Introduction

The Waste Management Plan's goal is to explain the waste management concepts, processes, and management at the BML MUNJAL UNIVERSITY campus. This Plan was created by THE ACTIVISTS to ensure that waste is reduced, reused, and recycled as much as feasible.

The Plan contains information on the following topics:

1. The types and amounts of trash produced throughout the procedure.
2. Waste collection and disposal procedures.
3. Measures that will be taken to reduce the amount of trash generated as a result of the development.

Abstract

Presently in India, about 960 million tonnes of solid waste is being generated annually as by-products of industrial, mining, municipal, agricultural and other processes. With rapid population growth and urbanization, this number will keep increasing at an unprecedented rate.

The substantial increase in solid waste generation results in the contamination of air, water, and land resources. Indian cities and towns are found littered with garbage, these practices create serious health, safety, and environmental consequences.

Poorly managed waste serves as a breeding ground for disease vectors and contributes to global climate change through methane generation.To safeguard the environment, efforts are being made for recycling different wastes and utilizing them in value-added applications. This report presents the status of the generation and utilization of solid wastes at BML Munjal University, their environmental implication is reported and discussed in detail.

Problem Statement

A problem statement is a short, clear explanation of an issue or challenge that summarizes what you want to change. It helps you, team members, and other stakeholders to focus on the problem, why it's important, and who it impacts

Contents

Literature Review

The authors[1] proposed a method called LeicaGAN, consisting of a textual-visual co-embedding network (TVE), a multiple priors aggregation network (MPA) and a cascaded attentive generator (CAG). The text encoder was a pre-trained Bi-directional LSTM and the visual encoder was built upon the Inception-v3 model. The MPA network fused the sentence level embeddings. This acted as an input in the CAG, where an attention block, two residual blocks, an upsampling block and a convolution layer make up the generator. Word and Sentence-context features were produced. Two adversarial losses were employed: a visual realism adversarial loss to ensure that the generators generate visually realistic images and a text-image pair-aware adversarial loss to guarantee the semantic consistency between the input text and the generated image. For effectiveness LeicaGAN was compared with AttnGAN. CUB and Oxford-102 datasets were used and evaluation was done based on the Inception Score. LeicaGAN outperformed AttnGAN, on both the datasets.

The authors[2] proposed ControlGAN. For this, they introduced a word-level spatial and channel-wise attention-driven generator that could disentangle different visual attributes. Also, they proposed a word-level discriminator. The backbone architecture they used was AttnGAN and the text encoder was a pre-trained bi directional RNN. Conditioning Augmentation was applied. The generator exploited the attention mechanism via incorporating a spatial attention module and the channel-wise attention module. The spatial attention module dealt with words with individual spatial locations. The model was experimented on CUB and MS COCO datasets. The model proposed was compared with AttnGAN and StackGAN++ and the performance metrics were Inception Score, R-precision and L2 Error. ControlGAN gave the best results among the three for the CUB dataset.(Inception Score = 4.58 ± 0.09, Top-1 Acc(%) = 69.33 ± 3.23, L2 error = 0.18). For the COCO dataset, AttnGAN was the best in Inception Score and Top-1 Acc(%), but ControlGAN had the lowest L2 error, i.e., 0.17.

The authors[4] proposed a StackGAN model. The model is built in two stages: Stage 1 GAN giving Low Resolution images and Stage 2 GAN giving High resolution images. The model first processes the text input and generates corresponding text embeddings to feed into the generative adversarial networks. The model included a text encoder and decoder implemented with a word-level bidirectional recurrent neural network (RNN) consisting of two long short-term memory (LSTM). The generator and discriminator receive a conditioning variable. Dataset used was COCO. The pre-trained StackGAN model has decent performance on generating images from a text input that is similar to its training set, although, when the input contains multiple objects, StackGan fails to generate the correct number of instances with clear boundaries and spatial relationships.

The authors[5] implemented DC-GAN conditioned on text features encoded by a hybrid character-level convolutional recurrent neural network. In the generator, a text query was encoded. The description embedding was compressed using a fully connected layer followed by LeakyReLU as the activation function. This was then concatenated with the noise. The discriminator consisted of several layers of strides-2 convolution with spatial batch normalization followed by LeakyReLU. The experiment was done on the CUB and Oxford-102 datasets. The GAN baseline was compared with GAN-CLS with image-text matching discriminator, GAN-INT learned with text manifold interpolation and GAN-INT-CLS which combined both.

This paper[6] described a transformer trained to autoregressive model the text and image tokens as a single stream of data. A two-stage training procedure was used: Training a discrete Variational Autoencoder to compress an 256 x 256 image into a 32 x 32 grid of image tokens and concatenating up to 256 BPE-encoded text tokens with the 32 × 32 = 1024 image tokens and train an autoregressive transformer to model the joint distribution over the text and image tokens. For a text-image pair, the lowercase caption is BPE-encoded using at most 256 tokens with vocabulary size 16,384. The image is encoded using 32 × 32 = 1024 tokens with

vocabulary size 8192. The image tokens are obtained using argmax sampling from the dVAE encoder logits. The text and image tokens are concatenated and modeled autoregressive as a single stream of data. The experimentwas carried out of Conceptual Captions, an extension of MS COCO. The model is compared with AttnGAN, DM-GAN and DF-GAN. The evaluation metrics were Inception Score and Fréchet Inception Distance. The zero-shot model obtained an FID score on MS-COCO within 2 points of the best prior approach, despite having
never been trained on the captions. But, the model fares significantly worse on the CUB dataset, for which there is a nearly 40-point gap in FID between it and the leading prior approach, i.e., DM-GAN.

The authors[7] implemented a GAN-CLS including a generator and discriminator. The inputs were batches of images and the matching text. Both the matching and mismatching text description are encoded, noise is added. Three inputs are passed to the Discriminator: Correct Text with actual image, Incorrect Text with actual image and Correct Text with fake image. These help in the better training of Discriminator. The dataset used was the Oxford-102 flower dataset.

The authors[8] implemented a StackGAN to generate a stylised output image directly from the model. The Stage 1 GAN generated low resolution images and a new conditioning is added to the Stage 2 GAN to generate higher resolution images in a given style. The discriminator is trained on stylised 256 x 256 images. The datasets used were COCO and CUB. No meaningful results were obtained from this method. The reason given, too much time to reasonably generate enough stylized training data and to perform a hyper-parameter search with enough iterations each time to find the best settings

Methodology

Data was collected with the help of a device named “Mind Wave Mobile” made by the NeuroSky company. It’s an EEG headset which is equipped with the biosensor technology that makes it easy for us to gather the EEG data from our brain. A very useful feature of this device is that it gathers data and then do the basic pre-processing after which we can fetch the data in a .csv format.

Fig. 2: MindWave Mobile [2]

Process of data collection is as follows:

1. Data is collected from a single human subject.
2. Subject is asked to sit in a room with minimal noise and disruption so that we can maintain data quality.
3. MindWave Mobile headset is mounted on the subject and correct placement of the headset is also necessary.
4. The ground of the headset is placed on the ear clip and the EEG sensor is placed on the end of the arm which is placed on the forehead.

Fig. 3: Subject during data collection

5. The headset connects to the phone via Bluetooth, and some parameters must be set by the user in order for the data to be recorded.
6. At the time of data collection, the subjects consider one of four directions: forward, backward, left, or right.
7. The subject is asked to sit for an hour and think about the four mentioned directions one at a time and data is stored in a .csv file.
8. The data is recorded for one minute at one-second intervals, yielding 60 rows of data in a single session.
9. This yields an average of 20 csv files, 5 in each direction, for a total of 1200 rows of data.

Once the process of data collection is completed then different Machine Learning and Neural Network Model are used for Classification

- **LSTM**

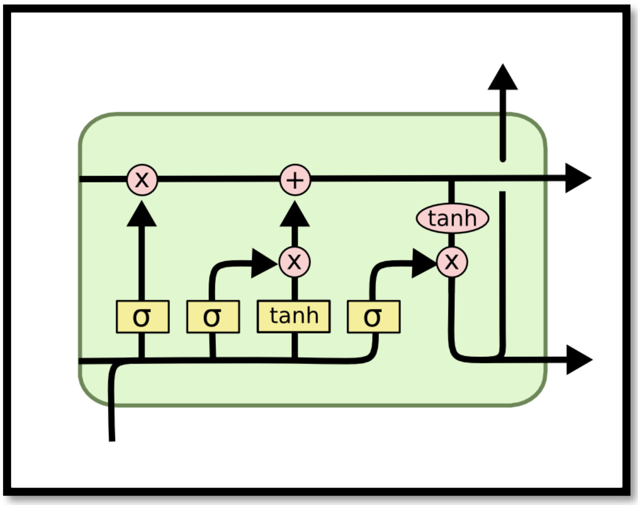

Fig. 4: Gated LSTM Cell

The data is first processed, basic algorithms for standardisation, outlier detection, etc. results in important features which are fed to the model. Then different layers are added to train the model and improve its accuracy.

- **KNN**

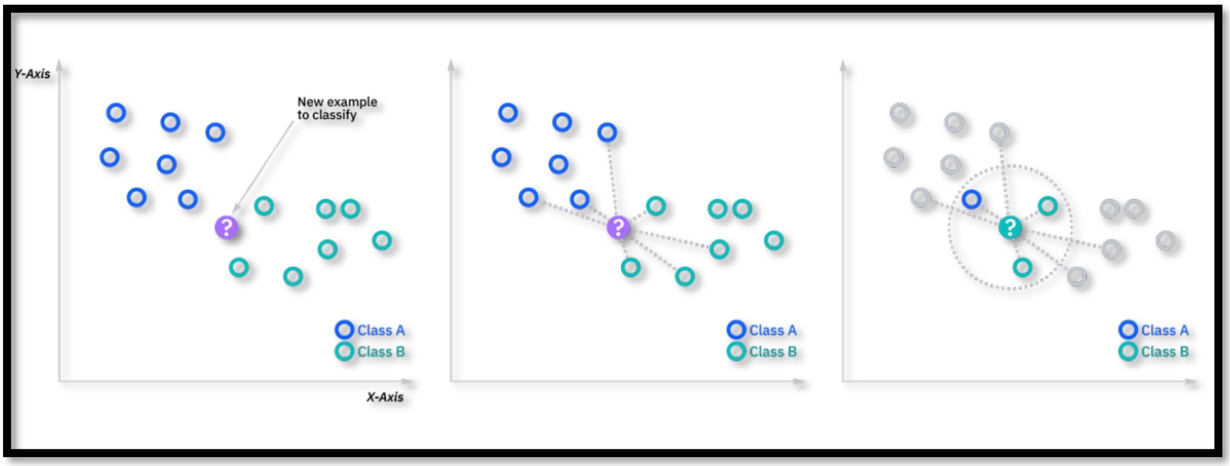

Fig. 5: KNN Diagram

The k-nearest neighbours algorithm, or KNN for short, is a supervised learning classifier that employs proximity to produce classifications or predictions about how a particular data point will be grouped. Although it can be applied to classification or regression issues, it is commonly employed as a classification algorithm because it relies on the idea that comparable points can be discovered close to one another. A class label is chosen for classification problems based on a majority vote, meaning that the label that is most frequently represented around a particular data point is used.

- **Random Forest**

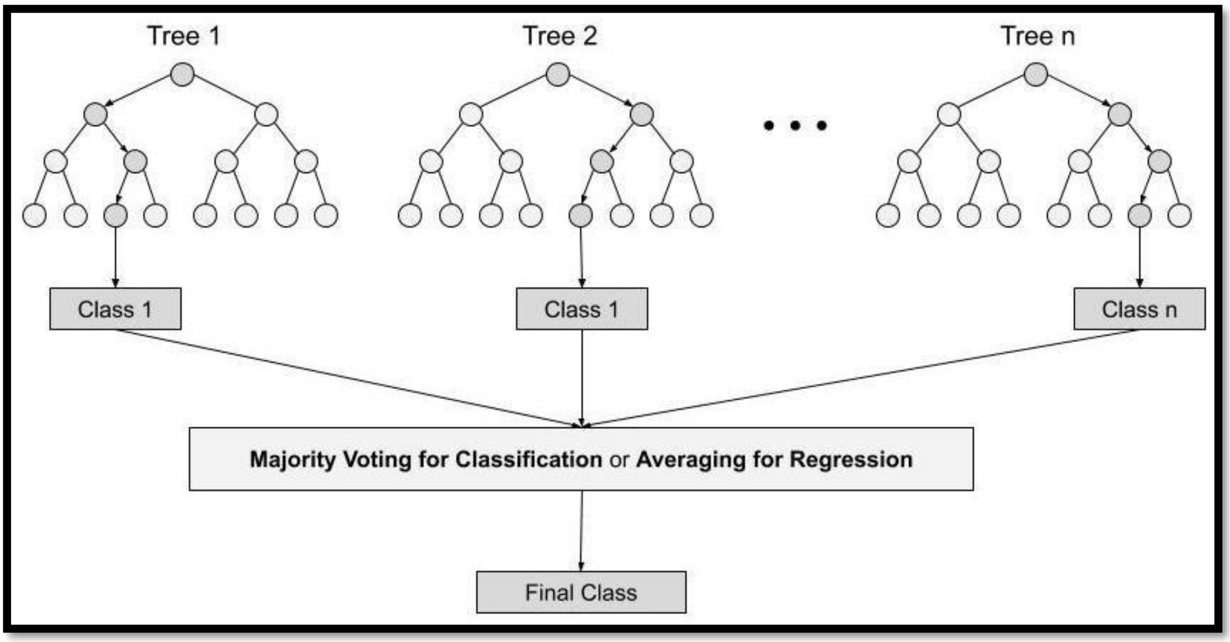

Fig. 6: Random Forest Diagram

A huge number of distinct decision trees work together as an ensemble in a random forest. The term "ensemble" refers to a class of methods that combine several learning algorithms to provide predictions that are more accurate than those produced by any one of the individual learning algorithms used in the ensemble. Every single tree in the random forest spits out a class forecast, and the classification that receives the most votes becomes the prediction made by our model.

Data Description

Over the course of four months, data was collected from a single subject in the directions forward, backward, left, and right. The data set contained 15,459 rows in total. We are now working with 11 columns in this project after removing unnecessary columns.

This is how the data set appears:

	attention	meditation	delta	theta	alphaLow	alphaHigh	betaLow	betaHigh	gammaLow	gammaMid	direction
0	50	63	47335	16755481	26686	6106	7214	2678	4946	1295	Forward
1	41	53	807962	381706	24030	11299	17020	22975	8633	2995	Forward
2	53	54	74205	31596	23686	4736	5837	11982	2429	3667	Forward
3	48	38	1154845	62690	10877	13968	9558	10730	6538	3292	Forward
4	66	29	4595	20664	5238	3855	4256	7509	14710	3759	Forward

Fig. 7: Data Set

Refrences