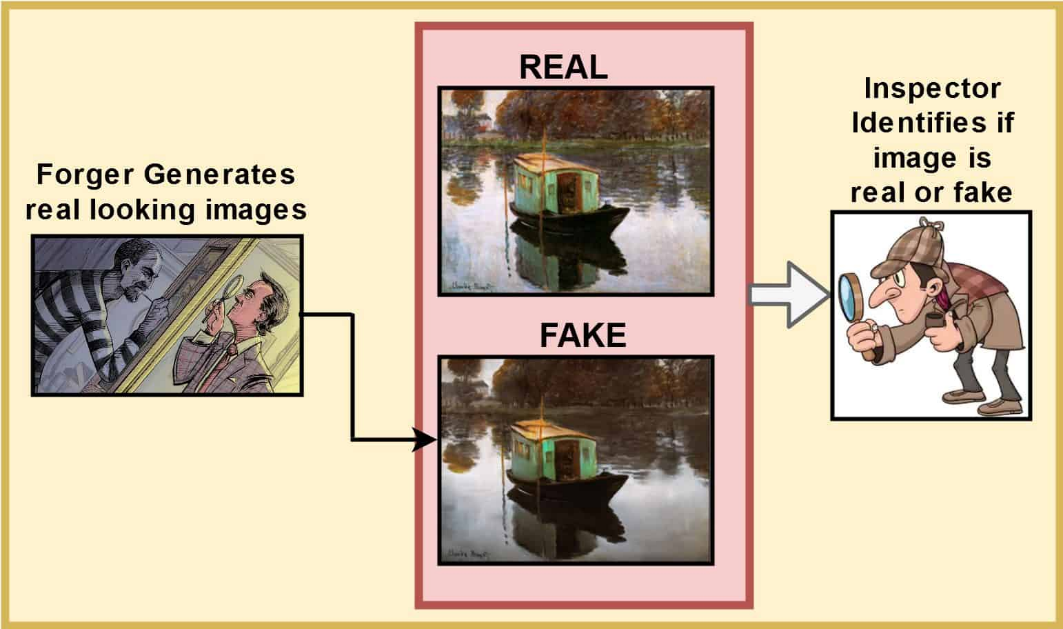


Figure *3* GAN in simplified image

2.2.2 Basic GAN Architecture and Learning

The basic of GAN architecture comprises of one generator and one discriminator. The generator is represented by G and the discriminator is represented by D. The generator takes an input vector z and generate an output which is represented by G(z). Both generator and discriminator consist of neural networks, one to create the required output and other for binary classifier.

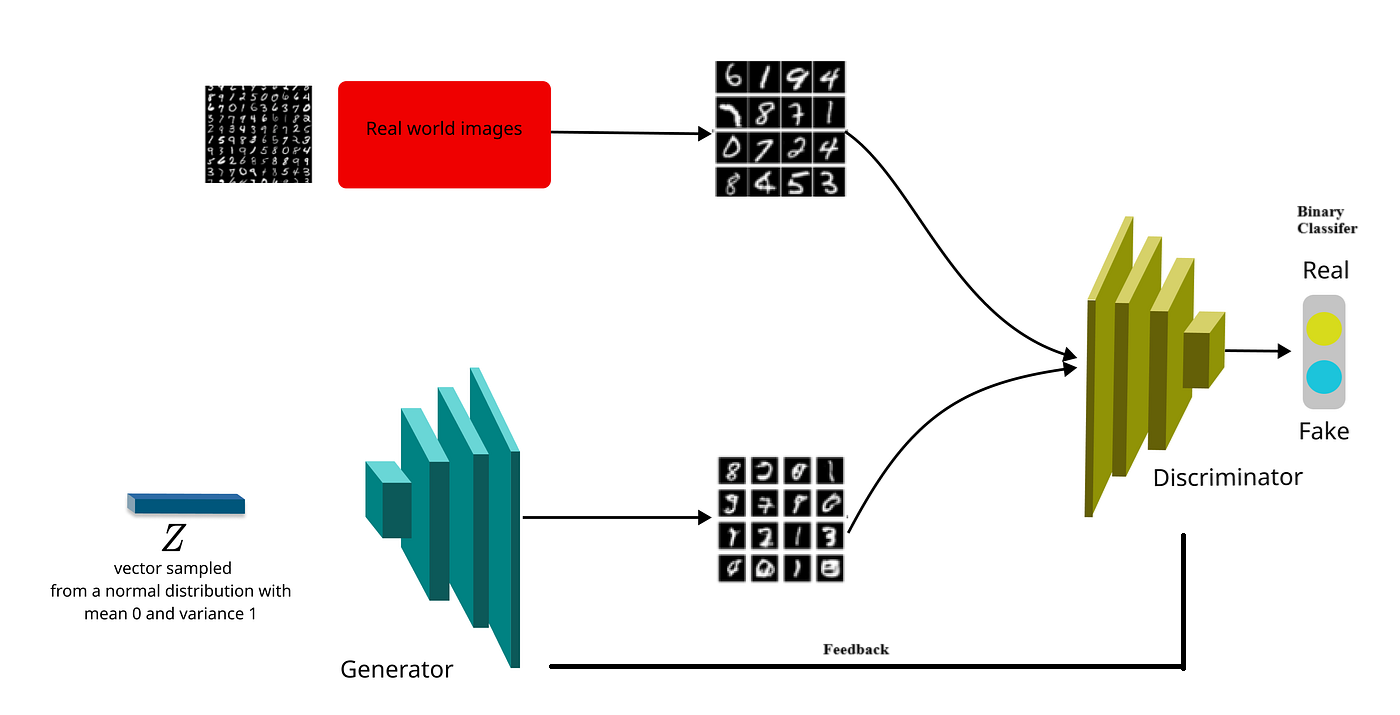


Figure *4* Basic GAN Architecture

The discriminator takes both the real data and the fake data from the generator and tries to decide which one is fake i.e. it models the probability of the fake data is real or not. After each iteration the feedback from the discriminator is sent to the generator to improve the quality of the fake data.

Over a period of time the generator creates better fake data and the discriminator learns identifying the fake images. Both models tries to beat each other due to which the adversarial term come from.

2.2.3 Loss function of GAN

GAN uses minmax loss function to optimize the neural network. The generator minimizes the function and the discriminator maximizes the loss.

mingmaxd V(D,G)

V(D,G) = Ex~pdata(x)[logD(x)] + Ez~pz(z)[log(1 – D(G(z)))]

Ex – Expected value over all real data samples

D(x) – Probability of discriminator if x is real

G(z) – Generator output

z – Random noise

D(G(z)) – Probability of discriminator generated data is real

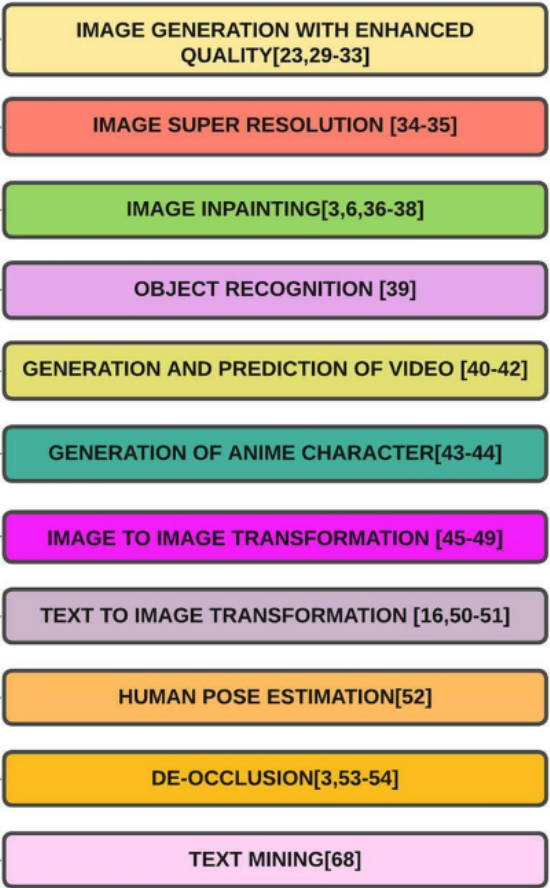
Ez – Expected value over all random data

2.2.3 Application of GAN

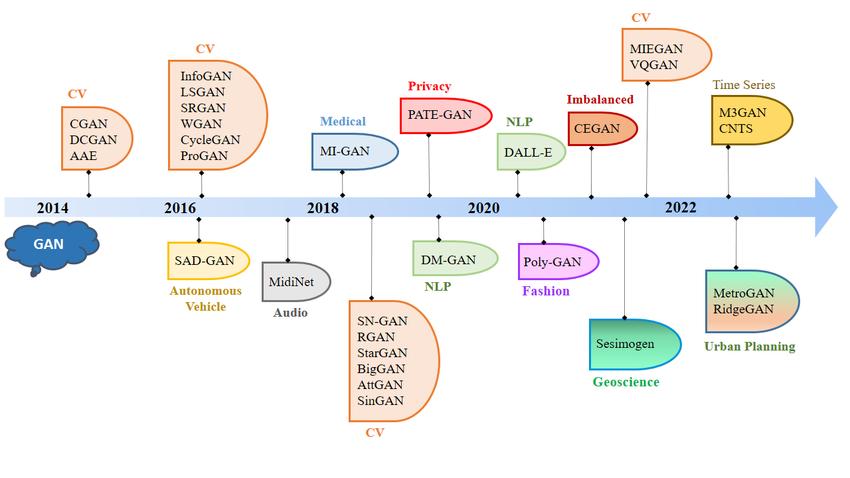
Generative Adversarial Networks has received considerable attention due to their ability to generate high-quality synthetic data.

Below are some basic applications in various industries:

* Image Generation and Enhancement
  + Generate new painting based on certain styles or novel artworks.
  + Increase the image resolution without losing any details.
  + Filling in missing or damaged parts of images.
  + Transfer the style of one image (e.g., a photo) to another (e.g., a Van Gogh painting).
* Video and Animation Generation
  + Generate deep fake videos by swapping the faces or creating new sequence.
  + Generate characters for video games and animations
* Text to Image
  + Generate advertising, creative arts, and virtual worlds based on the textual inputs describing the image to be generated.
* Healthcare
  + Generate synthetic medical images for training professionals and enhancing the diversity of training datasets.
  + Improving the quality of medical images such as MRIs or CT scans from incomplete data.
* Data Augmentation
  + Augment face data for improving face detection and recognition algorithms.
* Creative Content Generation
  + Generate new music tracks or compose original pieces.
  + Generate human-like text.
* Gaming and Simulation
  + Generate realistic scenes for video games which reduces the artists time in manually creating them.
  + Generate realistic physical behaviours for use in visual effects and gaming.



2.3 Research on evolution of GAN

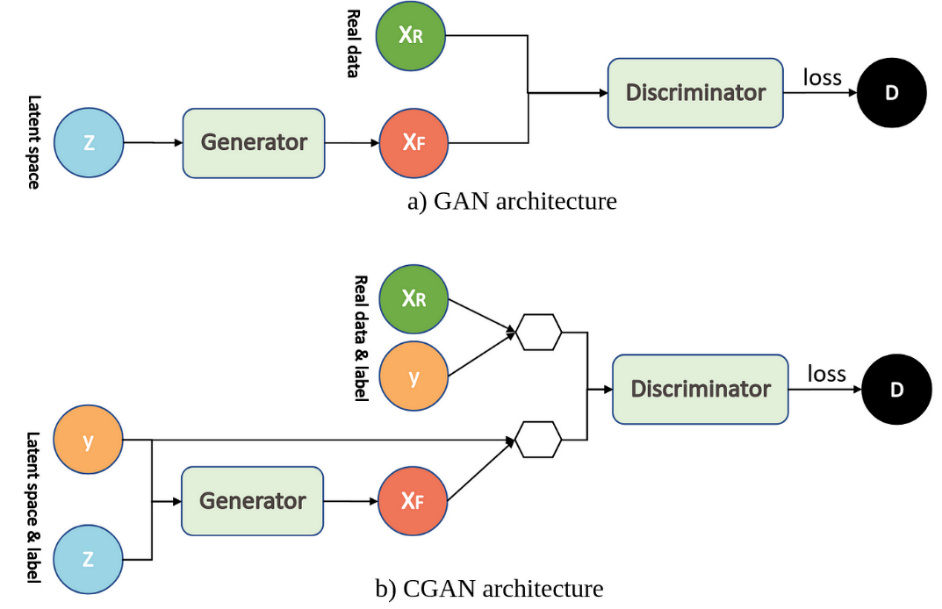


2.3.1 Conditional Generative Adversarial Networks (cGAN)

cGAN are an extension to GAN with additional conditional input along with the noise vector. This gives the generator and discriminator control over the mode of data which is generated. The additional input is denoted by y in the figure below. This can be in any form text, labels or other modalities.

The loss function for cGAN can be represented as:

minG maxD ​f(D,G)=Ex​[log(D(x∣y))]+Ez​[log(1−D(G(z∣y)))]



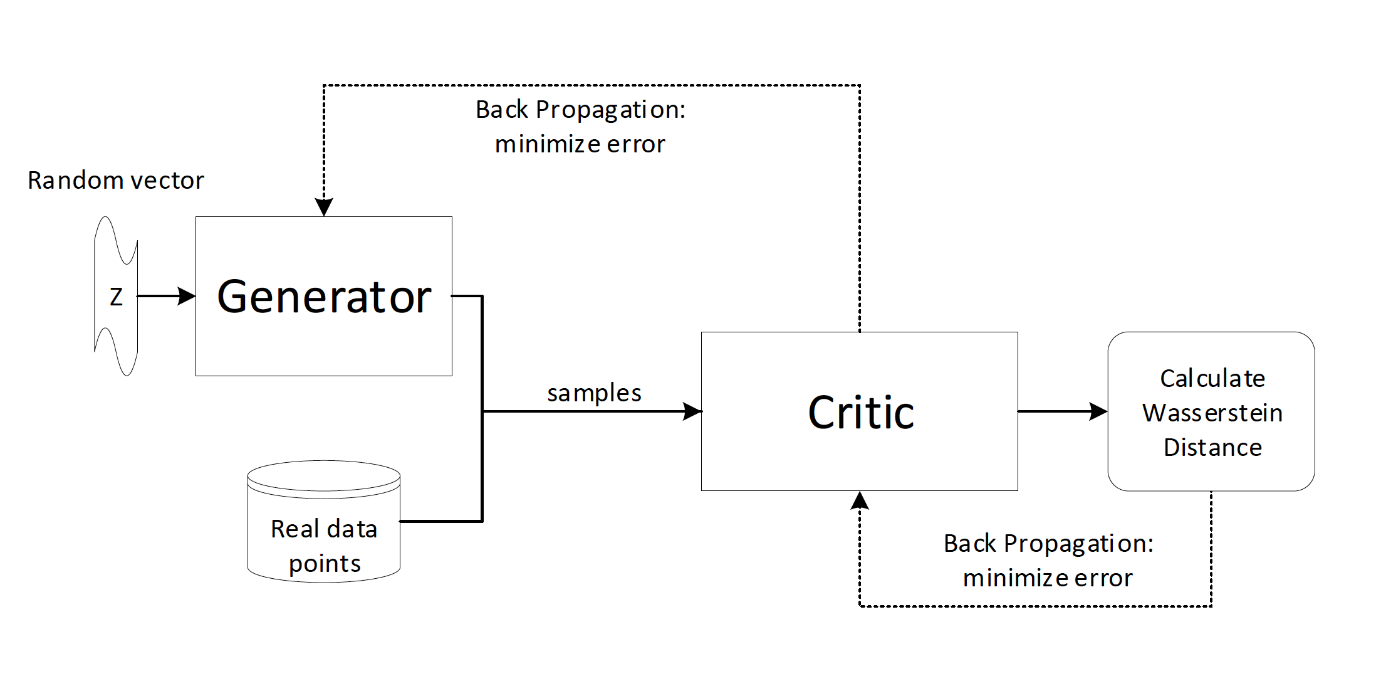
2.3.2 Wasserstein Generative Adversarial Networks (WGAN)

WGAN proposed a new approach to training GAN that addressed some of the inherent challenges of traditional GANs like instability and mode collapse. Instead of using the traditional divergence to measure the difference between the generated and real data it employs the Wasserstein distance. This provides a measure of distance for distributions that may not overlap significantly. It captures how much work is needed to change one distribution into another making it more stable during training.

To ensure that the discriminator/critic maintains the condition required for the Wasserstein distance the weights of the discriminator are constrained to a fixed range which prevents them from growing too big which could lead to instability. The use of Wasserstein distance and weight clipping helps mitigate common problems seen in traditional GAN training, such as mode collapse, where the model generates limited varieties of outputs. WGAN’s framework allows for smoother gradients, leading to more stable learning dynamics.

The loss function for WGAN can be represented as:

minGmaxD ​Ex∼Pr​​[fw​(x)]−Ez∼p(z)​[fw​(G(z))]



2.3.3 Deep Convolution Generative Adversarial Networks (DCGAN)

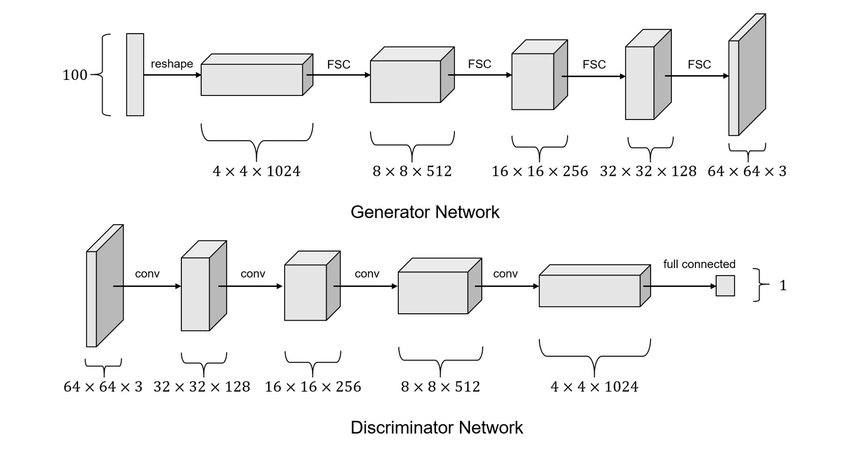
DCGANs use deep convolutional neural networks for both the Generator and Discriminator models. The original GAN architecture used only multi-layer perceptron’s or MLPs.

The key features of the DCGANs neural network architecture are:

* The Generator, convolutions are replaced with transposed convolutions, so the representation at each layer of the Generator is successively larger, as it maps from a low-dimensional latent vector onto a high-dimensional image. Replacing any pooling layers with strided convolutions (Discriminator) and fractional-strided convolutions (Generator).
* Use batch normalization in both the Generator and the Discriminator.
* Use ReLU activation in Generator for all layers except for the output, which uses Tanh. Use LeakyReLU activation in the Discriminator for all layers.
* Adam optimizer instead of SGD with momentum.

The loss function for WGAN can be represented as:

minGmaxD V(D,G) = ​Ex∼pdata(x)​​[logD(x|y)]+Ez∼p(z)​[log(1−D(G(z)))]



2.3.4 Stacked Generative Adversarial Networks (StackGAN)

StackGAN is a framework designed specifically for generating high-resolution images from textual descriptions. It tackles the challenge of producing detailed images while maintaining coherence with the provided textual input.

The Stack GAN consists of the following components:

* The variable input is converted into a fixed length vector embedding.
* The Stage 1 generator takes the textual input and generates a low-resolution image output.
* The Stage 2 generator takes the textual input along with the low-resolution image generated by the Stage 1 and generate a high-resolution image output.
* Stage 1 and Stage 2 discriminator



2.3.5 Recurrent Generative Adversarial Networks (RGAN) and Recurrent Conditional Generative Adversarial Networks (RCGAN)

RGAN and RCGAN is used to generate a sequence of data. It use recurrent neural networks for both the Generator and Discriminator models which is the main difference with the DCGAN. Here the RNNs can be a Long Short-Term Memory (LTSM) or Gated Recurrent Unit (GRU) which captures the temporal dependency of input. In case of RCGAN the generator and discriminator receives an addition auxiliary input. Many experiments show effective results while using RGAN and RCGAN for time series data.

The loss function for RCGAN can be represented as:

Dloss(Xn, yn) = -CE(RNND(Xn), yn)

Gloss (Zn) = Dloss (RNNG(Zn), 1) = -CE(RNND(RNNG(Zn)), 1)

Where Xn and yn are the input and zn is the random noise vector



2.3 Research on Emoji and its Importance

Emoji despite being more common in digital communication, is often dependent on characteristics, motives for using them and the context in which they are used (Cavalheiro et al., 2023). Over the past couple of years there has been extensive research in the use and importance of emojis in every life activity. Below are few of the recent research with description and findings:

* The research by (Cavalheiro et al., 2023) try to find the relationship of frequency of the emoji with relationship and individual characteristics. The research also found personal reasons to be one of the relates to high emoji frequency. The research also found age to be directly correlated with emoji frequency which was unexpected.
* The research by (Erle et al., 2022) discusses a creation of emojis in digital communication with the help of their model. The research also found that the emojis can depict the facial expression of the human.
* The research by (Lefebvre et al., 2024) is about the influence of emoji during tip suggestion with the tipping percentage. The research also found significant positive effects of emoji during the tipping.
* The research by (Kennison et al., 2024) relates the personal characteristics with the emojis used in the conversation. The research also found that the men used emojis less often than women. The research also found that users may me using emojis in place of word and used less dictionary words.

2.3 Research on Emoji Dataset

There are very limited datasets for emojis which are labelled and have quality images. Below are the datasets considered which were considered during the start of the research.

* Emoji-dataset on Hugging Face by valhalla

<https://huggingface.co/datasets/valhalla/emoji-dataset>

* Full Emoji Database on Kaggle by ELIAS DABBAS  
  <https://www.kaggle.com/datasets/eliasdabbas/emoji-data-descriptions-codepoints/data>
* Emoji-with-text on Hugging Face by ChengAoShen

<https://huggingface.co/datasets/ChengAoShen/emoji_with_text>

Finalized the 3rd dataset as the quality of the images was good and every image had their respective label.

2.4 Research on Emoji Generation

With the evolution generative models and the importance of emoji there has been couple of research done in the field of emoji generation. It has been noticed that most of the research has been around facial emojis and with a very limited set of them. Below are few of the recent research with description:

* The research (Mittal et al., 2020) uses supervised learning with multimodal input to GAN with U-Net like architecture for the generator to generate realistic hand drawn emojis. This research even though was able to generate the emojis it required the user to draw a rough sketch with the emotion to be given as input, also the dataset used had to be augmented from the facial emojis which don’t depict the real dataset.
* The research (Lee et al., 2023) creates a model which can generate emojis based on emotional degree for more complex and detailed usage on online conversations (Lee et al., 2023). The research uses conditional GAN for the experiment and also compares it with other baseline models. This
* The research (Tang, 2023) used unsupervised learning to generate new emojis using DCGAN which is a combination of Convolution Neural Network and Generative Adversarial Network. This research faced many challenges like after 1000 epochs the accuracy flattens out. The emojis generated where quite blurry and the user had no control over the emojis generated due to unsupervised learning.
* The research (Xu, 2021) used unsupervised learning with optimization techniques with Adam Optimizer to generate new emojis.
* The research (Peng and Zhao, 2021) used an encoder–decoder model, to predict the sequence of emojis based on short text. This research was able to correlate the emojis generated. The model was also able to learn the semantics between the emoji and text description.

Although the above research where able to generate emojis but most of this research had sparse data and limited the scope to generate facial emojis.

2.5 Research on Word Embeddings:

According to the research by (Asudani et al., 2023) the first preference should be given to domain specific embeddings. The research also compares the performance of the various word embedding techniques.

Word embeddings are word representation as vectors in a continuous vector space. This captures semantic relationships between words, making it easier for models to process text. Here’s concepts and techniques used in word embeddings:

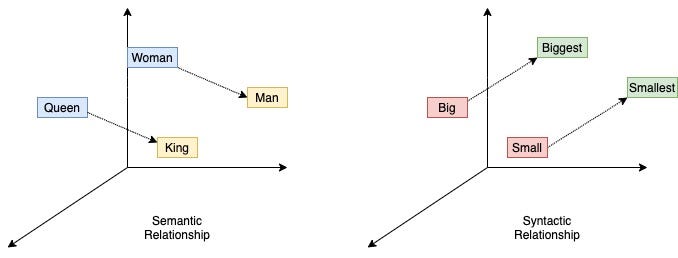
2.5.1 Key Concepts:

* Vector Space Model: In this model, words are mapped to points in a high-dimensional space, where similar words are located close to each other.
* Dimensionality: Commonly, embeddings are created with dimensions ranging from 50 to 300, balancing performance and computational efficiency.
* Semantic Relationships: Word embeddings can capture various relationships, such as:
  + Synonymy (e.g., "king" and "queen")
  + Analogies (e.g., "man" is to "woman" as "king" is to "queen")

2.5.2 Popular Techniques:

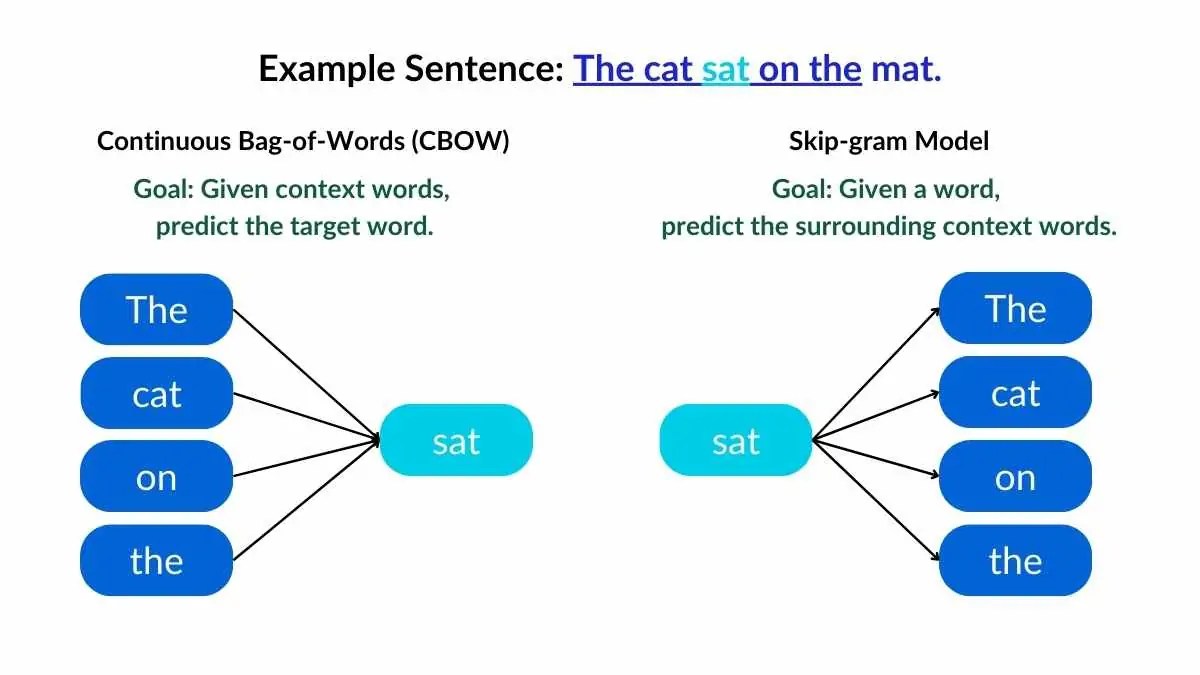
2.5.2.1 Word2Vec

It is a popular technique for natural language processing that transforms words into numerical vectors. Developed by a team at Google led by Tomas Mikolov, Word2Vec captures semantic relationships between words, enabling computers to understand text better.



The two training models :

* CBOW (Continuous Bag of Words) - This model used for learning word embeddings, which are dense vector representations of words that capture their semantic and syntactic properties.
* Skip-Gram – This model uses a target word to predict its surrounding context words. It’s particularly effective for learning rare words.

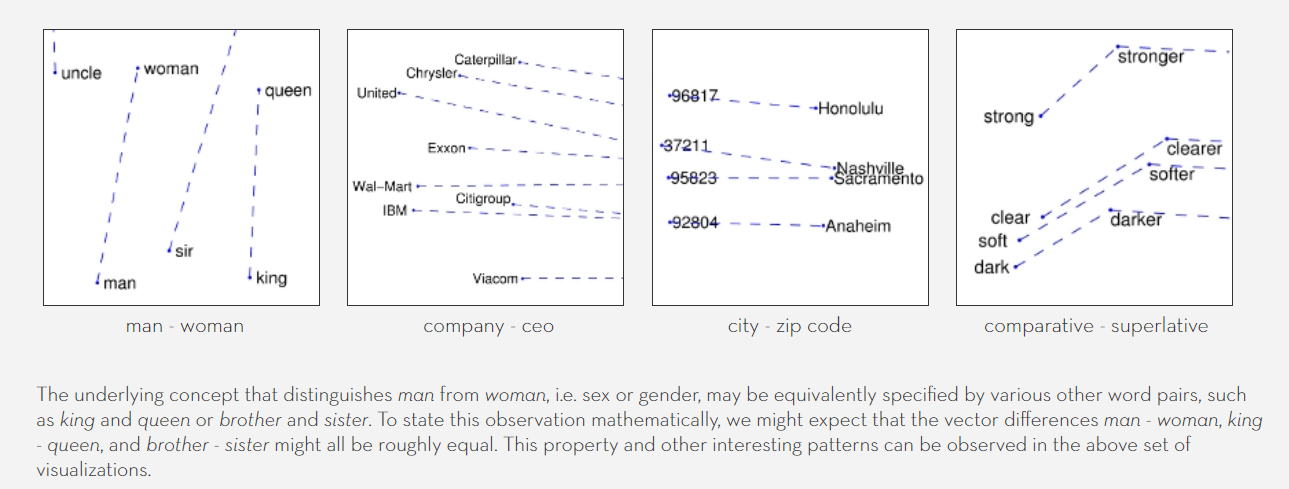


2.5.2.2 GloVe (Global Vectors for Word Representation)

Global Vectors for Word Representation (GloVe) is developed by researchers at Stanford. It is an unsupervised learning algorithm for obtaining vector representations for words. It captures the global statistical information of a sentence in a meaningful way.

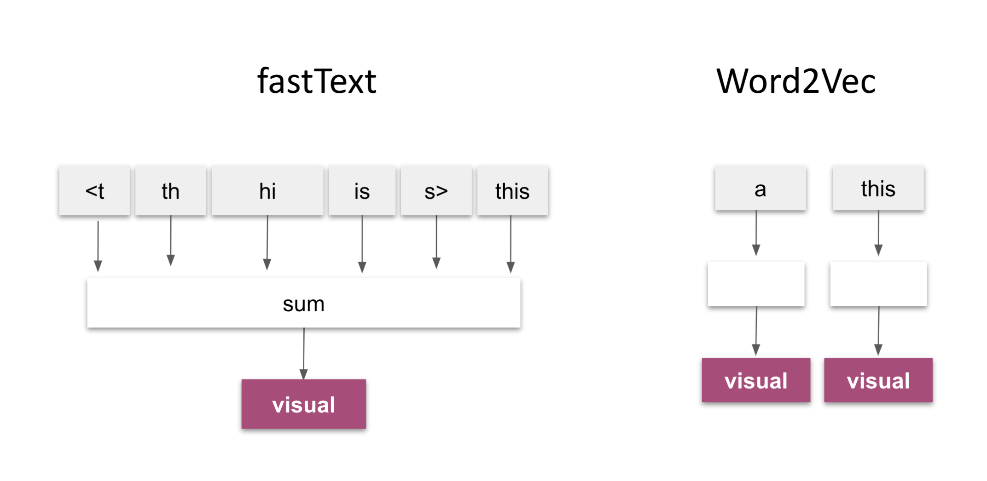
GloVe's use of global co-occurrence information allows it to generate embeddings that reflect broader semantic relationships. It uses euclidean distance or  cosine similarity to measuring the linguistic or semantic similarity of the corresponding words.

The output word vectors perform well on various NLP tasks, including analogy and similarity tasks.



2.5.2.3 FastText

FastText is a sophisticated Word embedding model recently introduced by Facebooks AI Research(FAIR) team performed substantially better than traditional word embeddings by considering subwords. It represents words as a bag of character n-grams (e.g., for the word "apple" it would use "app", "ppl", and "le"). That makes it learn the morphology of word construction and based on its pieces what an English word means. For each word, the model averages the vectors of its n-grams to create a vector for the word, which allows it to embed words that potentially were not even observed during training.



2.5.2.4 ELMo (Embeddings from Language Models)

ELMo is a deep learning model created by the Allen Institute for AI that generates word embeddings. These word embeddings are deep contextualized word representations based on a pre-trained bidirectional language model biLMs.Unlike traditional word embeddings, it captures context in a more sophisticated way.

It creates embeddings based on context of the word. This means that the same word can have different representations depending on its usage in a sentence.

ELMo utilizes a bidirectional Long Short-Term Memory network where it efficiently uses all layers of the bidirectional language model.

ELMo is trained on 1 billion tokens due to which it generates state-of-the-art embedding with contextually aware word representations.

2.5.2.5 BERT (Bidirectional Encoder Representations from Transformers)

BERT is developed by Google and is know for the versatility and performance with benchmark results on GLUE and SQuAD datasets.

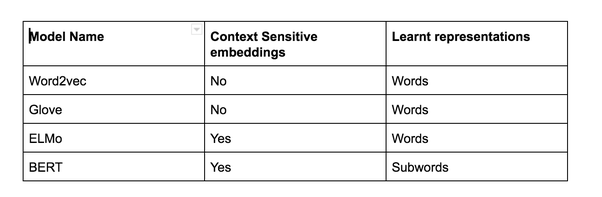
It processes text in both directions simultaneously which allows it to capture the full context of a word based on all surrounding words, increasing its understanding of language.

Its encoder utilize self-attention mechanisms which enable the model to weigh the importance of different words in a sentence leading to better representations.

It is trained on large datasets like Wikipedia and BooksCorpus and can be fine-tuned on specific downstream tasks like sentiment analysis and question answering.

It also produces dynamic embeddings based on the context of the word similar to ELMo

Below is the comparison between the different embeddings :

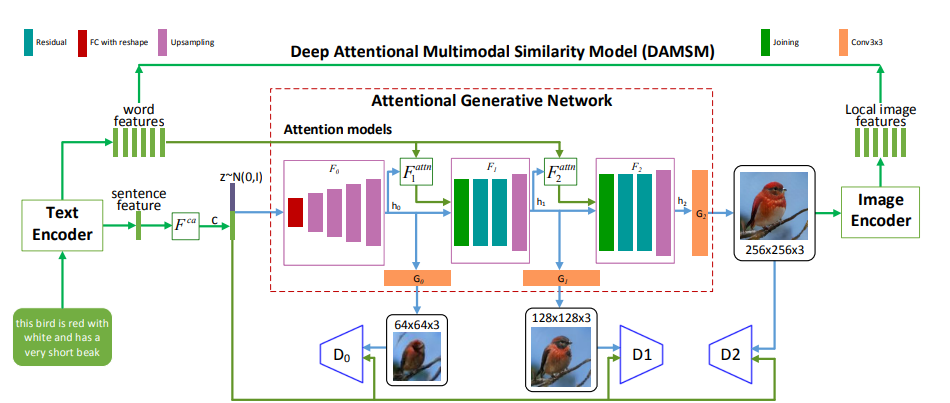


**2.6 Research on Methodology Specific to Text to Image Generation**

Below are the recent methodologies in the text to image domain developed on top of generative adversarial network architecture.

2.6.1 Attentional Generative Adversarial Network (Xu et al., 2018)

Attn-GAN proposed a new GAN with multistage refinement using attention. Each of this multistage attention model gets the most relevant word vectors for generating sub-regions of the image (Xu et al., 2018). This improves the model’s ability to handle complex details and generates more accurate visual elements tied to specific words. This is used to generate images from text input.



In the research a Stack-GAN was used to increase the resolution of the images generated.

Form the above figure we can see text encoder on the left, attention models in the middle, stacked generators in bottom middle and a discriminator on the right side.

DAMSM ensures that the generated image matches the text by comparing the image features the text features.

The AttnGAN significantly outperformed previous state-of-the-art GAN models, boosting the best reported inception score by 14.14% on the CUB dataset and 170.25% on the more challenging COCO dataset.

2.6.2 Stack- Generative Adversarial Network (Dhivya and Navas, 2020)

The research proposes the use of multiple GAN stacked on top of each other to generate high resolution photo realistic images (Trevisan de Souza et al., 2023). This is used to generate images from text input.



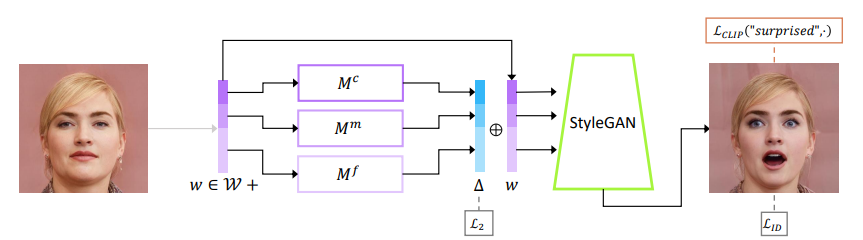
The architecture is divided into two main stages, where each stage focuses on generating and refining the image progressively. Form the above figure we can see Conditioning Augmentation on the left side, stage-I generator in the top centre stage-II generator in bottom centre and the discriminator pair in the right.

The two-stage process allowed the model to first generate a rough image and then progressively refine it which improved the overall quality and alignment with the text description. Also, the Conditioning Augmentation introduced variability in the generated images while still adhering to the given text helping improve diversity.

The Stack-GAN significantly help the other GANs to generate high quality images.

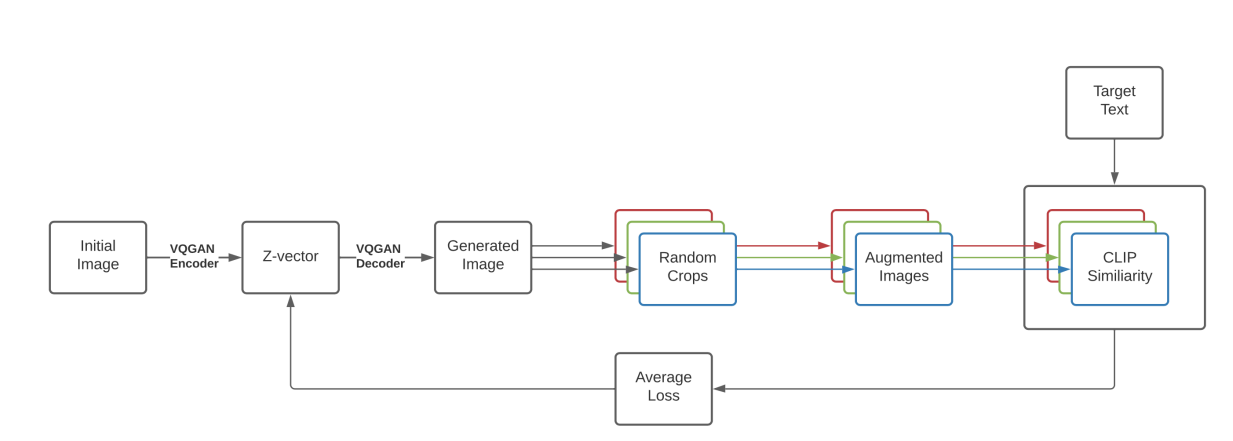
2.6.3 Style-CLIP (Patashnik et al., 2021)

This research proposes CLIP embeddings to allow image editing. The CLIP embeddings share the same latent space hence we can directly compare the text with the images (Trevisan de Souza et al., 2023). This is used to edit images based on text input.



2.6.4 VQGAN-CLIP (Crowson et al., 2022)

This research proposes a model where an encoder is used to create a latent vector which is used by the decoder to create the image. It also uses CLIP embedding with optimization techniques to have high similarity with the text prompt. (Trevisan de Souza et al., 2023). This is used to generate images from text input.



VQGAN-CLIP is a model that combines the powerful image generation capabilities of VQGAN with the understanding of CLIP. This combination allows users to generate excellent images from textual descriptions. Its ability to generate high-quality, creative visuals from text prompts has made it a favorite in the AI art community.

2.6.6 StyleGAN-T (Sauer et al., 2023)

This research proposes a model build from Style GAN-XL mentioned before. The network size of the model is increased to include text information from CLIP encoders.

(Trevisan de Souza et al., 2023). This is used to generate images from text input.

It was created to optimized for fast, large-scale text-to-image synthesis. It is built on the foundation of StyleGAN-XL but introduces significant architectural changes to the generator and discriminator to improve performance in text-conditioned image generation.

Architecture included residual convolutions to stabilize training and increase capacity.



This architecture combined the strengths of StyleGAN with the CLIP model's text guidance. It leverages a Transformer-based discriminator for improved realism and text alignment in image generation. This design allows the model to generate high-quality, text-aligned images with fine details, especially when complex prompts are provided.

2.8 Research on Evaluation metrics for GAN

We can measure the image quality the following metrics are used

* Inception Score (Shmelkov et al., 2018)

It was proposed in 2016 by Salimnas et al where the pretrained InceptionNet model was used to capture the properties of the generated samples. It is used to measure the quality of the images generated by comparing it with the real images from the dataset.

It calculates the KL divergence between the conditional label distribution of sample and marginal distribution of all samples. It access two characteristics mainly one image quality and another image diversity. Inception Score can be represented as :

Exp(Ex[KL(p(y|x)||p(y))])

Where :

p(y|x) – conditional label distribution

p(y) – marginal distribution

KL – Kullback-Leibler

* Fréchet Inception Distance (Shmelkov et al., 2018) (Lee et al., 2023)

It was proposed in 2017 by M Heusel. It is the measure to check how real the image generated are. The score tells how far the generated data vector is from real data vector. InceptionNet model is used to capture and embed features of an input image.

A multivariant Gaussian is used to summarize the embeddings by calculating mean and covariance for generated and real data. FID can be represented as :

FID(r,g) = ||ur-ug||22 + Tr(∑r + ∑g – 2((∑r∑g)1/2)

Where :

(ur, ∑r) – empirical mean and empirical covariance of real data

(ug, ∑g) – empirical mean and empirical covariance of generated data

* Learned Perceptual Image Patch Similarity (Lee et al., 2023)

It was proposed in 2018 by proposed by Richard Zhang and his collaborators. It evaluates the similarity between images which is particularly useful in tasks involving image generation. It is the distance measured between the patches in the latent space.

LPIPS can be represented as :

LPIPS(I1​,I2​)= ∑1​ wl​⋅d(fl(I1​),fl(I2​))

Where :

l – Indexes the layers of the CNN

wl – Learned weights for each layer, which adjust the importance of features extracted from different layers

d(.) – Distance metric (often L2 distance) computed on the feature maps

* Structural Similarity Index (Mittal et al., 2020)

It measures the degradation of the image quality

2.7 Challenges in GAN

Though Generative Adversarial Networks have made significant contribution in various applications but they face several challenges some of them are -

* The generator produces a limited variety of outputs, focusing on a few modes of the data distribution while ignoring others.
* GANs is very difficult to train. The balance between the generator and discriminator is very crucial and can be disrupted .
* The performance of GANs is very sensitive to its hyperparameters, such as learning rates, batch sizes, and network architectures which making tuning a challenging task.
* Training GAN requires huge resources in both RAM and VRAM.
* The discriminators can overfit on training data which makes it harder for the generator to learn.

Addressing these challenges continues to be an active area of research in the field of machine learning and deep learning.

2.8 Discussion

Current discussions around Generative Adversarial Networks (GANs) in 2023 and beyond focus on several key areas of research and application:

* Innovate GAN architectures like DALL E and DALL E 2 which used CLIP to generate high quality images.
* Improving GAN Stability and Performance such as mode collapse and vanishing gradients. Solutions such as spectral normalization, Wasserstein loss, and adaptive learning schedules are being further refined.
* Multi-scale GAN architectures like BigGAN and StyleGAN continue to set benchmarks in quality and diversity of generated images.
* Ethical Concerns and Responsible AI is one of the most disscusion areas particularly after the rise of deepfakes. Discussions in 2023 emphasize the need for better detection techniques, regulatory frameworks, and responsible AI use cases to mitigate harmful applications.

2.9 Summary

TODO

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Introduction

In this chapter we will be going through the end-to-end 2-stage Stacked Generated Adversarial Network architecture along with data preprocessing, data cleaning, data transformation techniques used and the evaluation metrics used in the research.

3.2 Research Methodology

In this section, we will cover the design experiments to generate emoji using textual input.

3.2.3 Data preprocessing

Preprocessing data is a crucial phase in the data mining process. It describes the processes of preparing data for analysis through integration, transformation, and cleaning. Enhancing the data's quality and adapting it to the particular data mining task is the aim of data preprocessing.

3.2.3.1 Data Transformation

This entails formatting the data so that it can be analysed properly. Normalization, standardization, and discretization are common methods used in data transformation. While standardization is used to change the data to have a zero mean and unit variance, normalization is used to scale the data to a common range. Continuous data can be discretized using the discretization process.

3.2.3.1.1 Normalization and Scaling

Transforming data values into a standard range or scale. Scaling values between a specified range (e.g., 0 to 1) is known as min-max normalization. Standardizing data to have a mean of 0 and a standard deviation of 1 is known as Z-score standardization. Decimal scaling, z-score normalization, and min-max normalization are some of the methods.

3.2.3.1.2 Latent Space Manipulation  
Latent space is the hidden layers in the architecture which can be seen as compressed input. As proposed in (Yang et al., 2021) we can use latent space factorization or manipulations to change the attributes of the image.

3.2.3.1.3 Word to Vector

The text description for the emojis will be converted to vector representation.  
They capture the meaning and their usage.

3.2.3.1.4 Clip embedding

The images and text vectors share the same latent space which is helpful in direct comparison of the both

3.2.3.2 Data Integration

This entails merging information from several sources to produce a single dataset. Because it involves handling data with various formats, structures, and semantics, data integration can be difficult. Data integration can be accomplished by using methods like record linkage and data fusion.

3.2.3.3 Data Cleaning

This entails locating and fixing mistakes or discrepancies in the data, including duplicates, outliers, and missing values. Data cleaning can be accomplished with a variety of methods, including imputation, removal, and transformation.

3.2.3.4 Data Reduction

This entails shrinking the dataset without sacrificing any of the crucial data. Techniques like feature selection and feature extraction can be used to reduce data. While feature extraction entails converting the data into a lower-dimensional space while maintaining the crucial information, feature selection entails choosing a subset of pertinent features from the dataset.

3.2.3.5 Data Normalization

Scaling the data to a common range, like between 0 and 1 or -1 and 1, is required for this. To manage data with various scales and units, normalization is frequently utilized. Decimal scaling, z-score normalization, and min-max normalization are examples of common normalization methods.

3.2.4 Architecture

3.2.4.1 Introduction

Here's a simplified explanation of GANs:

* The Forger (Generator): A computer program that starts by making very bad drawings   
  of the emoji
* The Detective (Discriminator): Another computer program that gets shown both real emoji and the Forger's fakes. The Detective tries to tell them apart.

Each time the Detective catches a fake, the Forger learns from its mistakes and tries to make even better fakes. This makes the Detective have to get smarter too.

3.2.4.2 High-Level Flow Diagram

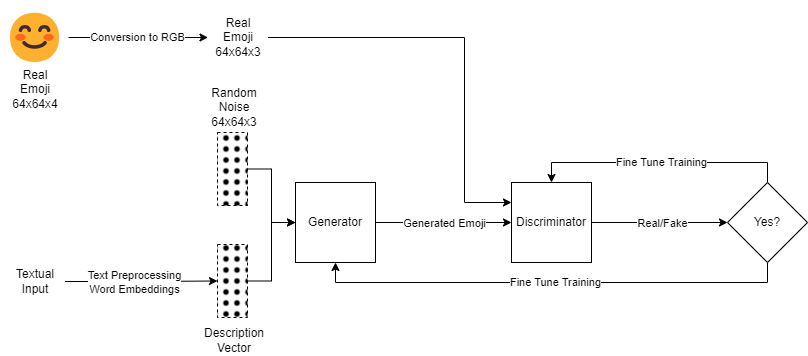


Figure *5* High Level Flow Diagram

3.2.4.3 Generator

The generator creates synthetic data (such pictures) using a random noise vector, which is usually sampled from a Gaussian or uniform distribution. In an attempt to "fool" the discriminator, its objective is to provide outputs that are as near to genuine data as feasible.

The generator gains the ability to map the distribution of noise into the distribution of data (cat pictures, for example).

In a GAN, the generator's goal is to create artificial samples that are convincing enough to trick the discriminator. To do this, the generator minimizes its loss function, JG. When the discriminator has a high likelihood of classifying the produced samples as real, or when the log probability is maximized, the loss is reduced. The equation that follows is provided below:

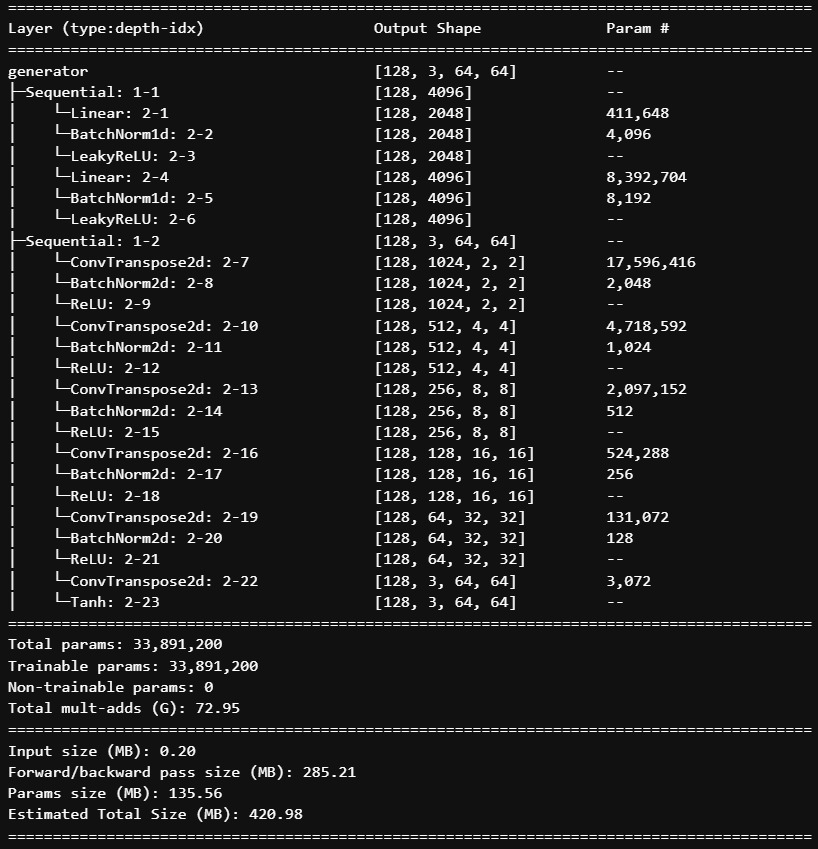
JG​=−1/m Σi=1 m​logD(G(zi​))

Where,

JG​ - measure how well the generator is fooling the discriminator.

log 𝐷(𝐺(𝑧𝑖)) - log probability of the discriminator being correct for generated samples.

3.2.4.3.1 Stage I Generator



3.2.4.4 Discriminator

By differentiating between genuine data (from the actual dataset) and bogus data (from the generator), the discriminator serves as a classifier.

Its objective is to accurately identify if the input is bogus or real.

Feedback about how "real" or "fake" the created data seems is given to the generator via the discriminator.

The discriminator reduces the negative log likelihood of correctly classifying both produced and real samples. With the following equation, this loss motivates the discriminator to correctly classify produced samples as real and bogus samples.

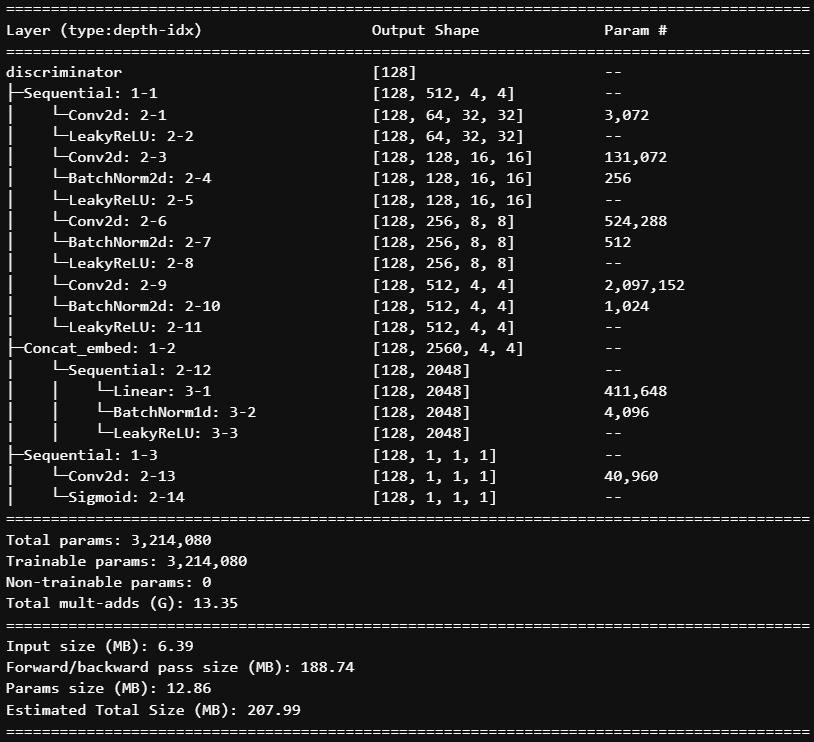
JD​=−1/m Σi=1 m​logD(xi​)–1/m ​Σi=1m​log(1–D(G(zi​))

Where,

JD​ - assesses the discriminator’s ability to discern between produced and actual samples.

𝑙𝑜𝑔𝐷(𝑥𝑖) - log likelihood that the discriminator will accurately categorize real data.

𝑙𝑜𝑔(1−𝐷(𝐺(𝑧𝑖))) - log chance that the discriminator would correctly categorize generated samples as fake.



3.2.5 Hyperparameter Tuning

The process of choosing the ideal set of parameters for a model is known as hyperparameter tuning. These parameters are manually set prior to training; they are not learned from the data during training.

3.2.5.1 Why Hyperparameter Tuning Is Important

A model's performance can be greatly enhanced by fine-tuning its hyperparameters, which can result in improved outcomes for tasks like generation, regression, and classification. By preventing overfitting, optimal hyperparameters guarantee that the model will perform well when applied to previously unseen data. By preventing pointless training runs, effective hyperparameter tuning can conserve computer resources.

3.2.5.2 Typical Hyperparameters:

* Learning Rate: Determines how big of an optimization step to take.
* Batch Size: The quantity of samples handled during every training cycle.
* Epochs: The quantity of full dataset iterations.
* Regularization: To avoid overfitting, employ strategies such as L1 or L2 regularization.
* The number of layers, neurons per layer, and activation functions that make up a network architecture.
* Optimizer: Model weight updating algorithm (e.g., SGD, Adam, RMSprop).

3.2.5.3 Hyperparameter Tuning Techniques

* Grid Search: A brute-force approach where you try all combinations of hyperparameters within a specified grid.
* Random Search: A more efficient method where you randomly sample hyperparameters from a specified distribution.

3.2.6 Training and Evaluation

3.2.6.1 Training Process

A generator and a discriminator neural network compete with one another during the training process of a Generative Adversarial Network (GAN). Below is a summary of the essential steps:

3.2.6.1.1 Initialization

* Set the weights of the generator and discriminator networks at random.
* Choose the number of epochs, batch size, learning rate, and other pertinent hyperparameters.

3.2.6.1.3 Data Preparation

* Make sure your training data is loaded in a format that the GAN can understand.
* To increase training stability, normalize the data to a specified range (0 to 1).

3.2.6.1.4 Training Loop

For each epoch:

1. **Sample Real Data:** Retrieve a batch of real samples from the training dataset.
2. **Generate Fake Data:** Feed random noise into the generator to generate a batch of fake samples.
3. **Train Discriminator:**
   1. Feed both real and fake samples to the discriminator.
   2. Calculate the discriminator loss using a binary cross-entropy loss function.
   3. Update the discriminator's weights using an optimizer (e.g., Adam, RMSprop).
4. **Train Generator:**
   1. Feed random noise into the generator to generate fake samples.
   2. Calculate the generator loss using a binary cross-entropy loss function, aiming to fool the discriminator.
   3. Update the generator's weights using an optimizer.
      * + 1. Evaluation:

Employ metrics to measure the image quality :

* Inception Score (Shmelkov et al., 2018)   
  It is used to measure the quality of the images generated by comparing it with the real images from the dataset.
* Fréchet Inception Distance (Shmelkov et al., 2018) (Lee et al., 2023)  
  It is the measure to check how real the images generated are.
* Structural Similarity Index (Mittal et al., 2020)  
  It measures the degradation of the image quality.
* Human Evaluation  
  Periodically visualize the generated samples to assess their quality and progress.

3.2.6.3 Model Selection

In this research, we will be using a StackGAN model where we will be comparing the emojis generated by Stage 1 with the emojis generated by Stage 2.

The StackGAN architecture's hierarchical structure and capacity to generate high-quality outputs make it an appealing option for emoji creation. The multi-stage approach of this model enables the generation of images to be refined gradually beginning with a coarse representation and progressively adding fine details.

Furthermore, StackGAN's capacity to produce high-resolution images is especially helpful for creating emojis because it produces symbols that are aesthetically pleasing and easily identifiable. The model's versatility further improves its fit for this task by enabling us to experiment with a large variety of emoji styles and designs.

3.3 Resource Requirements

3.3.1 Hardware Requirements

Initial research done on a laptop with the following configuration

CPU : i7 8th Gen

RAM : 16GB

GPU : 4GB Nvidia GPU

The later part of the research was done in a Jarvis cloud VM with the following configuration

RAM: 32GB

GPU: 16GB Nvidia GPU

3.3.2 Software Requirements

* Python
* PyTorch
* Matplotlib
* Pandas
* Numpy
* Easydict
* Datasets
* Pillow
* Wordcloud
* Nltk
* opencv-python
* Jupyter Notebook and Jupyter Lab
* Git Version Control

3.4 Summary

TODO

CHAPTER 4

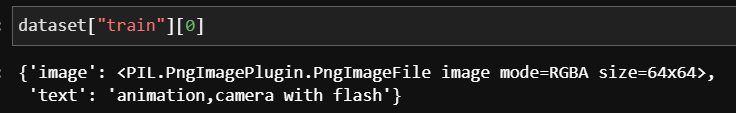
IMPLEMENTATION AND ANALYSIS

4.1 Introduction

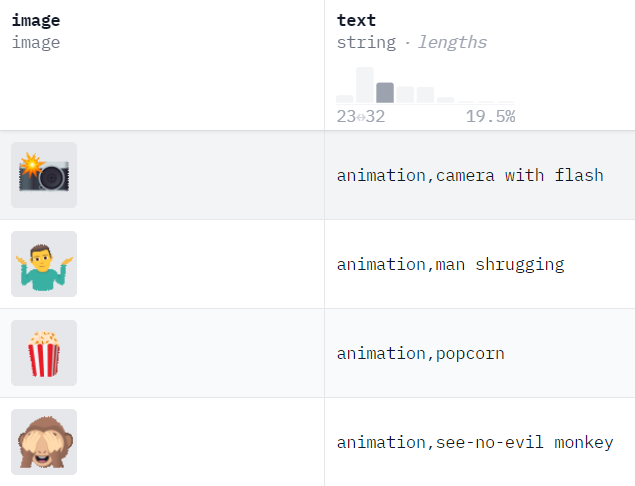
4.2 Dataset Description

The dataset (ChengAo Shen and Siyuan Mu, 2023) is uploaded on Hugging Face. The dataset was in parquet format which is an open-source column-oriented data storage known for efficiently storing and retrieving large volumes of complex data.

This dataset consists of 47192 rows and 2 columns (image, text). Each row consisted of an emoji and its text description.



Each image is of shape 64\*64 and has 4 channels (RGBA) and the text description format is app or company, emoji content and description information which are comma separated.



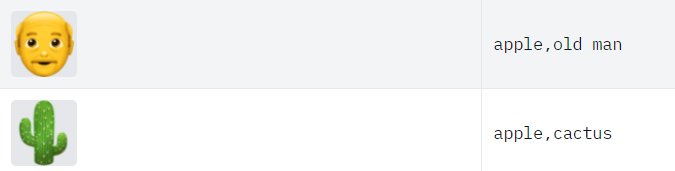
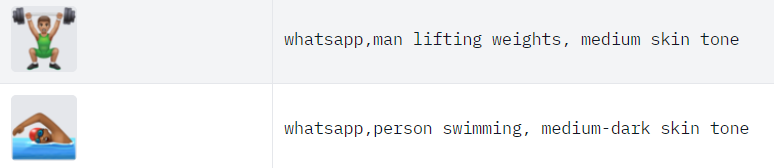
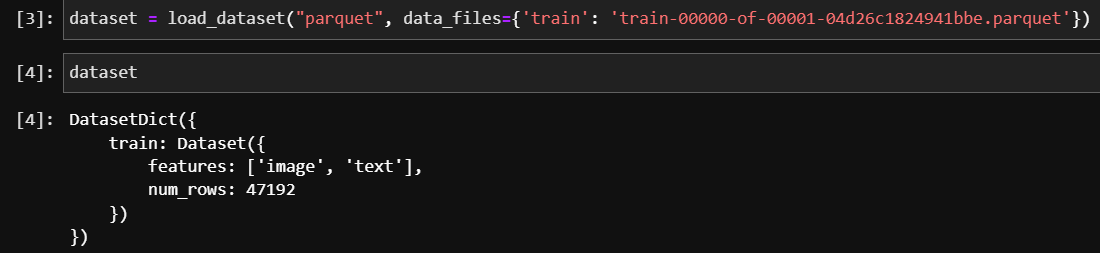


Figure *6* Dataset sample

4.3 Dataset Loading



4.3 Data Preparation

4.3.1 Feature Splitting

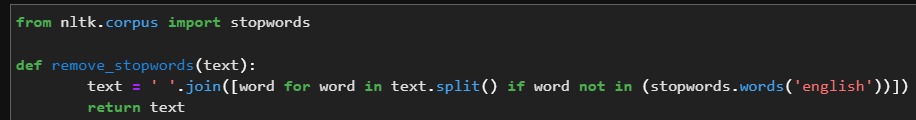
The text column is split into 3 new columns based on the following format

COMPANY, EMOJI CONTENT, DESCRIPTION

To text is split based on commas and trimmed to remove any spaces at the start or end.

4.3.2 Data Cleaning

The stop words are removed from emoji content and the description column to improve the accuracy of the model.

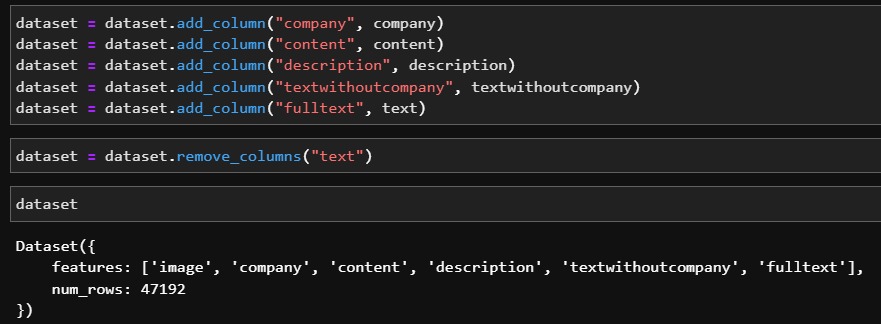


4.3.3 Elimination of Variable

After splitting, the previous text column is eliminated as the required data is captured in other columns.

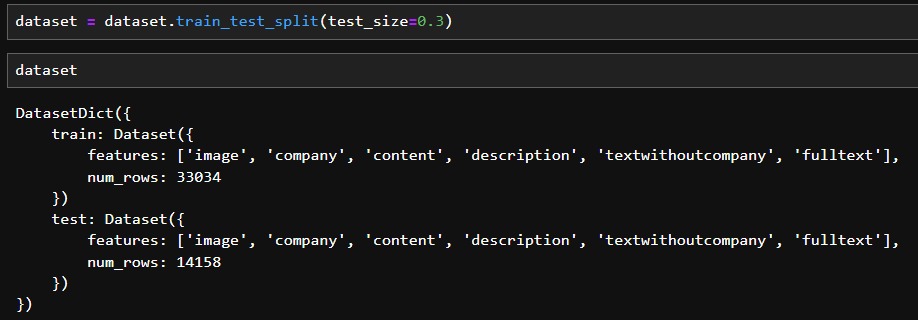
4.3.4 Transformation into New Feature

The emoji context and the description column are combined to create a new feature which will converted into embedding for model input.



4.3.5 Splitting of original dataset

The dataset is split into two sets train and test with train containing 70% and test containing 30%.

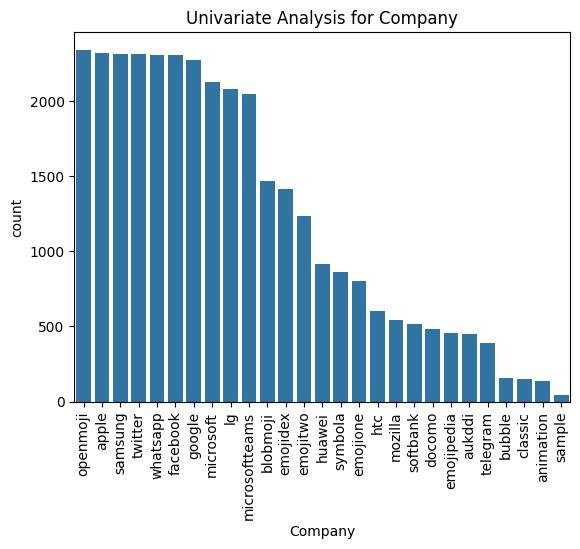


4.4 Exploratory Data Analysis

4.4.1 Univariate Analysis

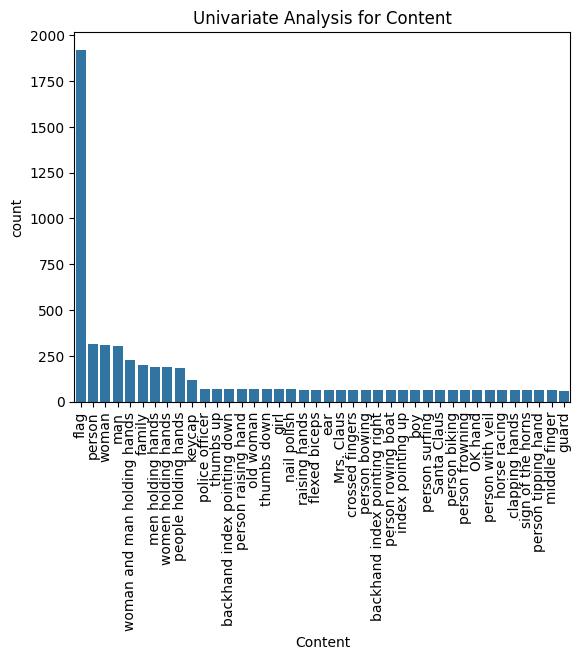
4.4.1.1 Company

The "openmoji" company is the most frequent, with a count significantly higher than any other company. Apple and Samsung follow closely behind openmoji, suggesting they also have a large number of emoji associated with them. Popular social media platforms like WhatsApp, Twitter, Facebook, and Google also have a considerable presence in the dataset. Companies like Microsoft, Microsoft Teams, and Telegram, which are primarily known for their technology products, also have a noticeable number of emoji associated with them.



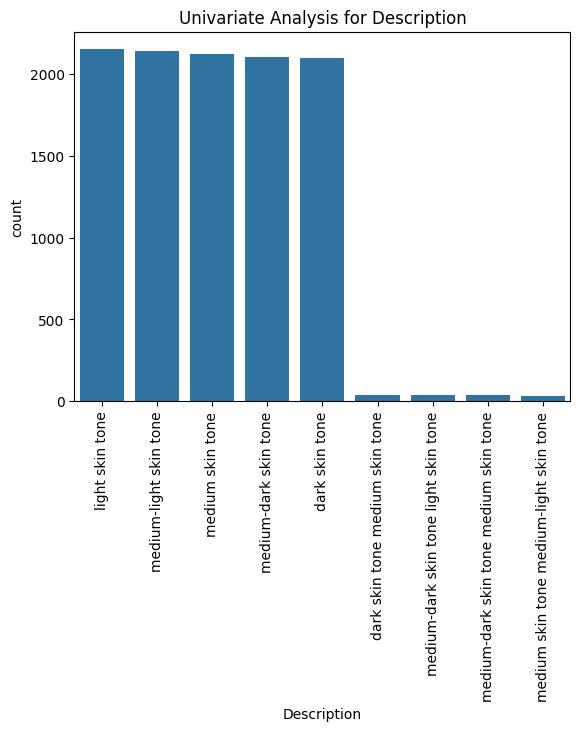
4.4.1.1 Content

The "flag" content is the most frequent, with a significantly higher count than any other content type. Content related to people, such as "person," "woman," "man," and various hand gestures, is also prevalent. Content depicting family and relationships, like "family," "woman and man holding hands," and "people holding hands," is represented.



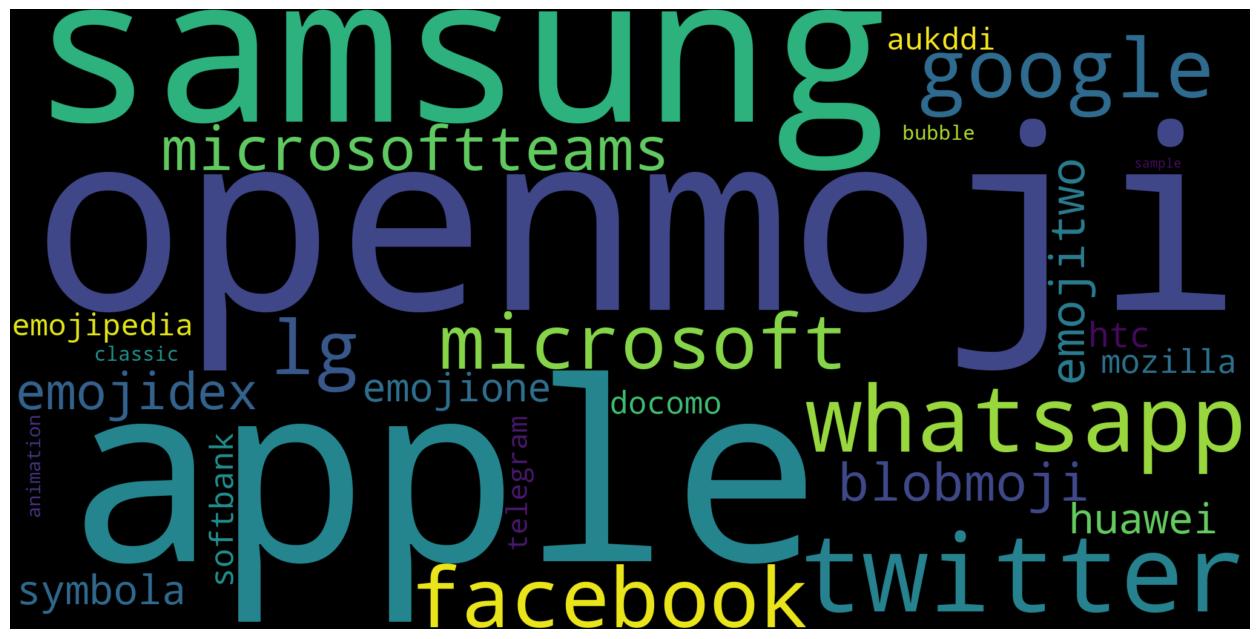
4.4.1.1 Description

The "light skin tone" category is the most frequent, with a significantly higher count than any other skin tone. The description column is dominated by the skin tone types with “light skin tone”, “medium-light skin tone”, “medium skin tone”, “medium-dark skin tone” and “dark skin tone”.

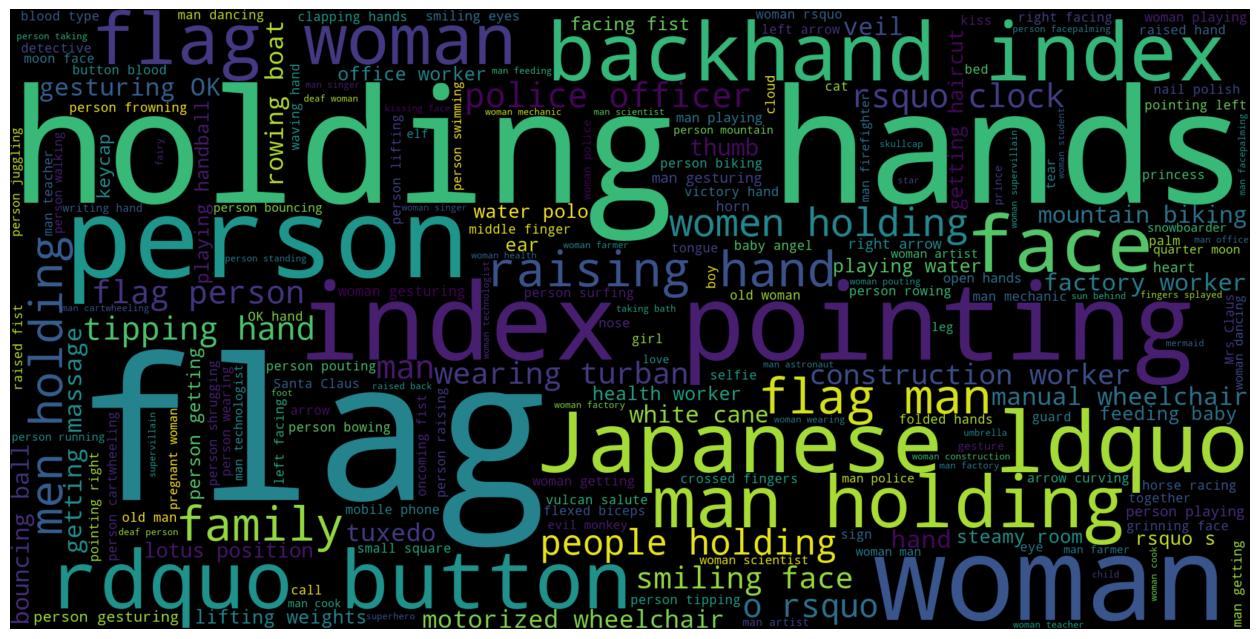


4.4.2 Word Cloud

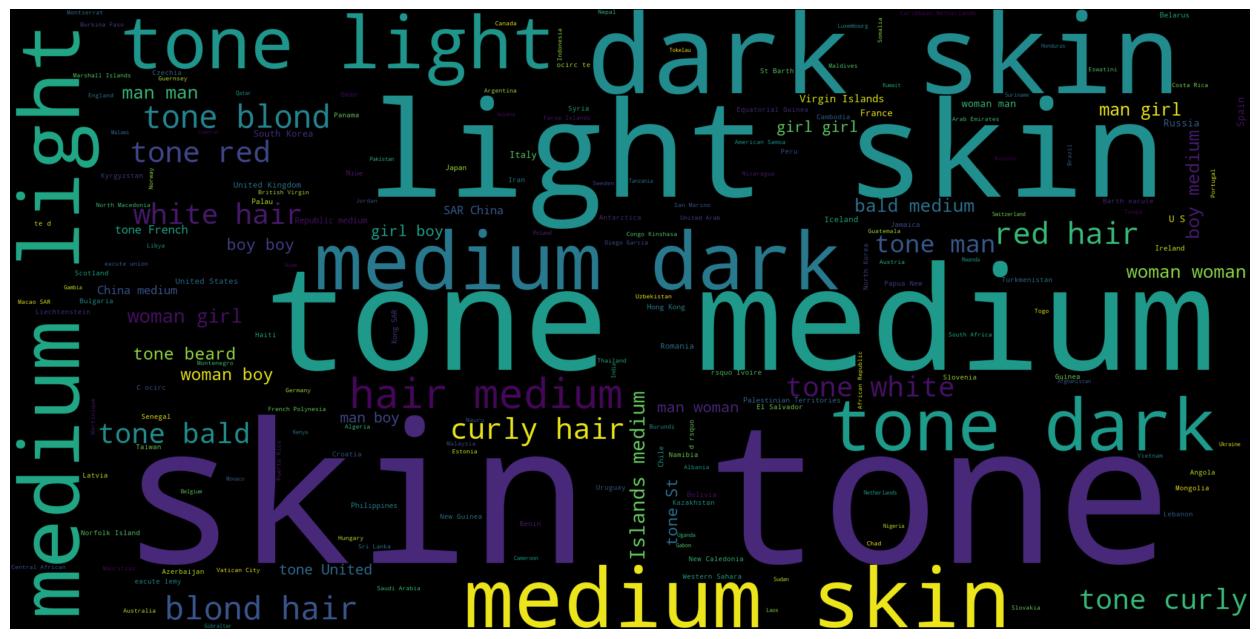
4.4.2.1 Company



4.4.2.2 Content



4.4.2.3 Description



4.5 Hyperparameter and the best parameters

**lr (Learning Rate):**

* It controls the step size during optimization.
* Smaller values (like 0.0005) often lead to more stable training but can be slower.
* Larger values (like 0.005) can accelerate training but may risk instability.

**noise\_dim (Noise Dimension):**

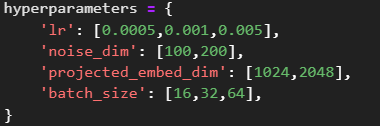
* It is the dimensionality of the random noise vector that is fed into the generator.
* A higher dimension can potentially generate more complex outputs.

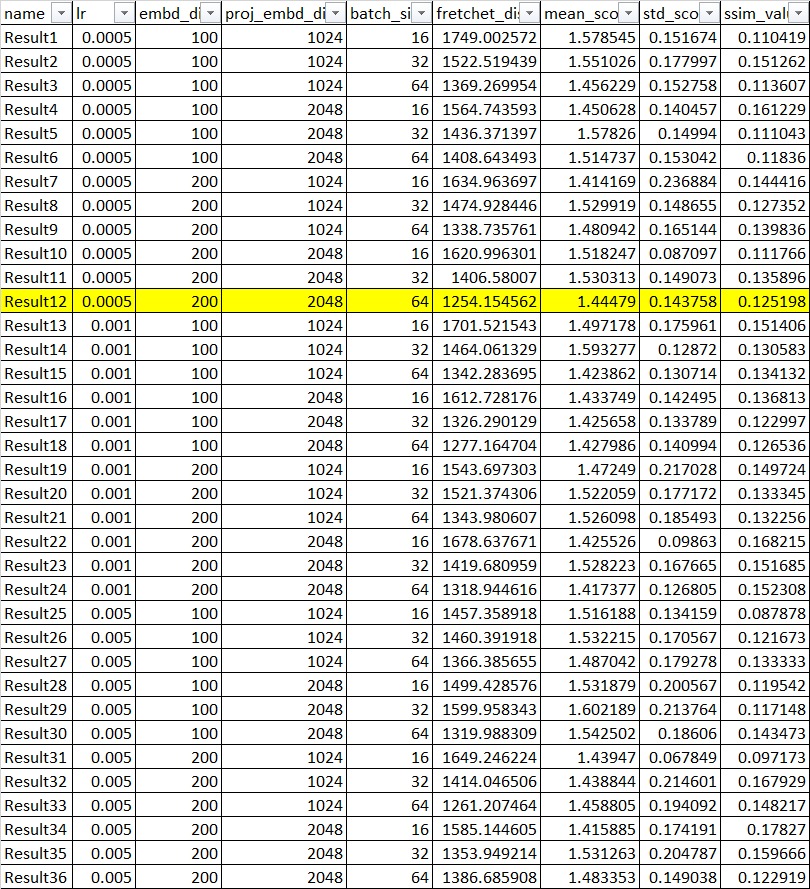
**projected\_embed\_dim (Projected Embedding Dimension):**

* This hyperparameter is related to a GAN architecture where the input text vector combined with the noise vector is projected to a higher dimension.

**batch\_size:**

* The number of samples processed in each training iteration.
* Larger batch sizes can improve training stability but require more memory.





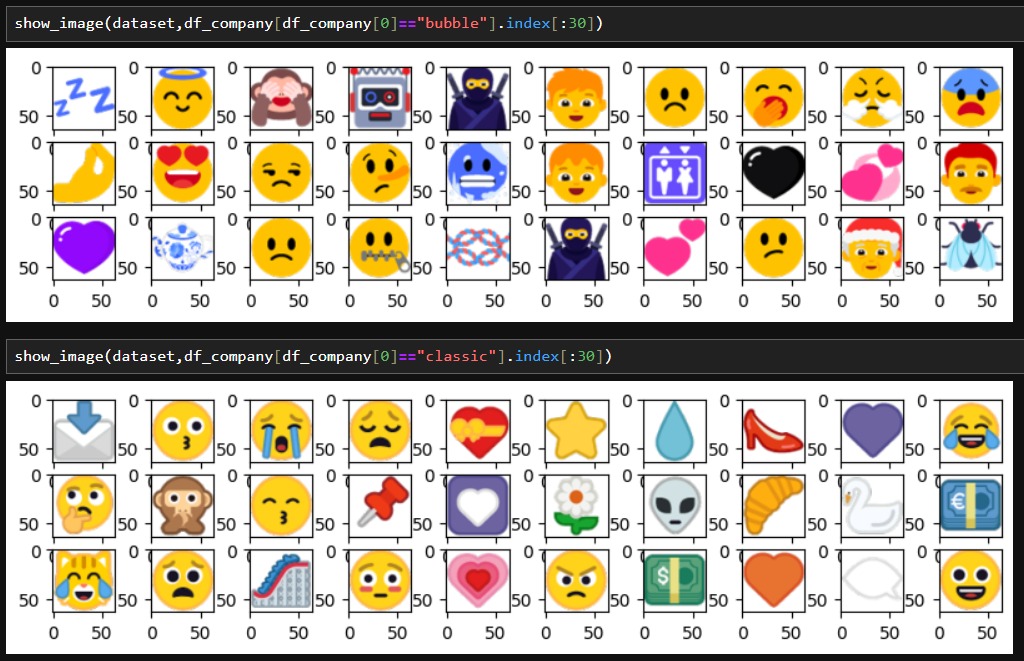
Key Observations from the above hyperparameter tuning:

* A learning rate of 0.0005 seems to consistently produce better results than higher learning rates, indicating that smaller steps might be more effective in this case.
* Increasing the embedding dimensions generally leads to improved results, suggesting that richer representations are beneficial.
* While there's some variation, larger batch sizes tend to perform slightly better than smaller ones, possibly due to increased stability during training.

Based on the analysis, the following hyperparameter settings might be considered as a starting point:

* **lr:** 0.0005
* **embd\_di:** 200
* **proj\_embd\_di:** 2048
* **batch\_size:** 64 or more (depending on computational resources)

4.6 Data Visualization



4.7 Summary

CHAPTER 5

RESULTS AND EVALUATION

5.1 Introduction

5.2 Interpretation of Visualizations

5.3 Evaluation of Sampling Methods and Results

5.4 Testing on Validation Dataset

5.6 Summary

CHAPTER 6

CONCLUSIONS AND RECOMMENDATIONS

6.1 Introduction

6.2 Discussion and Conclusion

6.3 Contribution to Knowledge

* GAN network performance on a large emoji dataset
* Comparison of different evaluation metrics on the emoji dataset
* Comparison of GAN with StackedGAN

6.4 Future Recommendations

REFERENCES

Asudani, D.S., Nagwani, N.K. and Singh, P., (2023) Impact of word embedding models on text analytics in deep learning environment: a review. *Artificial Intelligence Review*, [online] 569, pp.10345–10425. Available at: https://doi.org/10.1007/s10462-023-10419-1.

Cavalheiro, B.P., Rodrigues, D.L. and Prada, M., (2023) Who (and with whom) uses more emoji? Exploring individual, relational, and motivational characteristics driving emoji use. *Telematics and Informatics*, [online] 83, p.102023. Available at: https://www.sciencedirect.com/science/article/pii/S0736585323000874.

Crowson, K., Biderman, S., Kornis, D., Stander, D., Hallahan, E., Castricato, L. and Raff, E., (2022) Vqgan-clip: Open domain image generation and editing with natural language guidance. In: *European Conference on Computer Vision*. Springer, pp.88–105.

Dhivya, K. and Navas, N.S., (2020) Text to realistic image generation using stackgan. In: *2020 7th International Conference on Smart Structures and Systems (ICSSS)*. IEEE, pp.1–7.

Erle, T.M., Schmid, K., Goslar, S.H. and Martin, J.D., (2022) Emojis as social information in digital communication. *Emotion*, 227, p.1529.

Kennison, S.M., Fritz, K., Morales, M.A.H. and Chan-Tin, E., (2024) Emoji use in social media posts: relationships with personality traits and word usage. *Frontiers in Psychology*, 15.

Ku, H. and Lee, M., (2023) TextControlGAN: Text-to-Image Synthesis with Controllable Generative Adversarial Networks. *Applied Sciences*, 138, p.5098.

Lee, S., Kim, S., Chu, Y., Choi, J., Park, E. and Woo, S.S., (2023) EAE-GAN: Emotion-Aware Emoji Generative Adversarial Network for Computational Modeling Diverse and Fine-Grained Human Emotions. *IEEE Transactions on Computational Social Systems*, pp.1–11.

Mittal, P., Aggarwal, K., Sahu, P.P., Vatsalya, V., Mitra, S., Singh, V., Veera, V. and Venkatesan, S.M., (2020) Photo-realistic emoticon generation using multi-modal input. In: *Proceedings of the 25th International Conference on Intelligent User Interfaces*. pp.254–258.

Patashnik, O., Wu, Z., Shechtman, E., Cohen-Or, D. and Lischinski, D., (2021) Styleclip: Text-driven manipulation of stylegan imagery. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp.2085–2094.

Provine, R.R., Spencer, R.J. and Mandell, D.L., (2007) Emotional expression online: Emoticons punctuate website text messages. *Journal of language and social psychology*, 263, pp.299–307.

Sauer, A., Karras, T., Laine, S., Geiger, A. and Aila, T., (2023) Stylegan-t: Unlocking the power of gans for fast large-scale text-to-image synthesis. *arXiv preprint arXiv:2301.09515*.

Shmelkov, K., Schmid, C. and Alahari, K., (2018) How good is my GAN? In: *Proceedings of the European conference on computer vision (ECCV)*. pp.213–229.

Trevisan de Souza, V.L., Marques, B.A.D., Batagelo, H.C. and Gois, J.P., (2023) A review on Generative Adversarial Networks for image generation. *Computers & Graphics*, [online] 114, pp.13–25. Available at: https://www.sciencedirect.com/science/article/pii/S009784932300064X.

Urumutta Hewage, G.S., Liu, Y., Wang, Z. and Mao, H., (2021) Consumer responses toward symmetric versus asymmetric facial expression emojis. *Marketing Letters*, 32, pp.219–230.

Xu, T., Zhang, P., Huang, Q., Zhang, H., Gan, Z., Huang, X. and He, X., (2018) Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp.1316–1324.

APPENDIX A: RESEARCH PROPOSAL

emoji generation based on textual input using generative adversarial networks

DARSHIL AJAY PAREKH

Research Proposal

FEBRUARY 2024

Abstract

With the increase in popularity with its wide variety of uses, emojis have become so necessary and important that sending or responding to a message and commenting or reacting to a post on social media without emoji seems to be impossible. Emoji represents emotions, culture, age, gender, actions, gesture, expressions, sports, foods, places and objects.

Despite the importance the number of emojis are countable and finite in quantity. In this research we will be creating a generative model to generate new unique emojis using short text for more complex and distinction emojis

We will be using the state-of-the-art generative models which consist of mainly two-part generator and discriminator. The generator and discriminator will work one after the other. First the generator will be trained to generate new emojis and after couple of rounds the discriminator will be trained to detect is the emoji is fake or real. This is followed by generator and discriminator rounds. This process continues until the discriminator is not able to detect is the emoji is fake.

This will help the community to generate wide variety of emojis based on the requirement rather than choosing from a finite set of emojis.

This research will help to generate high quality and diverse emojis. It will also help the tech community with the comparison of the different GAN models with different evaluation metrics.

**TABLE OF CONTENTS**

[Abstract 2](#_Toc160349438)

[LIST OF FIGURES 4](#_Toc160349439)

[LIST OF TABLES 5](#_Toc160349440)

[LIST OF ABBREVIATIONS 6](#_Toc160349441)

[1. Background 7](#_Toc160349442)

[2.1 Problem Statement 9](#_Toc160349443)

[2.2 Related Work 10](#_Toc160349444)

[2.2.1 Research on Emoji and its Importance 10](#_Toc160349445)

[2.2.3 Research on Word Embeddings: 11](#_Toc160349446)

[3. Research Questions 13](#_Toc160349447)

[3.1 Research Questions 13](#_Toc160349448)

[4. Aim and Objectives 13](#_Toc160349449)

[5. Significance of the Study 14](#_Toc160349450)

[5.1 Contributions 14](#_Toc160349451)

[5.2 Beneficiaries 14](#_Toc160349452)

[6. Scope of the Study 14](#_Toc160349453)

[7. Research Methodology 15](#_Toc160349454)

[7.1 Introduction 15](#_Toc160349455)

[7.2 Dataset description 15](#_Toc160349456)

[7.3 Data preprocessing 16](#_Toc160349457)

[7.4 Transformation 16](#_Toc160349458)

[7.5 Models 16](#_Toc160349459)

[7.6 High Level Flow Diagram 17](#_Toc160349460)

[7.7 Evaluation metrics 17](#_Toc160349461)

[8. Requirements or Resources 18](#_Toc160349462)

[8.1 Hardware Requirement 18](#_Toc160349463)

[8.2 Software, Web Services and Libraries Requirement 18](#_Toc160349464)

[9. Research Plan 19](#_Toc160349465)

LIST OF FIGURES

[Figure 1 Sample emojis used frequently in informal communication. 7](#_Toc160313889)

[Figure 2 Emoji used in a feedback form 8](#_Toc160313890)

[Figure 3 Dataset sample 15](#_Toc160313891)

[Figure 4 High Level Flow Diagram 17](#_Toc160313892)

[Figure 5 Research Plan 19](#_Toc160313893)

LIST OF TABLES

[Table 1 Related works on Methodologies 12](#_Toc160313894)

[Table 2 State-of-the-art model to be used for emoji generation 17](#_Toc160313895)

LIST OF ABBREVIATIONS

GAN Generative Adversarial Network

VQGAN Vector Quantized Generative Adversarial Network

DCGAN Deep Convolution Generative Adversarial Network

CLIP Contrastive Language-Image Pre-Training

AI Artificial Intelligence

ML Machine Learning

RGB Red Green Blue

RGBA Red Green Blue Alpha

1. Background

Emoji is a visual representation of human interaction based on Human emotions, living beings, objects and symbols. These are in widespread use across the internet such as in textual messaging application, social media platforms which are known as informal modes of communication. Below is an image about the most frequently used emojis during informal modes of communication.

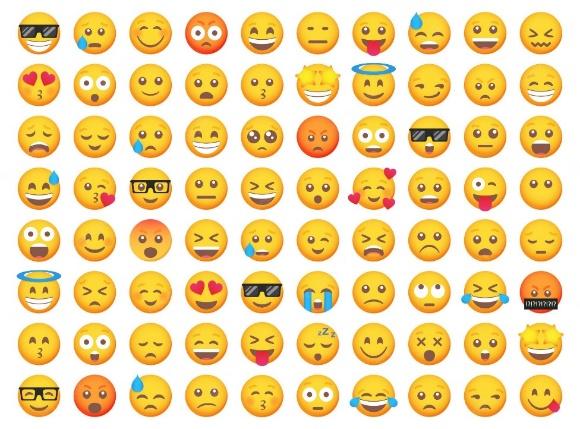


Figure *8* Sample emojis used frequently in informal communication.

According to (Kennison et al., 2024) emojis represent faces with human expressions, objects, animals, and actions.

Emojis are often used to in replacement of words, thus acting as a part of communication and language (Provine et al., 2007). As per the research (Urumutta Hewage et al., 2021) the emojis are static images which are often preserved as human like emotion and expressions. Further, as per (Erle et al., 2022) the communication has greater emotional intensity when we communicated to people using emojis rather than without emojis.

Other application of emojis other than informal communication includes

* Rating/Feedback Form : The feedback forms using emoji instead of stars or numbers. The research has found that the chances of the user giving feedback is higher when emojis are used.
* Providing a Reaction : These days most online applications or websites provide emojis to react to a post, blog, article or tweet.
  + Thumbs up and down emojis are used to like and dislike.
  + Facial emojis are used to express feeling like surprise, wow or shocked.

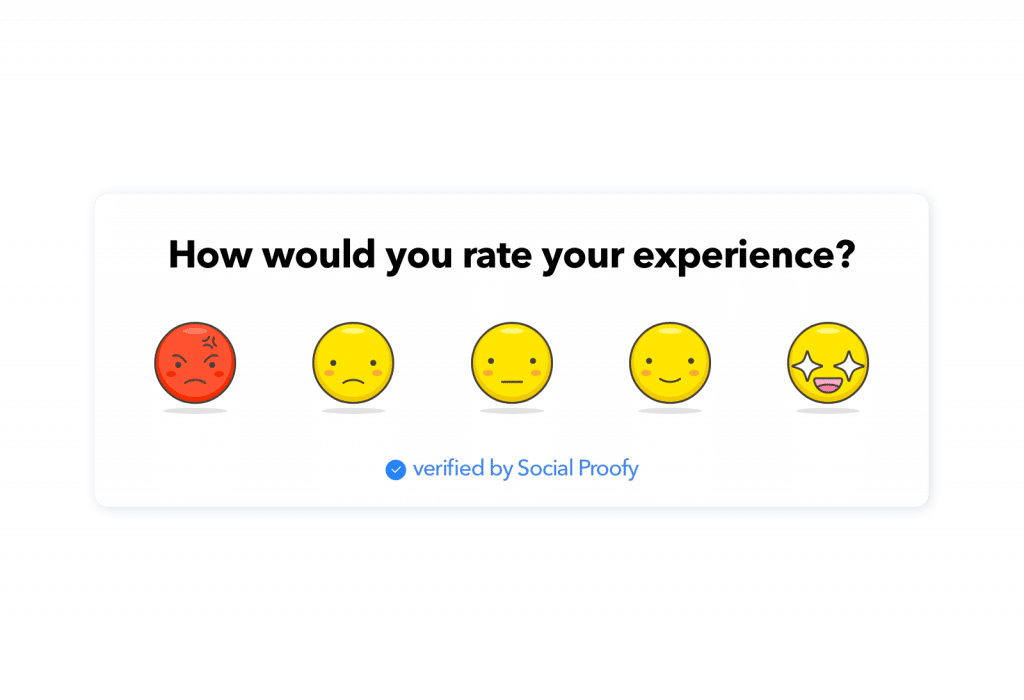


Figure *9* Emoji used in a feedback form

As per the emojipedia.org analysis of over 2.934 billion global tweets between 2017 to 2021, the percentage of tweets with at least one emoji has increased from 13.5% to 21.5%. With the increase of the use of emoji, the demand for newer and customised emoji is increasing. The creation of emoji is a currently is a tedious process of submitting an emoji to Unicode Technical Committee, which takes the final decision for the inclusion, this takes 18-24 months for the complete process Using the state-of-the-art models for generating images using text, we will be able to create new emojis based on the user input.

Prior research has focused on generating facial emojis and emotion based facial emojis. With the state-of-the-art model and architecture, we will be generating not only facial emojis but objects also.

Prior research used techniques like unsupervised learning , Deep Convolution based Generative Adversarial Networks and Conditional Generative Adversarial Networks.

One of the greatest debates on emojis includes whether emoji can be used “as punctuation” or ”after punctuation” which was included in “The Great Emoji Debate”.

In this research we will be creating a state-of-the-art model to generate emojis using emoji content and descriptive information of the emoji and comparing the model with prior models.

We will be looking at the various generative adversarial models :

* VQGAN-CLIP
* TextControl-GAN
* StyleGAN-T

**2. Problem Statement and Related Work**

2.1 Problem Statement

Currently the process of creating new emojis is a tedious and time-consuming task which takes a couple of days or months to create and years to get approval from the Unicode Technical Committee. Currently there are only 3,664 emojis as per Unicode standard and the need for need and diverse emoji is increasing. Due to this the current library faces limitations in capturing all the human emotion and experiences. This can lead to miscommunication and inadequate representation of culture and identities.

As per the current trend the use of emoji has been steadily increasing in informal conversation and social media. The need for new emojis to be generated quickly based on the text from the user is high. This issue mostly affects the Gen Z generation as they are the ones who use them mostly during the social media platforms, Messaging apps and content creation. It is important as it can improve the quality of the conversation with improved emotions and objects. This solution will have a huge impact on the conversation by improving the quality of the conversation with new and better emojis which can express emotions and objects accurately.

2.2 Related Work

2.2.1 Research on Emoji and its Importance

Emoji despite being more common in digital communication, is often dependent on characteristics, motives for using them and the context in which they are used (Cavalheiro et al., 2023). Over the past couple of years there has been extensive research in the use and importance of emojis in every life activity. Below are few of the recent research with description and findings:

* The research by (Cavalheiro et al., 2023) try to find the relationship of frequency of the emoji with relationship and individual characteristics. The research also found personal reasons to be one of the relates to high emoji frequency. The research also found age to be directly correlated with emoji frequency which was unexpected.
* The research by (Erle et al., 2022) discusses a creation of emojis in digital communication with the help of their model. The research also found that the emojis can depict the facial expression of the human.
* The research by (Lefebvre et al., 2024) is about the influence of emoji during tip suggestion with the tipping percentage. The research also found significant positive effects of emoji during the tipping.
* The research by (Kennison et al., 2024) relates the personal characteristics with the emojis used in the conversation. The research also found that the men used emojis less often than women. The research also found that users may me using emojis in place of word and used less dictionary words.

**2.2.2 Research on Emoji Generation**

With the evolution generative models and the importance of emoji there has been couple of research done in the field of emoji generation. It has been noticed that most of the research has been around facial emojis and with a very limited set of them. Below are few of the recent research with description:

* The research (Mittal et al., 2020) uses supervised learning with multimodal input to GAN with U-Net like architecture for the generator to generate realistic hand drawn emojis. This research even though was able to generate the emojis it required the user to draw a rough sketch with the emotion to be given as input, also the dataset used had to be augmented from the facial emojis which don’t depict the real dataset.
* The research (Lee et al., 2023) creates a model which can generate emojis based on emotional degree for more complex and detailed usage on online conversations (Lee et al., 2023). The research uses conditional GAN for the experiment and also compares it with other baseline models. This
* The research (Tang, 2023) used unsupervised learning to generate new emojis using DCGAN which is a combination of Convolution Neural Network and Generative Adversarial Network. This research faced many challenges like after 1000 epochs the accuracy flattens out. The emojis generated where quite blurry and the user had no control over the emojis generated due to unsupervised learning.
* The research (Xu, 2021) used unsupervised learning with optimization techniques with Adam Optimizer to generate new emojis.
* The research (Peng and Zhao, 2021) used an encoder–decoder model, to predict the sequence of emojis based on short text. This research was able to correlate the emojis generated. The model was also able to learn the semantics between the emoji and text description.

Although the above research where able to generate emojis but most of this research had sparse data and limited the scope to generate facial emojis.

2.2.3 Research on Word Embeddings:

According to the research by (Asudani et al., 2023) the first preference should be given to domain specific embeddings. The research also compares the performance of the various word embedding techniques.

**2.2.4 Research on Text to Image Generation Methodology**

Below are the recent methodologies in the text to image domain developed on top of generative adversarial network architecture.

|  |  |  |
| --- | --- | --- |
| **Methodology** | **Use** | **Description** |
| **Attn-GAN** (Xu et al., 2018) | Text to Image Generation | The research proposes a new GAN with multistage refinement using attention. Each of this multistage attention model gets the most relevant word vectors for generating sub-regions of the image (Xu et al., 2018). |
| **Stack-GAN** (Dhivya and Navas, 2020) | Image Generation based on text | The research proposes the use of multiple GAN stacked on top of each other to generate high resolution photo realistic images (Trevisan de Souza et al., 2023). |
| **Style-CLIP**(Patashnik et al., 2021) | Text-guided image editing | This research proposes CLIP embeddings to allow image editing. The CLIP embeddings share the same latent space hence we can directly compare the text with the images (Trevisan de Souza et al., 2023). |
| **VQGAN-CLIP**(Crowson et al., 2022) | Text-to-image synthesis | This research proposes a model where an encoder is used to create a latent vector which is used by the decoder to create the image. It also uses CLIP embedding with optimization techniques to have high similarity with the text prompt. (Trevisan de Souza et al., 2023). |
| **TextControl-GAN** (Ku and Lee, 2023) | Text to Image | The research proposes a neural network as regressor to learn the textual features. The research also uses data augmentation techniques to increase the performance(Ku and Lee, 2023). |
| **StyleGAN-T**(Sauer et al., 2023) | Text-to-image synthesis | This research proposes a model build from Style GAN-XL mentioned before. The network size of the model is increased to include text information from CLIP encoders.  (Trevisan de Souza et al., 2023). |

Table 1 Related works on Methodologies

3. Research Questions

3.1 Research Questions

Generate new realistic emoji in real time based on the textual input from the user.

4. Aim and Objectives

The aim of this research is to propose a generative model to generate new emojis is to create a new and diverse design based on textual input from the user and compare it with the baseline and previous models which are visually appealing and useful.

Below are the list of research objectives which will be accomplished in this research:

* To create a generative model that can create new high quality and realistic emojis.
* To generate emoji based on the platform of use.
* To compare between the state-of-the-art generative models.
* To evaluate the performance of the emojis generated using the generative models.
* To finetune the various parameters used in the generative model.

5. Significance of the Study

5.1 Contributions

* Offer users a wider range of options to accurately express their emotions and experiences using emoji,
* Reflects the variety of individuals and communities.
* Contribute to the development and application of GANs
* Valuable tool for individuals and communities to create custom emojis

5.2 Beneficiaries

* Users of social media platforms.
* Content creators like article and blog writers.
* The AI and ML industry.
* Emoji Designers.
* Unicode and emoji organisations like emojipedia.

6. Scope of the Study

The scope of a study on generating new emojis are as follows:

* Type of architecture : Generative Adversarial Networks
* Data : <https://huggingface.co/datasets/ChengAoShen/emoji_with_text>
* Input : Short textual input
* Output : 64x64 emoji
* Evaluation techniques :
  + Inception Score (Shmelkov et al., 2018)
  + Fréchet Inception Distance (Shmelkov et al., 2018) (Lee et al., 2023)
  + Learned Perceptual Image Patch Similarity (Lee et al., 2023)
  + Structural Similarity Index (Mittal et al., 2020)

7. Research Methodology

In this section we will be going through the end-to-end architecture which included data preprocessing, data transformation, the state-of-the-art methodology and the evaluation metrics used in the research.

7.1 Introduction

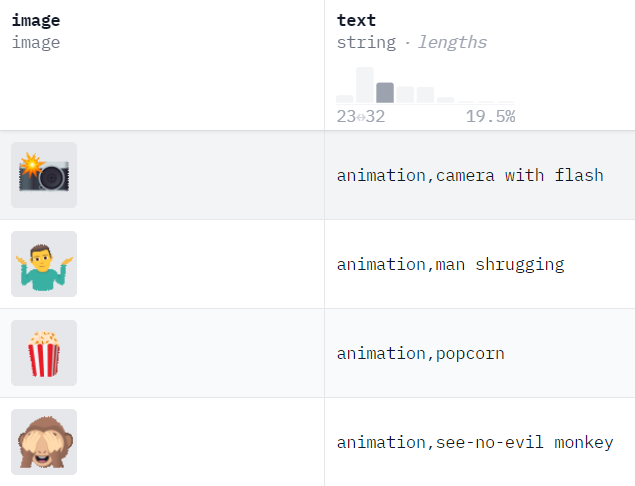
Here's a simplified explanation of GANs:

* The Forger (Generator): A computer program that starts by making very bad drawings   
  of the emoji
* The Detective (Discriminator): Another computer program that gets shown both real emoji and the Forger's fakes. The Detective tries to tell them apart.

Each time the Detective catches a fake, the Forger learns from its mistakes and tries to make even better fakes. This makes the Detective have to get smarter too.

7.2 Dataset description

The dataset (ChengAo Shen and Siyuan Mu, 2023) is uploaded on Hugging face. This dataset consists of 47200 emojis with the text description. Each image is of shape 64\*64 and has RGBA channels. Description text format : app or company, emoji content, description information



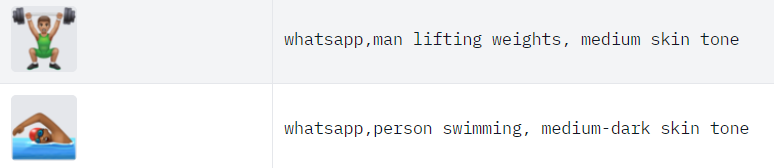
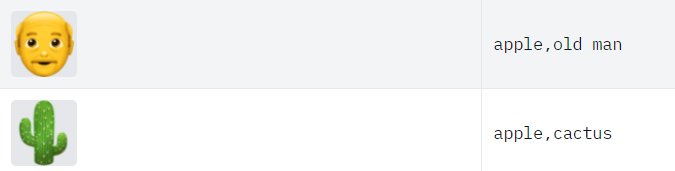
 

Figure *10* Dataset sample

7.3 Data preprocessing

The dataset doesn't require any cleaning or filtering as it has already been done by the author of the dataset. All the emoji are of the shape 64x64x4 where 4 is the number of channels which represent RGBA (Red, Green, Blue, Alpha) which can be converted to RGB(Red, Green, Blue) format. The text description will be split into 3 columns: app/company, emoji content, description information using regular expressions. These 3 columns will be processed to remove stop word and name entity recognition will be used to identify the company and keywords from the description. The dataset will be organised and divided into 3 parts for training, testing and validation.

7.4 Transformation

* Latent Space Manipulation  
  Latent space is the hidden layers in the architecture which can be seen as compressed input. As proposed in (Yang et al., 2021) we can use latent space factorization or manipulations to change the attributes of the image.
* Word to Vector  
  The text description for the emojis will be converted to vector representation.  
  They capture the meaning and their usage.
* Clip embedding

The images and text vectors share the same latent space which is helpful in direct comparison of the both

7.5 Models

Below are the recent shortlisted state of the art model which we will be using and comparing to generate the emojis using the textual input.

|  |  |
| --- | --- |
| **Methodology** | **Description** |
| **VQGAN-CLIP** (Crowson et al., 2022) | This research proposes a model where an encoder is used to create a latent vector which is used by the decoder to create the image. It also uses CLIP embedding with optimization techniques to have high similarity with the text prompt. (Trevisan de Souza et al., 2023). |
| **TextControl-GAN** (Ku and Lee, 2023) | The research proposes a neural network as regressor to learn the textual features. The research also uses data augmentation techniques to increase the performance(Ku and Lee, 2023). |
| **StyleGAN-T** (Sauer et al., 2023) | This research proposes a model build from Style GAN-XL mentioned before. The network size of the model is increased to include text information from CLIP encoders.  (Trevisan de Souza et al., 2023). |

Table 2 State-of-the-art model to be used for emoji generation

7.6 High Level Flow Diagram

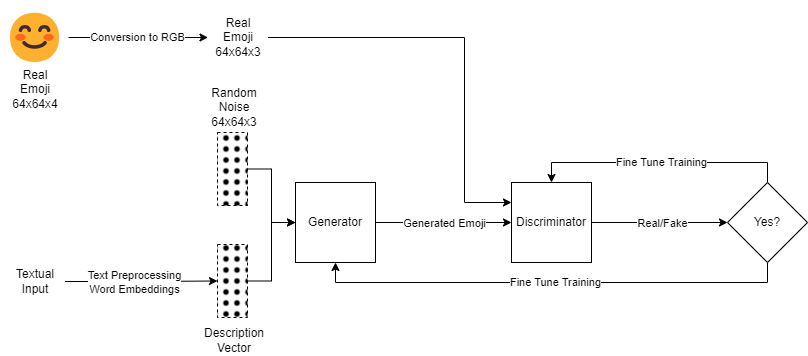


Figure *11* High Level Flow Diagram

7.7 Evaluation metrics

We can measure the image quality the following metrics are used

* Inception Score (Shmelkov et al., 2018)   
  It is used to measure the quality of the images generated by comparing it with the real images from the dataset.
* Fréchet Inception Distance (Shmelkov et al., 2018) (Lee et al., 2023)

It is the measure to check how real the image generated are.

* Learned Perceptual Image Patch Similarity (Lee et al., 2023)

It is the distance measured between the patches in the latent space.

* Structural Similarity Index (Mittal et al., 2020)

It measures the degradation of the image quality

8. Requirements or Resources

Initially the data preprocessing, analysis, model creation and partial training will be done on local laptops and for the final training of the complete dataset and validation will be done on cloud VMs

8.1 Hardware Requirement

* Intel i7 processor
* 16gb RAM
* 4gb Video RAM/GPU
* 1 cloud VM with 1 Nvidia A100 GPU

8.2 Software, Web Services and Libraries Requirement

Below are the list of software requirements

* Python 3.12.2
* JupyterLab
* Microsoft Azure/GCP
* PyTorch
* Numpy
* Pandas
* Scikit-learn
* Tensorflow
* Keras
* Matplotlib

9. Research Plan



Figure *12* Research Plan

References

1. Asudani, D.S., Nagwani, N.K. and Singh, P., (2023) Impact of word embedding models on text analytics in deep learning environment: a review. *Artificial Intelligence Review*, [online] 569, pp.10345–10425. Available at: https://doi.org/10.1007/s10462-023-10419-1.
2. Cavalheiro, B.P., Rodrigues, D.L. and Prada, M., (2023) Who (and with whom) uses more emoji? Exploring individual, relational, and motivational characteristics driving emoji use. *Telematics and Informatics*, [online] 83, p.102023. Available at: https://www.sciencedirect.com/science/article/pii/S0736585323000874.
3. Crowson, K., Biderman, S., Kornis, D., Stander, D., Hallahan, E., Castricato, L. and Raff, E., (2022) Vqgan-clip: Open domain image generation and editing with natural language guidance. In: *European Conference on Computer Vision*. Springer, pp.88–105.
4. Dhivya, K. and Navas, N.S., (2020) Text to realistic image generation using stackgan. In: *2020 7th International Conference on Smart Structures and Systems (ICSSS)*. IEEE, pp.1–7.
5. Erle, T.M., Schmid, K., Goslar, S.H. and Martin, J.D., (2022) Emojis as social information in digital communication. *Emotion*, 227, p.1529.
6. Kennison, S.M., Fritz, K., Morales, M.A.H. and Chan-Tin, E., (2024) Emoji use in social media posts: relationships with personality traits and word usage. *Frontiers in Psychology*, 15.
7. Ku, H. and Lee, M., (2023) TextControlGAN: Text-to-Image Synthesis with Controllable Generative Adversarial Networks. *Applied Sciences*, 138, p.5098.
8. Lee, S., Kim, S., Chu, Y., Choi, J., Park, E. and Woo, S.S., (2023) EAE-GAN: Emotion-Aware Emoji Generative Adversarial Network for Computational Modeling Diverse and Fine-Grained Human Emotions. *IEEE Transactions on Computational Social Systems*, pp.1–11.
9. Mittal, P., Aggarwal, K., Sahu, P.P., Vatsalya, V., Mitra, S., Singh, V., Veera, V. and Venkatesan, S.M., (2020) Photo-realistic emoticon generation using multi-modal input. In: *Proceedings of the 25th International Conference on Intelligent User Interfaces*. pp.254–258.
10. Patashnik, O., Wu, Z., Shechtman, E., Cohen-Or, D. and Lischinski, D., (2021) Styleclip: Text-driven manipulation of stylegan imagery. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp.2085–2094.
11. Provine, R.R., Spencer, R.J. and Mandell, D.L., (2007) Emotional expression online: Emoticons punctuate website text messages. *Journal of language and social psychology*, 263, pp.299–307.
12. Sauer, A., Karras, T., Laine, S., Geiger, A. and Aila, T., (2023) Stylegan-t: Unlocking the power of gans for fast large-scale text-to-image synthesis. *arXiv preprint arXiv:2301.09515*.
13. Shmelkov, K., Schmid, C. and Alahari, K., (2018) How good is my GAN? In: *Proceedings of the European conference on computer vision (ECCV)*. pp.213–229.
14. Trevisan de Souza, V.L., Marques, B.A.D., Batagelo, H.C. and Gois, J.P., (2023) A review on Generative Adversarial Networks for image generation. *Computers & Graphics*, [online] 114, pp.13–25. Available at: https://www.sciencedirect.com/science/article/pii/S009784932300064X.
15. Urumutta Hewage, G.S., Liu, Y., Wang, Z. and Mao, H., (2021) Consumer responses toward symmetric versus asymmetric facial expression emojis. *Marketing Letters*, 32, pp.219–230.
16. Xu, T., Zhang, P., Huang, Q., Zhang, H., Gan, Z., Huang, X. and He, X., (2018) Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp.1316–1324.