Synthetic Data Generation for Food Waste Pelletizer using TGANs, TVAEs

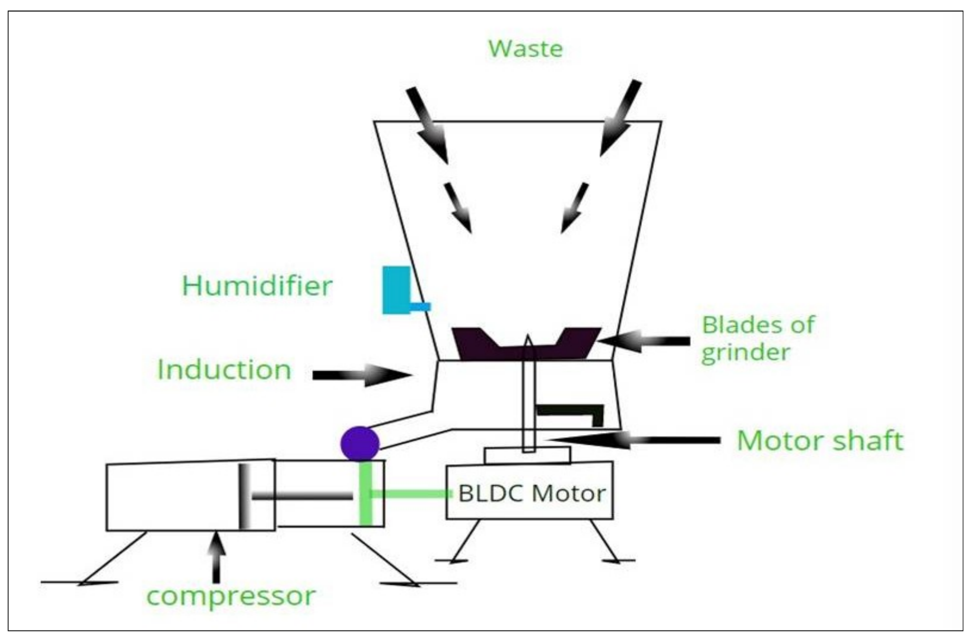
Pattan Afrid Ahmed | Dr. Priyadarshini J | SCOPE

Introduction:

Our main goal of synthetic data generation is because building an experimental dataset is costly and time consuming. For automation of food waste pelletizer using Fuzzy logic and the working require defining a proper set of rules or membership functions and validating with simulations referencing towards the dataset. Generative algorithms like Tabular Generative Adversarial Network (TGANs), Tabular Variational Auto-encoders (TVAEs) are mainly used for synthetic tabular data generation. Generated dataset is evaluated using different metrics like KV divergence, KS Test, GM log Likelihood. Simulation of the fuzzy control system applied with respect to variable speed and temperature.

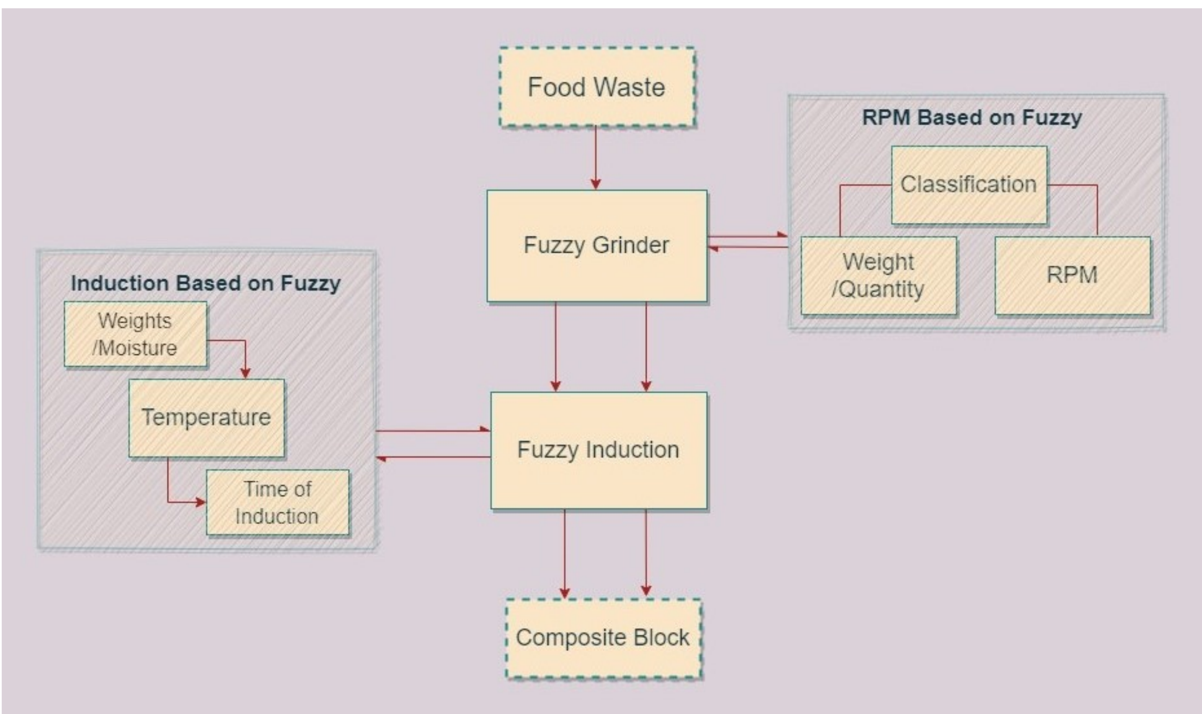
SCOPE of Project:

- Prepared and raw food waste materials often establish challenges to the food makers in its dumping and prompt transportation.



- The proposed project aims at clean dumping of the food wastes with a significantly reduced space and its usage as cattle-feed which consists of both hardware and software portions.
- The level of operation and other process parameters are controlled by fuzzy logic.

Methodology:



Conditional TGAN are applying the mode-specific normalization to overcome the non-Gaussian and multimodal distribution and deal with the imbalanced discrete columns.

Generator Network

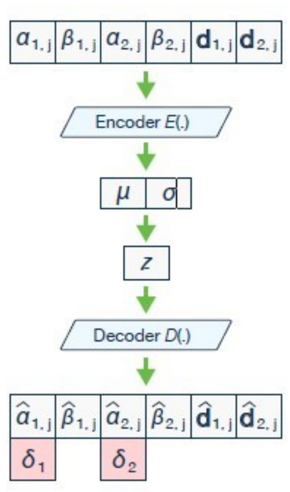
$$\begin{cases} h_0 = z \oplus cond \\ h_1 = h_0 \oplus \texttt{ReLU}(\texttt{BN}(\texttt{FC}_{|cond|+|z|\to 256}(h_0))) \\ h_2 = h_1 \oplus \texttt{ReLU}(\texttt{BN}(\texttt{FC}_{|cond|+|z|+256\to 256}(h_1))) \\ \hat{\alpha}_i = \texttt{tanh}(\texttt{FC}_{|cond|+|z|+512\to 1}(h_2)) & 1 \le i \le N_c \\ \hat{\beta}_i = \texttt{gumbel}_{0.2}(\texttt{FC}_{|cond|+|z|+512\to m_i}(h_2)) & 1 \le i \le N_c \\ \hat{\mathbf{d}}_i = \texttt{gumbel}_{0.2}(\texttt{FC}_{|cond|+|z|+512\to |D_i|}(h_2)) & 1 \le i \le N_d \end{cases}$$

Discriminator Network

$$\begin{cases} h_0 = \mathbf{r}_1 \oplus \ldots \oplus \mathbf{r}_{10} \oplus cond_1 \oplus \ldots \oplus cond_{10} \\ h_1 = \texttt{drop}(\texttt{leaky}_{0.2}(\texttt{FC}_{10|\mathbf{r}|+10|cond|\to 256}(h_0))) \\ h_2 = \texttt{drop}(\texttt{leaky}_{0.2}(\texttt{FC}_{256\to 256}(h_1))) \\ \mathcal{C}(\cdot) = \texttt{FC}_{256\to 1}(h_2) \end{cases}$$

$$\begin{cases} \mathbf{r}_j = \texttt{cat}(\alpha_{1,j}, \beta_{1,j}, \ldots, \alpha_{N_c,j}, \beta_{N_c,j}, \mathbf{d}_{1,j}, \ldots, \mathbf{d}_{N_d,j}) \\ h_1 = \texttt{ReLU}(\texttt{FC}_{|\mathbf{r}_j|\to 128}(\mathbf{r}_j)) \\ h_2 = \texttt{ReLU}(\texttt{FC}_{128\to 128}(h_1)) \\ \mu = \texttt{FC}_{128\to 128}(h_2) \\ \sigma = \exp(\frac{1}{2}\texttt{FC}_{128\to 128}(h_2)) \\ q_\phi(z_j|\mathbf{r}_j) \sim \mathcal{N}(\mu, \sigma\mathbf{I}) \end{cases}$$

By using CTGANs with VGM models along Wasserstein GAN loss penalty, we were able to generate data which closely approximated the original data.



Variatinal Auto-encoder (TVAE) outperformed the CTGAN. The major key between TVAE and CTGAN is that, generator network does not have direct access to the real data distribution during training, unlike TVAE. It works by approximating the data distribution.

Results:

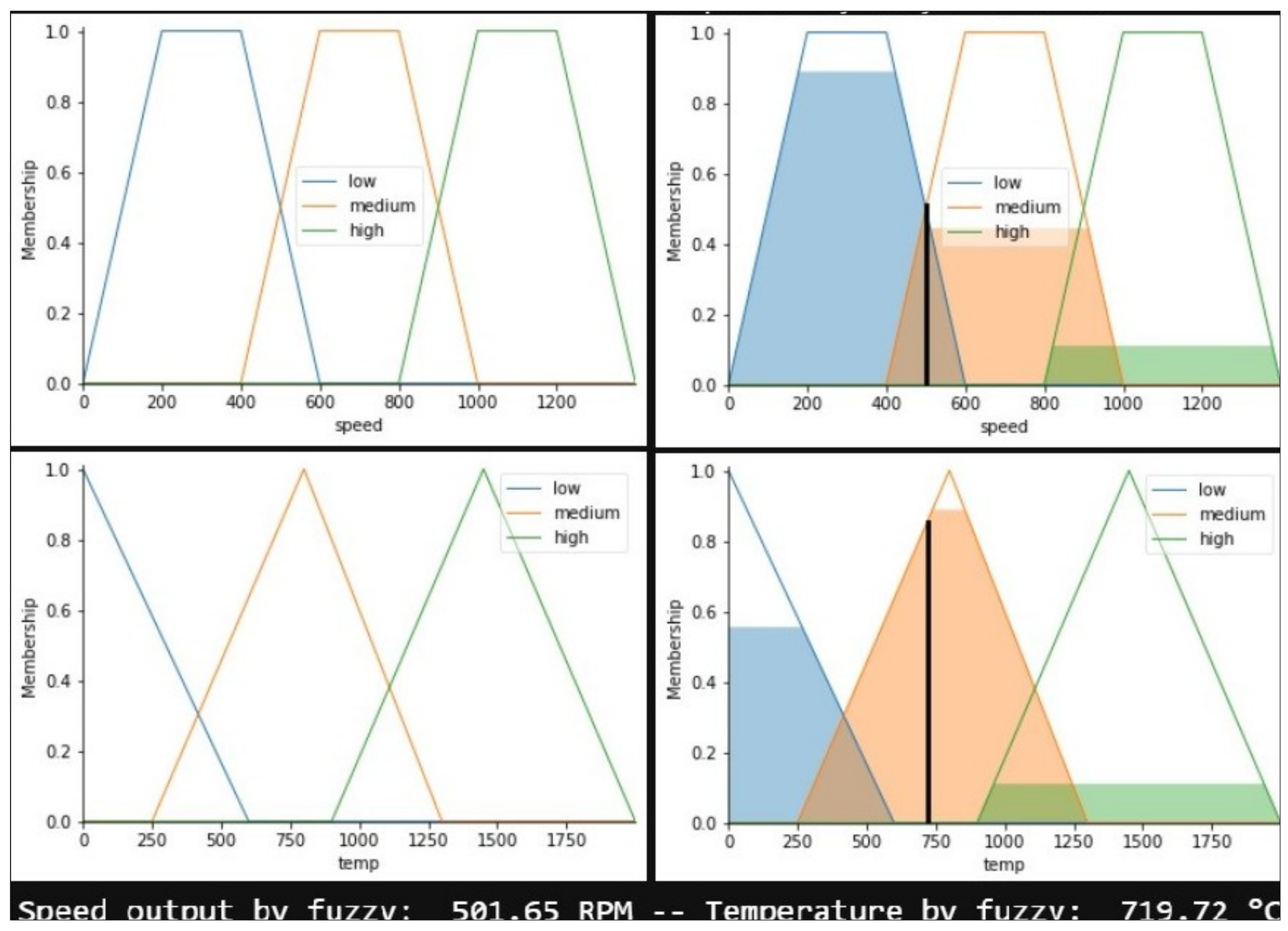
TVAE synthetic data scores

	metric	name	raw_score	normalized_score
	LogisticDetection	LogisticRegression Detection	7.738095e-01	0.773810
	SVCDetection	SVC Detection	8.690476e-01	0.869048
	GMLogLikelihood	GaussianMixture Log Likelihood	-2.871795e+09	0.000000
	KSTest	Inverted Kolmogorov-Smirnov D statistic	7.348485e-01	0.734848
	KSTestExtended	Inverted Kolmogorov-Smirnov D statistic	7.348485e-01	0.734848
	ContinuousKLDivergence	Continuous Kullback–Leibler Divergence	2.382109e-01	0.238211

CTGAN synthetic data scores :

	metric	name	raw_score	normalized_score
	LogisticDetection	LogisticRegression Detection	6.496599e-01	0.649660
	SVCDetection	SVC Detection	3.639456e-01	0.363946
	GMLogLikelihood	GaussianMixture Log Likelihood	-1.339852e+10	0.000000
	KSTest	Inverted Kolmogorov-Smirnov D statistic	6.477273e-01	0.647727
	KSTestExtended	Inverted Kolmogorov-Smirnov D statistic	6.477273e-01	0.647727
	ContinuousKLDivergence	Continuous Kullback–Leibler Divergence	1.697142e-01	0.169714

Fuzzy logic simulation graphs:



Conclusion:

Generated data using CTGANs is used for simulations with respect to temperature and speed. By applying fuzzy logic the whole automated process of grinding, moisture reduction by induction heating and conversion to compost block can be achieved as an end product. This reduces the storage space, transportation cost, release of harmful gases and it can be fed to cattle or used for agriculture purposes.

Contact Details:

pattanafrid.ahmed2020@vitstudent.ac.in

priyadarshini.j@vit.ac.in

References:

[1] Intelligent Waste Separator and Recycling using IoT and Evolutionary Fuzzy Logic Technology with energy efficient sensing during Hajj and Umrah crowd seasons - Salma Mahgoub Gaffer Elhag, ISSN 2393-8021, Vol. 6, Issue 2, February (2019).

[2] “Thermal treatment on sewage sludge by electromagnetic induction heating: Methodology and drying characterization, Waste Management” - Yongjie Xue, Chen , Zhenhua Hu, 0956-053X, (2018).

[3] Modeling Tabular data using Conditional GAN - Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, Kalyan Veeramachaneni- arXiv:1907.00503 (2019).

[4] Synthesizing Tabular Data using Conditional GAN - Lei Xu, MIT (2020).