scientific reports

Check for updates

OPEN

Deep neural architecture for natural language image synthesis for Tamil text using BASEGAN and hybrid super resolution GAN (HSRGAN)

M. Diviya[1] & A. Karmel[2]✉

Tamil is a language that has the most extended history and is a conventional language of India. It has antique origins and a distinct tradition. A study reveals that at the beginning of the twenty-first century, more than 66 million people spoke Tamil. In the present time, image synthesis from text emerged as a promising advancement in computer vision applications. The research work done so far in intelligent systems is trained in universal language but still has not achieved the desired development level in regional languages. Regional languages have a greater scope for developing applications and will enhance more research areas to be explored, ruling out the barrier. The current work using Auto Encoders failed at the point of providing vivid information along with essential descriptions of the synthesised images. The work aims to generate embedding vectors using a language model headed by image synthesis using GAN (Generative Adversarial Network) architecture. The proposed method is divided into two stages: designing a language model TBERTBASECASE model for generating embedding vectors. Synthesising images using Generative Adversarial Network called BASEGAN, the resolution has been improved through two-stage architecture named HYBRID SUPER RESOLUTION GAN. The work uses Oxford-102 and CUB-200 datasets. The framework efficiency has been measured using F1 Score, Fréchet inception distance (FID), and Inception Score (IS). Language and image synthesis architecture proposed can bridge the gap between the research ideas in regional languages.

Image synthesis from natural language descriptions is a field of research focusing on generating visual content, such as images or illustrations, based on textual descriptions or prompts. It involves training machine learning models, typically deep learning techniques, to understand the connection between text and corresponding visual output. Text-driven image synthesis aims to develop a system that can generate meaningful and accurate images based on text features. Various applications include creative Design generation of visual content for various design purposes, such as illustrations for books, magazines, or websites.

Virtual Worlds and Gaming: It can automatically generate visual assets, such as landscapes, characters, or objects, in virtual worlds or video games based on textual descriptions or procedural generation. Data Augmentation for creating synthetic images for training machine learning models in computer vision tasks, helping to expand the size and diversity of available training data. Storytelling and Visualization: It can aid in the generation of visual representations for storytelling, helping to illustrate scenes or concepts described in written narratives.

Text-to-image synthesis typically involves training a variational autoencoder (VAE) or a generative adversarial network (GAN) on a large dataset of paired text and corresponding images. The model understands to map the text to a visual feature by capturing the underlying patterns and relationships in the training data. While significant progress has prompted Visual content generation from text, highly detailed and realistic images that accurately capture the nuances of textual descriptions remain challenging. However, recent advancements in deep learning techniques, including larger models and improved training methodologies, have shown promising results in generating visually coherent and contextually relevant images from text prompts.

[1]School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, Tamilnadu 600127, India. [2]School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, Tamilnadu 600127, India. ✉email: karmel.a@vit.ac.in

A unique text-based image stylization system. Several steps are unsupervised. Style images are initially divided into foreground and background images. Then, the primary foreground colour is gathered, and binarized spatial shapes like text, symbols, and icons are assigned to create content images. Picture style transfer applies the foreground picture's style to the content image[1]. Researchers proposed a technique that stylizes shapes such as text, symbolic representations, and patterns in binary images using an input-style image. Then the styled geometric shape and backdrop image are combined. Legibility-preserving structure and texture transfer methods have been employed to reduce discrepancies between both images[2]. Followed with other architectures, the proposed model may draw the necessary outcomes for creating higher-quality image synthesis by synthesising from simple to complicated and easy to difficult[3].

Generative Adversarial Networks (GANs), proposed by Ian Good Fellow, is an eminent idea that plays various roles in every computer vision application. It is an unsupervised model that acts as a generative model in synthesizing novel images from similar images, generating new tunes from the existing ones[4]. The model proposed by the researchers works better for natural scenes and faces. The work aims to bring out the robustness of GAN apart from its performance[5]. Yanhua Li et al. worked in synthesizing dressing images from the given input images, which is termed fitting GAN. The model learns the input images, and the output generated according to the fitting. The model used L1 regularization and adversarial loss function[6]. Jezia et al. created a model that helps synthesis realistic images from a story text using spatial relations of the words. Followed by using two-stage models of generator and discriminator, the low-resolution images are converted to high-resolution images, and a KNN is used to compare the image quality score[7]. Valuable progress has been made in computer vision in image synthesis applications. The researchers concentrated on preserving the semantic information of the images. Super-resolution GAN is used for enhancing the resolution and quality of low-resolution images. It is advantageous to upscale an image without losing too much detail or introducing artifacts. The basic idea behind a super-resolution GAN is to train a generator network to produce improved-resolution images from low-resolution counterparts.

BERT (Bidirectional Encoder Representations from Transformers) is a well-trained language model. Transformer architecture is the basic block and has overtaken the other natural language processing tools. It is trained using massive corpus of text data and can understand the context and meaning of words and sentences. The critical feature of BERT is its bidirectional training, considering contexts at both ends of a word during training. This allows us to capture deeper semantic understanding and handle tasks that require understanding the closeness across different sentence parts. The model has been trained on large amounts of text from means like books, articles, and websites, which enables it to learn a broad range of language patterns and nuances.

After pre-training, BERT can be further trained on task-specific datasets with minimal adjustments to the model architecture. This fine-tuning process allows us to adapt to specific tasks and improve performance. It has paved the way for advancements in NLP. It has inspired the development of several other transformer-based models, such as GPT-3, RoBERTa, and ALBERT. These have achieved remarkable results in a comprehensive understanding of language and generation tasks.

The hands of language models have acquired various fields of applications. The researchers worked on BERT model for news classification with enormous data combined with spark. The performance of the BERT proves to be outstanding even when we have extensive data under study[8]. In NMT, BERT works better as contextual Embedding than fine-tuning for downstream language understanding tasks. This inspires us to improve BERT's NMT use. The BERT-fused model first uses BERT to uproot depictions for an input sequence, then uses attention mechanisms to fuse the representations with every layer of encoder and decoder of the NMT model[9,10].

Related works

Many applications include various language models. The researchers proposed a language model using relational features where the text is represented as feature types including the n-grams of words, Character n-grams, etc., The method analyses the distance between the available tokens across the document. It concentrates on binary features, whereas the model does not consider longer-range relation features. The second approach followed was by using keyword detection and vectorization of the corresponding keywords words[11]. Various field works have been attempted in various applications for languages. However, the researchers developed a pre-trained model in extractive summarization for the biomedical domain, which is a challenging approach. They trained the sentence using the BIOBERTSUM language model, which follows the sentence position embedding mechanism[12]. While working with the embedding mechanism, the model taken in hand by the researchers is a unified model that can identify multiple paraphrases in an input sentence pair. The preprocessing method generates sentence embedding vectors using the sentence-BERT model[13]. Many researchers proposed multiple streams of methodologies in processing text input to arrive at better information for further processing. The survey done by the authors in lining up various word representation models starting from raw text into meaningful vectors includes preprocessing by tokenization, noise removal, segmenting words followed by stemming, and lemmatization ends up with pos tagging[14]. Even though applications may vary, the primary task that involves text input records needs an understanding of how statistical properties of the sentence or word are analyzed. They have incorporated methodologies such as ensemble models instead of Continuous Bags of Words, Skip-Gram, etc.[15].

The publication of BERT and, more subsequently, GTP-3 represented a significant advancement for NLP. They provided an overview of the most significant language representation learning models developed for NLP and discussed the development of these models over time. In addition, it provides a summary, a comparison, and a contrast of these various models on sentiment analysis, then proceeds to evaluate their primary benefits and drawbacks[16]. Researchers proposed two parameter-reduction methods to reduce BERT memory usage and training speed. The methods scale better than BERT. A self-supervised loss that models inter-sentence coherence

reliably improves preliminary tasks with multi-sentence inputs. Top model achieves state-of-the-art scores on the GLUE, RACE, and \squad benchmarks with less parameters than BERT-large[17]

BERT is an all-purpose language representation model that allows computers to use the rich, two-way context contained in natural language texts. The sequence-transduction model transformer is used for the attention mechanism. In the approach, classification is carried out in parallel for each word in the input word sequence to determine whether each word can function as an antecedent. The machine-readable text requires a language model. A linguistic model can predict the likelihood of a context-related term. Most language models in research are based on unidirectional training, which seems daunting. BERT is bidirectional, unlike Elmo. Transformer encoder-based word Embedding is used[18]. A fine-tune and classify patents with the model. The technique outperforms CNN with word embeddings on over two million patent datasets. They concentrated on patent claims alone. Indicating that patent claims alone can yield state-of-the-art classification results, contrary to popular belief[19].

They fine-tun the pre-trained BERT with much text, yielding rich text information for image production. Position Embeddings (PEs) have been proposed to reflect word order in Transformer-based systems like BERT. No formal framework exists to investigate these empirically-driven, high-performing models. Translation invariance, monotonicity, and symmetry of PEs capture word distance in vector space. These features formalise PE behavior and enable principled sinusoidal PE reinterpretation. The proposed work offers a new probing test, "identical word probing," and indicators using mathematical functions to objectively detect general attention patterns related to the qualities described above[20].

Multi-perspective fusion was introduced to improve image synthesis. The generator uses a dynamic selection approach to match text and image features. In contrast, the discriminator uses a multi-class discriminant method using mask segmentation images as the different types to improve discrimination[21]. The joint probability of image tokens and their related layout tokens are observed using a joint-decoding transformer which gives extra observed data to describe complicated scenes—added with Layout-Vqgan invested in encoding and decoding extra information concerning complicated scenarios.

Attention approach to object generator may provide fine-grained features on targeted objects using an effective dual generator[22]. When the initial images are not properly formed, the fuzzy image contents can be refined with the help of a dynamic memory module introduced by the proposed method. The researchers developed a novel approach called multi-perspective fusion to improve text-to-image synthesis. In this approach, the generator incorporates a dynamic selection mechanism to match text features with image features, enabling more accurate synthesis. Meanwhile, the discriminator utilizes a multi-class discriminant method, where mask segmentation is introduced as an additional type to enhance its discrimination capacity[23]. The proposed framework, called RaSeedGAN (RAndomly-SEEDed super-resolution GAN), is designed to evaluate field quantities from randomly sparse sensors without relying on full-field high-resolution training. By utilizing random sampling, the algorithm gains fragmentary perspectives of the high-resolution underlying distributions. Even when sparse or noisy, the findings are promising[24]. The authors considered methodologies for generating statistical properties of the text. On the other hand, synthesizing the corresponding images for text input is challenging[25]. Synthesising high-quality images for computer vision applications is a challenging task. To overcome the disadvantage of a single-stage GAN network, the authors brought a two-stage network, resulting in an enhanced resolution image[26–28]. This work proposes keyframes selection strategy for video description using a boundary-based method that allows the system to encode visual information by picking a small subset of keyframes and construct a video description without considerable degradation[29].

Text-to-image synthesis model

The architecture is designed for synthesizing images for corresponding Tamil text descriptions. The architecture proposed works well based on the statistical information. Since Tamil is a morphologically rich language, it is necessary to look into the language model and the synthesizer network. A new dataset was created by applying Google Translator to translate the English text of Caltech UCSD -Birds 200 and Oxford-102 datasets into the Tamil language. The image synthesis architecture is shown in Fig. 1.

The model leverages the ability of TBERTBASECASE to generate meaningful sentence embedding. The model is fused with TBERTBASECASE and BASEGAN by introducing language model word Embedding. The proposed method generates improved resolution over realistic images for Tamil text on two demanding datasets.

Dataset. The Oxford 102 flower dataset and CUB 200 text have been collected. Tamil sentence for the corresponding English text is translated using Google Translate to create the corpus and was trained. The Oxford-102 contains 102 classes, each with a set of image files and descriptions. Figure 1 shows how Google Translate translated English sentences into Tamil text files, probably about 3000 sentences. The CUB 200 dataset was also developed for experimentation. Figure 2 shows the translation.

TBERT BASECASE model. The language model is a preliminary work that focuses on understanding the statistical properties of the input text, which results in the generation of text vectors. The BERT model is the initial transformer framework developed for processing text which is enhanced into various versions such as masked, unmasked, cased, and uncased, depending upon the corpus under learning and language taken for training. The proposed language model tried to use the BERT BASECASE model trained over Tamil text corpus by fine tuning the last layer of processing. The sentence's feature study is done using a BERT named BERT BASECASE model. The proposed model is the TBERTBASECASE model, which is trained over Tamil input text and aims to study the properties of the sentence and generate corresponding text vectors. The BERT model is advantageous over other language models like GLOVE embedding, etc.[30,31]. Since the algorithm works bi-directional

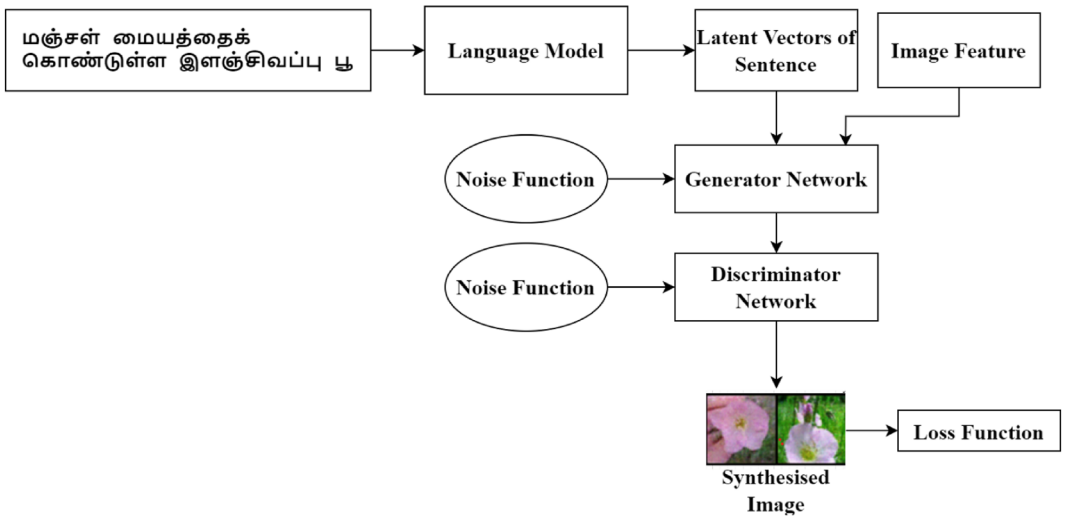

Figure 1. Text to Image Synthesis Architecture.

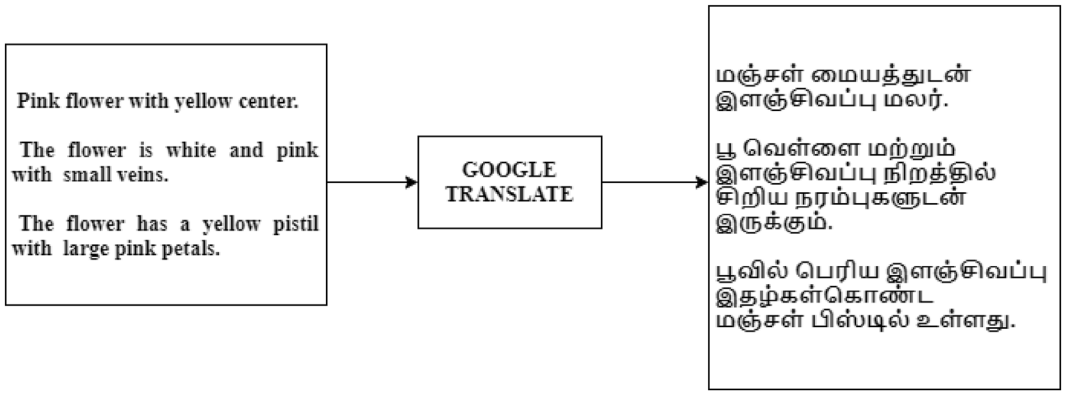

Figure 2. Text Translation using Google Translate.

from left and right, it is unique in processing the text features[32]. The proposed TBERT BASE CASE model takes input text sentences followed by hidden layers. The TBERT BASECASE model has 12 layers which have 768 hidden units. It can process data with hyperparameters which are around 110 M. In Tamil the morphology of the language itself is a challenging task.

Token embeddings are the initial token representations which is feed in to BERT transformer blocks. Transformer blocks start with self-attention. It lets each token recognise and value all other tokens in the input sequence. Self-attention compares tokens and creates weighted sums of token embeddings to calculate attention weights. This method helps the model grasp local and global input sequence dependencies. After self-attention, the transformer block has residual connections. Residual connections let the model preserve token embedding information and avoid the vanishing gradient problem during training. Self-attention output is added element-by-element to token embeddings. A transformer block refines the tokens in the input sequence. Contextualized embeddings combine static token embeddings with contextual information from self-attention and the FFN. BERT has numerous transformer blocks stacked. The model captures more complicated relationships and refines token representations at each layer by passing the output of each block to the next block. BERT's word embeddings are the last transformer block's token representations.

The schematic representation of text processing using the TBERTBASECASE is presented in Fig. 3, and embedding vectors of the text are depicted in Table 1.

Consider the given set of sentences as t and the text token values from $[t_1 \ldots t_n]$ fed as input tokens to the input layer. The results pass through various hidden layers, which are customized according to the given Tamil input feature tokens and generate feature vectors $[f_1, f_{2..} f_n]$ by employing concatenation or summing up the last layer vectors of the hidden layers of the transformer depicted as H_o. The objective function of the TBERT BASE CASE model is represented as,

$$\varphi(t) = t_1, t_2, \ldots t_n \tag{1}$$

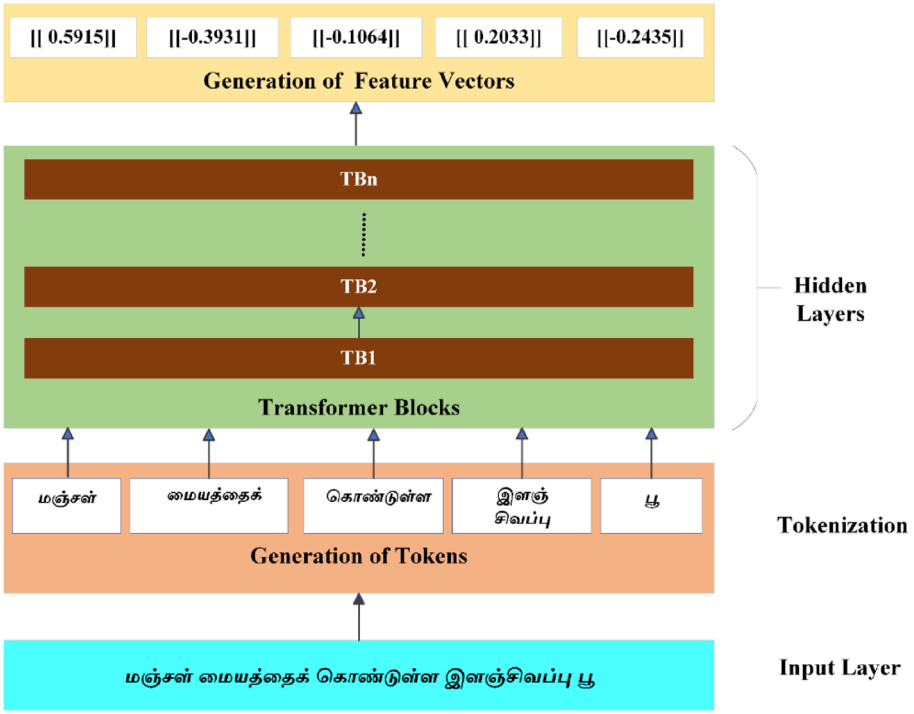

Figure 3. TBERTBASECASE language model.

Tamil text	Tokens	Romanized pronunciation	Meaning	Text vector
மஞ்சள் மையத்தைக் கொண்டுள்ள இளஞ்சிவப்பு பூ	மஞ்சள்	məɲdʒəl̪	Yellow	[[0.5915]]
	மையத்தைக்	məjjət̪t̪əjk	Center	[[− 0.3931]]
	கொண்டுள்ள	koɳɖʉl̪l̪ə	Containing	[[− 0.1064]]
	இளஞ்சிவப்பு	ʲɪl̪əɲdʒɪʋəppʉ	Pink	[[0.2033]]
	பூ	pu	Flower	[[− 0.2435]]

Table 1. Feature vector representation of the input text.

$$H_o(t) = H(t_1), H(t_2), \ldots H(t_n) \tag{2}$$

$$logT_{BERTBASECASE} \approx \sum_{1}^{n} log(H(t_1)||\cdots||H(t_n)) \tag{3}$$

$$Embedding_{Text} = Elemetwise\ sum[T_i \ldots T_n] + [C_0 \ldots C_k] \tag{4}$$

where n denotes the number of tokens in the given text, t is the Tokens of the given text, and Eq. (1) represents the various text tokens of the text. The concatenation of vectors in hidden layer is represented in Eq. (2). Equation (3) represents the objective function of the TBERT BASE CASE model. Each hidden layer consists of T_i... T_N-Token embeddings and C_0...C_k—contextualised embeddings and its Elementwise sum is drafted in Eq. (4).

BASEGAN model for image generation. The Generator Discriminator network (GAN) proposed is a base image synthesis model deployed using Tamil text-to-image synthesis network. Earlier, many image synthesis model was proposed, which works using the encoder output, image captions, and so on[33,34]. The model under study works by receiving latent vectors from the previous language model, named the TBERT BASE CASE

model. The latent output vectors are fed into the generator network, the noise function, and the preprocessed image feature vectors. Similarly, the output of the generator network is fed into the discriminator along with the generator noise function.

The two noise functions sent along with the text feature vectors are MINIMAX and Adversarial loss function[35]. The architecture is represented in Fig. 4.

The BASEGAN model has input image vectors of the image as I and text feature vectors as φ(t). The image vectors are transformed to tensor values of size $I_x \times I_y \times 3$ which is the size of the image vector. The generic objective function of a GAN network s represented as Eq. (5),

$$minmaxGD(G, D) = E_{x\sim pdata}[loglog(D(x)] + E_{z\sim pz}[loglog(1 - D(G(z)))] \tag{5}$$

The noise function $N(0, 1)$ is supplied as a random value to the generator G, and the discriminator D. Let φ(t), the text vector, and I be the corresponding image vector. The GAN model is trained in a fashion where the generator maximizes the loss L_D of the discriminator and minimizes the generator loss L_G. The minimizing and maximizing equation is given by,

$$L_D = E_{(I,t)\sim pdata}[loglogD(I, \varphi(t))] + E_z \sim pz[log(1 - D(G(z), \varphi(t)))] \tag{6}$$

$$L_G = -E_{z\sim pz}[log(D(G(z), \varphi(t))] \tag{7}$$

where $G(Z)$—Noise Function of Generator. $D(x)$—Discriminators probabilistic value that real instance is real. $E_{x\sim pdata}$-Expected overall real data. L_D—Discriminator Loss Function. L_G—Generator Loss Function. $E_{(I,t)\sim pdata}$—Expected over-all real data(image and text vectors). $E_{z\sim pz}$—Expected overall random inputs to the generator (input image and text vectors)

Equation (6 and 7) represents the loss function of a discriminator in maximizing and minimizing its objective function. *pdata* refers to the distribution of the feature vector over the naturalistic image, and *pz* refers to the data distribution over the latent space concerning the discriminator. In Eq. (6)The first term denotes the expectation over the logarithm of the discriminator's output when given real samples x. The discriminator aims to maximize this term, correctly classifying real samples as close to 1. The second term represents the expectation over the logarithm of the discriminator's output when given generated samples G(z). Here, z represents the random noise vector sampled from a prior distribution. The discriminator aims to minimize this term, classifying generated samples as close to 0.Eq. (7) represents the generator loss function.The expectation over the logarithm of the generated samples G(z). z is the random noise vector sampled prior distribution. The generator maximises this term to make the discriminator classify the generated samples as real.

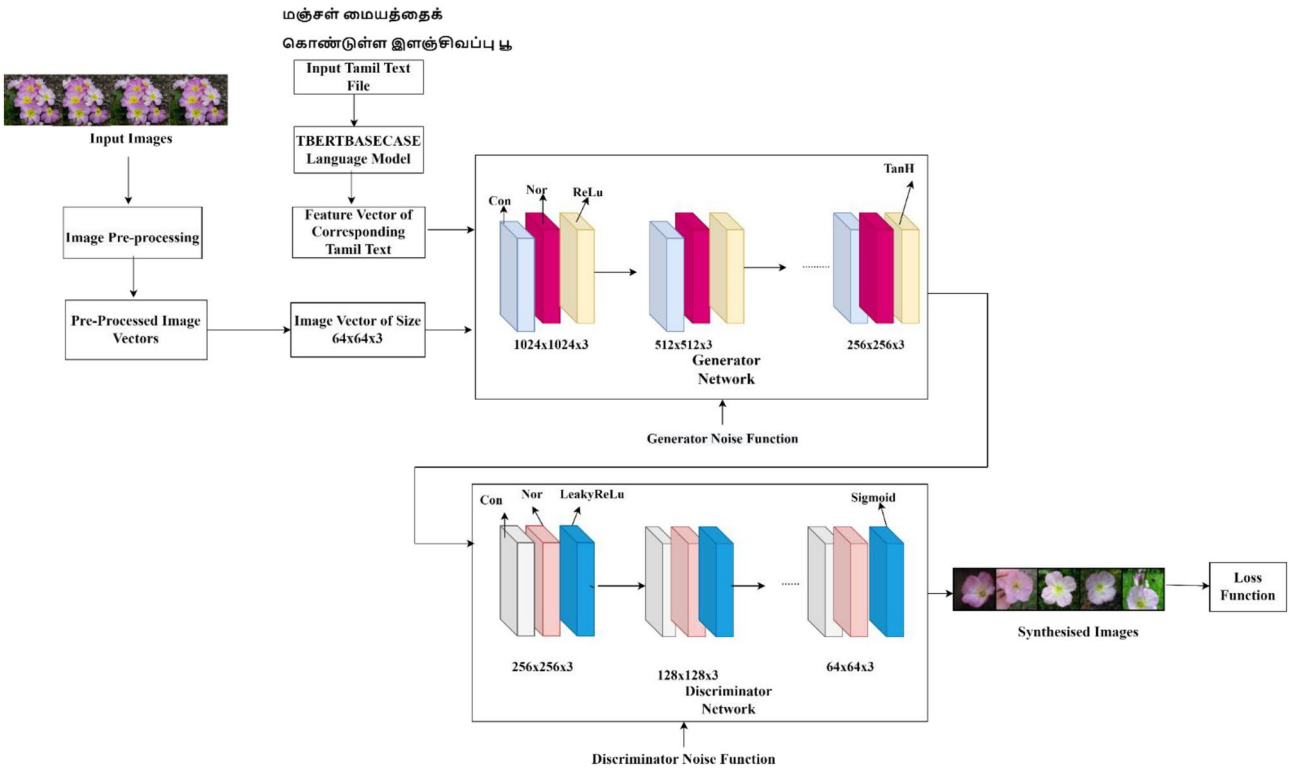

Figure 4. BASEGAN image synthesis network.

Hybrid super resolution GAN (HSR GAN). The Hybrid Super Resolution GAN (HSR GAN) works similarly to a two-stage GAN model. The super-resolution GANs[36] proved to be a better image synthesis model in various applications than the conventional GAN model. The two-stage model works by super-resoluting the synthesised images from stage I. Many applications work with single-stage GAN network, but still it lacks minute information which are essential parts of a synthesised image. The model works with residual blocks. The Super Resolution GAN[37] has two-stage models which work similarly to the Base GAN model considered in stage I, and the output is fed into stage II for improving resolution. The model works like residual blocks[38] that have splitting, transforming, and aggregating functions. The vector representation of the image vectors is provided in the form of latent vectors where the essential feature vectors from latent space[39] can be pointed to generate images by the generator. The image size of 1024×1024 is fed as an input to the generator of stage I, along with word embeddings from the TBERT BASECASE language model. The model architecture of the two-stage Hybrid Super Resolution GAN network is described in Fig. 5.

The Stage II GAN receives output images from stage I to further improve its resolution. The hybrid super-resolution GAN works. The Wasserstein loss function is employed in both the generator and discriminator[40]. The Wasserstein function's significant advantage is bringing down the difference between original and fake images generated[41]. The two components involved are the natural distribution of values P_{Real} and $P_{Generated}$, the distribution of values over the images generated. In mathematical form, it is the minimum distance to follow the Earth mover's distance calculated by considering the transfer plan in which data is converted to authentic and generated images. The gain of experiencing the Wasserstein loss function is to create a large gap across real and fake images generated, which intends to solve the problem of understanding the realness of the image that is being generated. Moreover, it also helps in stabilizing the model during training of the model. The Wasserstein loss function creates a notion for the discriminator as a critic where it tries to bring out important gradient information everywhere in the training model.

The representation of HSRGAN is as follows: I^{HR} refers to the high resolution of the image being generated and I^{LR} being the representation of the low-resolution image generated by the stage I stage network. The objective function is represented in Eq. (8)

$$minmaxE_{I^{HR}}(G,D) = E_{x\sim pdata}\left[log\log D\left(I^{HR}\right)\right] + E_{I^{LR}\sim pI^{LR}}\left[log log\left(1 - D\left(G\left(I^{LR}\right)\right)\right)\right] \tag{8}$$

The loss function corresponding to stage II is Wasserstein loss which is represented as,

$$Loss_{WDG} = E(C(x_{real})) - E(C(G(z)) \tag{9}$$

From Eq. (9), it is clear that the Wasserstein loss function aims to evaluate the critic value of the real image distribution and the fake images generated. C denotes the critic value of the loss function. The critic value is any number, and the discriminator output is determined as critic output which depends on the Lipschitz constant, which is depicted in Eqs. (10 and 11)

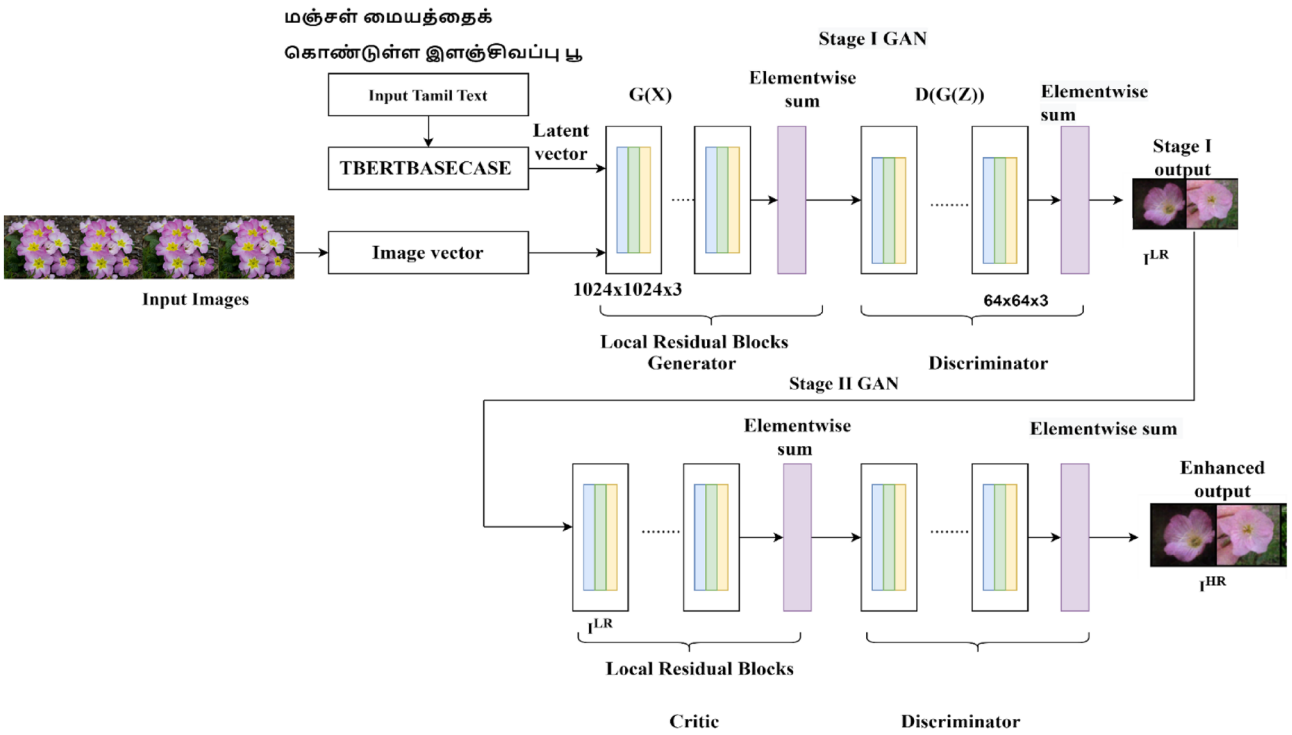

Figure 5. The architecture of Hybrid Super Resolution GAN (HSR GAN) model.

$$c = 1 - Lipschitz\ Continuous \tag{10}$$

$$Distance_{EMD} = \sum_{i=1}^{m} \sum_{j=1}^{n} [x_{Real} G(z)]ij \tag{11}$$

where I^{LR}—being the representation of low-resolution image that is generated by stage-I stage network. I^{HR}—refers to the high resolution of the image that is being generated by the stage-II network. $E_{I^{HR}}$—Expected overall Real Data of high-resolution image generated by stage-II network. $E_{I^{LR} \sim pI^{LR}}$—Expected overall Real Data of low-resolution image generated by stage I network. $D\left(G\left(I^{LR}\right)\right)$—Probabilistic value that a fake instance of a low-resolution image is actual. C—The critic value is any number, and the discriminator output is determined as the critic output. $Distance_{EMD}$—Earth mover distance between real and fake distribution.

Results and discussions

The model is evaluated using the F1 score, FID, and IS. The IS calculates the Kullback–Leibler (KL) divergence between conditional distribution p(y|x) and marginal distribution p(y), the InceptionScore.F1 score calculates the Precision and Recall values of the synthesised images. The resolution of the images tends to be of lower grade, and it can be enhanced over a two-stage GAN which is explained in the forthcoming model. The image obtained as resultant is convincing, but still it lacks the high-resolution features. It paves the way for enhancing the output from a BASEGAN model using the HSRGAN model. The testing and training set is represented in Table 2.

Figures 6 and 7 shows the images generated as a result of BASEGAN and resolution improvement through HSRGAN; from Fig. 8a,b, resolution improvement to a better level which can be dealt with by the Mean Squared Error (MSE) and Structural Similarity Index measure (SSIM) across epochs of training and the values on SSIM. When MSE is considered to evaluate the minimal similitude of the natural and fake images, the score more resembles the images because it provides a better reconstruction of the images. In initial epochs, the MSE value is at a higher rate, gradually decreasing, which helps reconstruct images. MSE figures out the deviations of the estimated features of images generated in terms of the square of the difference of pixel values. In the given case, there is no discrepancy between the images of stage I and those generated from stage II in terms of similar pixel values, which proves that there is only enhancement of features at two levels.

Datasets	Train	Test
CUB -200	9414	2374
Oxford-102	7034	1155

Table 2. Dataset of CUB-200 and Oxford-102.

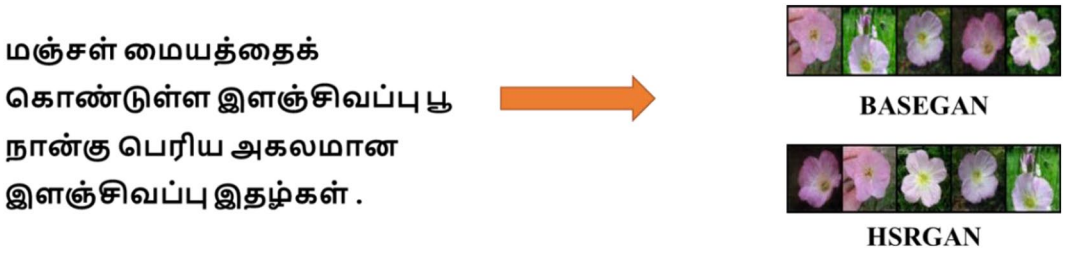

Figure 6. Generated Images Using BASEGAN and HSRGAN Network on Oxford-102 Dataset.

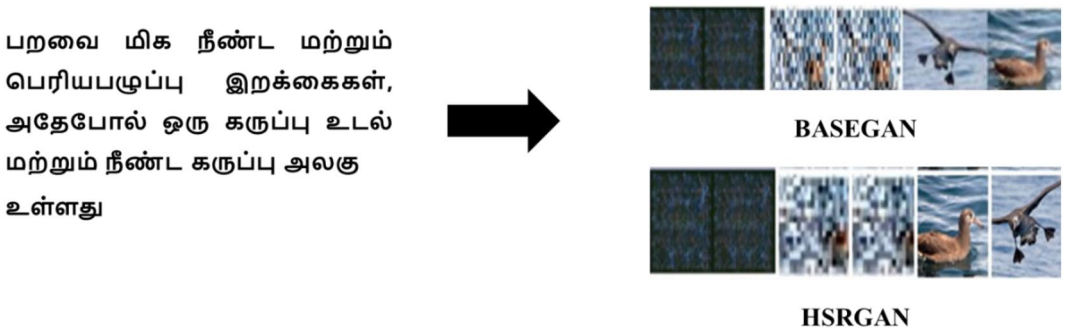

Figure 7. Generated Images Using BASEGAN and HSRGAN Network on CUB Dataset.

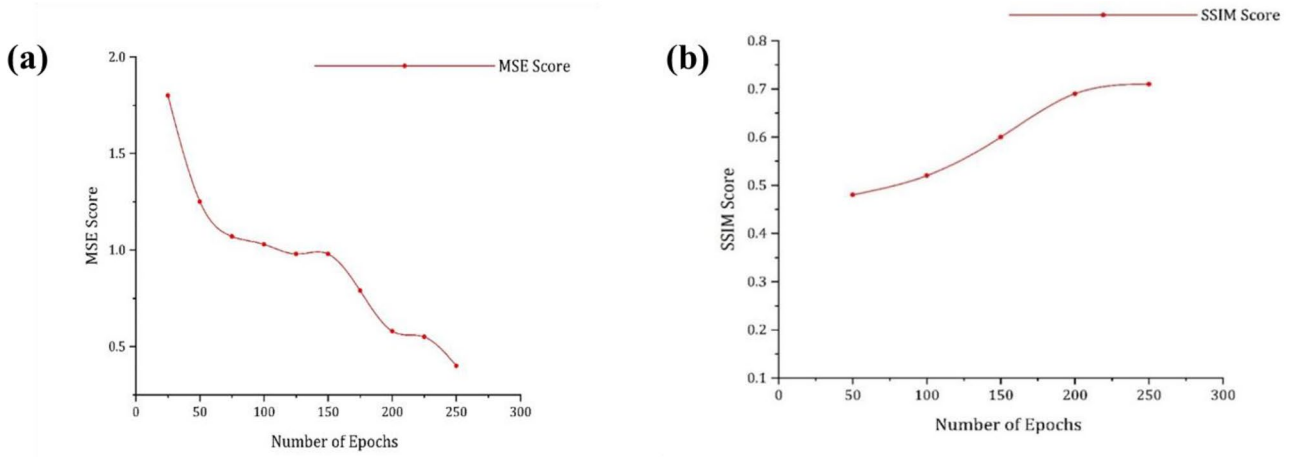

Figure 8. (**a**) MSE score for generated image quality Using HSRGAN (**b**) SSIM similarity index for generated image quality Using HSRGAN.

Similarly, SSIM is also used in measuring perceived image quality. It measures the brightness; the contrast of the image, and it also symbolizes that the higher the value of SSIM then, the quality of the image generated. The formulation of MSE and SSIM Eqs. (12 and 13).

$$MSE = \frac{1}{PQ}\sum_{n=0}^{P}\sum_{m=1}^{Q}[I^{HR}(n,m) - I^{LR}(n,m)]^2 \tag{12}$$

$$SSIM(I^{HR}, I^{LR}) = \frac{(2\mu_{HR}\mu_{LR} + C_1)(2\sigma_{HR}\sigma_{LR} + C_2)}{(\mu_{HR}^2\mu_{LR}^2 + C_1)(\sigma_{HR}^2\sigma_{LR}^2 + C_2)} \tag{13}$$

where μ_{HR} and μ_{LR} are the means, σ_{HR} and σ_{LR} are the standard deviations, and σ_{HRLR} is the cross-covariance for generated images x and real image input sequentially.

The Inception Score is represented in Eq. (14) as follows,

$$IS = \exp\left(E_x D_{KL}\left(P(y|x)||P\left(y\right)\right)\right) \tag{14}$$

x is a produced imagery analysis, and y is a pre-trained Inception v3 network image label[40]. Higher IS produces higher-quality images that are classified. Another Fréchet distance metric is FID. FID predicts feature space pictures using the pre-trained Inception v3 network. Equation (15) shows the FID score.

$$FID = ||\mu_x - \mu_z||^2 + Trace\left(\sum\nolimits_x + \sum\nolimits_z - 2 * \sqrt{\sum\nolimits_x \sum\nolimits_y}\right) \tag{15}$$

Where, mean and covariance of the real and synthesised image. In contrast to IS, FID on the minimum value shows more realistic images and the distributions are alike. The F1 measure provides a combined way to represent precision and recall that captures both properties. Equation (16) represents the F1 measure,

$$F1\ Measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{16}$$

GAN proved to be of outstanding practice in image synthesis applications. The model in hand for image synthesis for Tamil has two levels of performance the Base GAN model and the HSRGAN model. The primary concept of the proposed work is that it is the novel idea brought into the limelight for the Tamil language, which is at the development stage. It has a long way to explore. The proposed architecture can be a starting point for further enhancement in Tamil language that supports various applications in the regional language. At the initial stage, the language model aims to generate vector representations for Tamil text input. On the rear end, the GAN models synthesize images corresponding to the text vectors. The strength of the model lies in how far it can generate images.

Applications in English language have been at par, and various models have been proposed. The model proposed by the authors works for text-to-image and reversal in a vice-versa format[42]. The model employed by the authors is the Image-Text-Image model using the GAN model. The method can be used for any calligraphy. The CASIA-HWDB dataset and Chinese database of handwritten characters were tested with the model. Transfer models gave them convincible accuracy over the characters'[43]. Compared to English, much has been done. Pre-trained English text vectors like WordtoVec, Skip thinking vectors, Glove embeddings, and others can preprocess text. The GAN model matches AttnGAN. GAN uses image and text encoders. Using a common MS-COCO

dataset, bidirectional LSTMs processed text and data from white, top, pillows, and table groups. Evaluation scored using BLUE-1, BLUE-2, 3, and 4.

However, past studies generated photos with English text descriptions, which required much technology[44]. The methodology proposed uses challenging and relevant Tamil text descriptions. The primary model is the work brought to lime light in the Tamil language. Previous work in this research is done for the English language. So, a comparative performance evaluation has been done with embedding vectors generated by the TBERTBASECASE language model along with the pre-trained GAN network like AttnGAN, and Stack + + GAN. The proposed architecture is evaluated with the existing architecture, and its performance is assessed.

Figure 9 shows the valuation of IS metric on CUB-200 and Oxford-102 test sets. Among the proposed models, HSRGAN outperforms the BASEGAN and pre-trained GAN architecture. BASEGAN is comparatively better, along with the other pre-trained models. Figure 10 portrays the performance concerning the FID score. Here too, the proposed HSRGAN outperforms the trained architecture. The scores are convincing and satisfactory. Next on the run, Fig. 11 proves the performance of the architectures for the F1 score. Among the provided models, the projected work proves to be better in evaluation. Despite Tamil text processing's complexity, the proposed model outcome matches StackGAN + + . Thus, quantitative comparisons indicate that a persuasible performance in synthesising realistic and high-resolution images conditioned with Tamil text has been accomplished.

Conclusion

Synthesis of images for text is an intriguing stream of research in computer vision. Image synthesis for Tamil text is critical since it has a rich morphology. TBERTBASECASE language and BASEGAN models are in the early stages of combining the Tamil language with the well-known GAN model. However, still, the performance of the basic architecture is convincible. It needs an improvement which is incorporated using a super-resolution

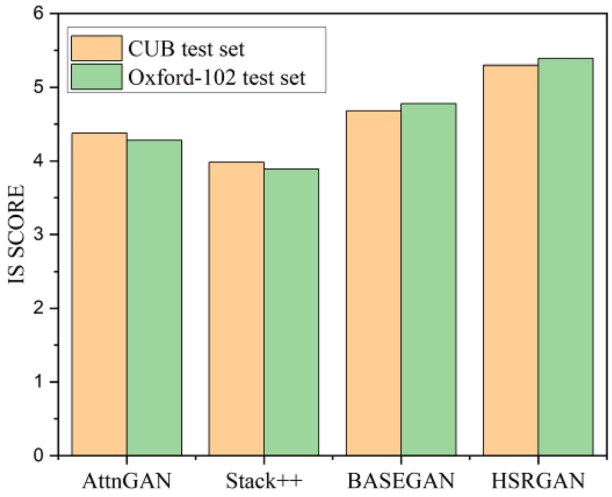

Figure 9. Inception Score evaluation.

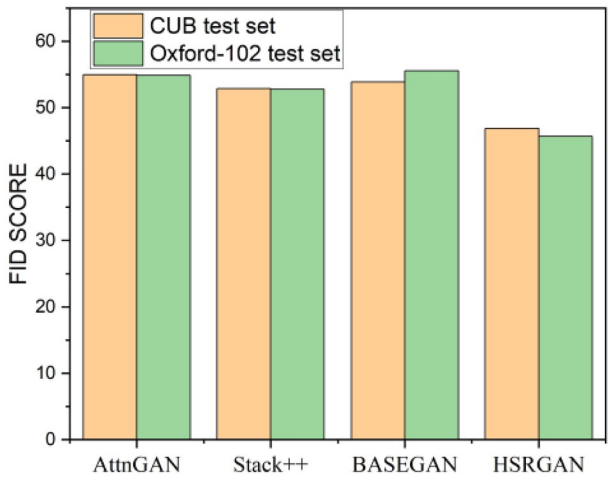

Figure 10. FID score performance estimation.

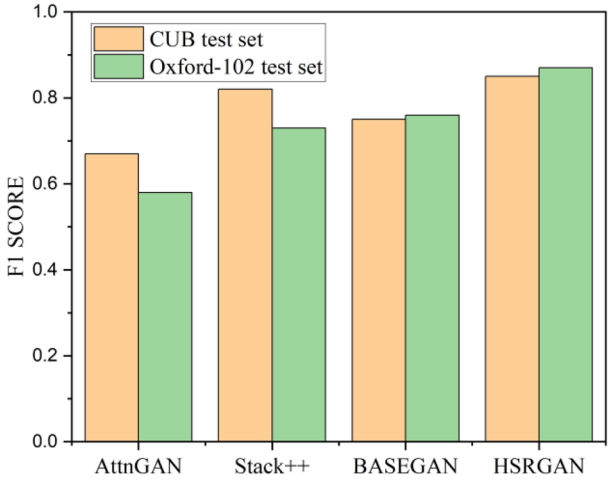

Figure 11. F1-Score valuation.

GAN architecture. The BASEGAN model is enhanced by using a two-stage GAN model named Hybrid Super Resolution GAN(HSRGAN) to revamp the resolution of the images further. The model can be further improved using enhanced patterns of high-resolution GAN models and autoregressive models.

Data availability

The datasets generated and analysed during the current study are not publicly available since it is related to the doctoral thesis. After the submission of the research work, it would be provided. Nevertheless, are available from the corresponding author on reasonable request.

Received: 6 June 2023; Accepted: 28 August 2023
Published online: 02 September 2023

References

1. He, Y., Li, J., & Zhu, A. Text-Based Image Style Transfer and Synthesis. In *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, pp. 43–48. IEEE. (2019).
2. Yang, S., Liu, J., Yang, W. & Guo, Z. Context-aware text-based binary image stylization and synthesis. *IEEE Trans. Image Process.* **28**(2), 952–964 (2018).
3. Zhang, Z., Zhou, J., Yu, W., & Jiang, N. Drawgan: Text to image synthesis with drawing generative adversarial networks. In *ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4195–99. IEEE. (2021).
4. Isola, P., Zhu, J. Y., Zhou, T., & Efros, A. A. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1125–34. (2017)
5. Chrysos, G. G., Kossaifi, J. & Zafeiriou, S. Rocgan: Robust conditional gan. *Int. J. Comput. Vision* **128**, 2665–2683 (2020).
6. Li, Y., Wang, J., Zhang, X., & Cao, Y. FittingGAN: fitting image generation based on conditional generative adversarial networks. In *2019 14th International Conference on Computer Science & Education (ICCSE)*, pp. 741–45. IEEE. (2019).
7. Zakraoui, J., Saleh, M., Al-Maadeed, S. & Jaam, J. M. Improving text-to-image generation with object layout guidance. *Multimed. Tools Appl.* **80**(18), 27423–27443 (2021).
8. Nugroho, K. S., Sukmadewa, A. Y., Yudistira, N. Large-scale news classification using BERT language model: Spark NLP approach. In *6th International Conference on Sustainable Information Engineering and Technology 2021*, pp. 240–46. (2021).
9. Zhu, J. *et al*. Incorporating BERT into neural machine translation. arXiv preprint arXiv:2002.06823. (2020).
10. Na, S., Do, M., Yu, K. & Kim, J. Realistic Image Generation from Text by Using BERT-Based Embedding. *Electronics* **11**(5), 764 (2022).
11. Wang, Z., Quan, Z., Wang, Z. J., Hu, X., & Chen, Y. Text to image synthesis with bidirectional generative adversarial network. In *2020 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6. IEEE. (2020).
12. Škrlj, B., Martinc, M., Lavrač, N. & Pollak, S. autoBOT: evolving neuro-symbolic representations for explainable low resource text classification. *Mach. Learn.* **110**, 989–1028 (2021).
13. Du, Y., Li, Q., Wang, L. & He, Y. Biomedical-domain pre-trained language model for extractive summarization. *Knowl. Based Syst.* **199**, 105964 (2020).
14. Naseem, U., Razzak, I., Khan, S. K. & Prasad, M. A comprehensive survey on word representation models: From classical to state-of-the-art word representation language models. *Trans. Asian Low-Resour. Lang. Inf. Process.* **20**(5), 1–35 (2021).
15. Liu, S.-H., Chen, K.-Y. & Chen, B. Enhanced language modeling with proximity and sentence relatedness information for extractive broadcast news summarization. *ACM Trans. Asian Low-Resour. Lang. Inf. Process. (TALLIP)* **19**(3), 1–19 (2020).
16. Gao, P., You, H., Zhang, Z., Wang, X., & Li, H. Multi-modality latent interaction network for visual question answering. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 5825–35. (2019).
17. Lan, Z. *et al*. Albert: A lite bert for self-supervised learning of language representations. arXiv preprint arXiv:1909.11942. (2019).
18. Kim, Y., Ra, D. & Lim, S. Zero-anaphora resolution in Korean based on deep language representation model: BERT. *ETRI J.* **43**(2), 299–312 (2021).
19. Lee, J.-S. & Hsiang, J. Patent classification by fine-tuning BERT language model. *World Patent Inf.* **61**, 101965 (2020).
20. Wang, B. *et al*. On position embeddings in BERT. In *International Conference on Learning Representations*. (2021).
21. Zhang, Z., Fu, C., Zhou, J., Yu, W., & Jiang, N. Text to image synthesis based on multi-perspective fusion. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE. (2021).
22. Wu, F., Liu, L., Hao, F., He, F., Cheng, J. Text-to-image synthesis based on object-guided joint-decoding transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18113–22. (2022).

23. Zhang, H., Zhu, H., Yang, S. & Li, W. DGattGAN: Cooperative up-sampling based dual generator attentional GAN on text-to-image synthesis. *IEEE Access* **9**, 29584–29598 (2021).
24. Lei, J. *et al.* HFF-SRGAN: super-resolution generative adversarial network based on high-frequency feature fusion. *J. Electron. Imaging* **31**(3), 033011–033111 (2022).
25. Güemes, A., Vila, C. S., & Discetti, S. Super-resolution GANs of randomly-seeded fields. arXiv preprint arXiv:2202.11701. (2022).
26. Wang, M. *et al.* End-to-end text-to-image synthesis with spatial constrains. *ACM Trans. Intell. Syst. Technol. (TIST)* **11**(4), 1–19 (2020).
27. Mukherjee, S., Kumar, P. & Roy, P. P. Fusion of spatio-temporal information for indic word recognition combining online and offline text data. *ACM Trans. Asian Low-Resour. Lang. Inf. Process. (TALLIP)* **19**(2), 1–24 (2019).
28. Zhang, H. *et al.* Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 5907–15. (2017).
29. Singh, A., Meetei, L. S., Singh, S. M., Singh, T. D., & Bandyopadhyay, S. an efficient keyframes selection based framework for video captioning. In *Proceedings of the 18th International Conference on Natural Language Processing (ICON)* pp. 240–250. (2021).
30. Chiu, S. H., Chen, B. Innovative BERT-based reranking language models for speech recognition. In *2021 IEEE Spoken Language Technology Workshop (SLT)*, pp. 266–71. IEEE. (2021).
31. Sabharwal, N, Agrawal, A., Sabharwal, N., & Agrawal, A. BERT Model Applications: Question answering system. In *Hands-on Question Answering Systems with BERT: Applications in Neural Networks and Natural Language Processing*: 97–137. (2021).
32. Geetha, M. P. & Renuka, D. K. Improving the performance of aspect based sentiment analysis using fine-tuned Bert Base Uncased model. *Int. J. Intell. Networks* **2**, 64–69 (2021).
33. Zakir, H. M., Sohel, F., Shiratuddin, M. F., Laga, H. & Bennamoun, M. Text to image synthesis for improved image captioning. *IEEE Access* **9**, 64918–64928 (2021).
34. Hidayaturrahman, Dave, E., Suhartono, D. & Arymurthy, A. M. Enhancing argumentation component classification using contextual language model. *J. Big Data* **8**, 1–17 (2021).
35. Kula, S., Kozik, R. & Choraś, M. Implementation of the BERT-derived architectures to tackle disinformation challenges. *Neural Comput. Appl.* https://doi.org/10.1007/s00521-021-06276-0 (2021).
36. Balakrishnan, V. *et al.* A deep learning approach in predicting products' sentiment ratings: A comparative analysis. *J. Supercomput.* **78**(5), 7206–7226 (2022).
37. Li, Y., Cao, J., Li, Z., Oh, S., & Komuro, N. Lightweight single image super-resolution with dense connection distillation network. In *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 17(1s): 1–17. (2021).
38. Wang, X. *et al*. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*. (2018).
39. Xie, S., Girshick, R., Dollár, P., Tu, Z., & He, K. Aggregated residual transformations for deep neural networks. 2016. arXiv preprint arXiv:1611.05431. (2017).
40. Li, Z. *et al.* Interpreting the latent space of gans via measuring decoupling. *IEEE Trans. Artif. Intell.* **2**(1), 58–70 (2021).
41. Diviya, M., & Karmel, A.TAM GAN: Tamil Text to Naturalistic Image Synthesis Using Conventional Deep Adversarial Networks. *ACM Trans. Asian Low Resour. Lang. Inf. Process.* 1–18 (2023).
42. Dong, H., Zhang, J., McIlwraith, D., & Guo, Y. I2t2i: Learning text to image synthesis with textual data augmentation. In *2017 IEEE international conference on image processing (ICIP)*, pp. 2015–19. IEEE. (2017).
43. Chang, B., Zhang, Q., Pan, S., & Meng, L. Generating handwritten Chinese characters using cyclegan. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 199–207. IEEE. (2018).
44. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818–26. (2016).

Author contributions

D.M. wrote the main manuscript text and prepared the figures and tables. K.A. administered and gave suggestions on improvement of work All authors reviewed the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to A.K.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.