VFAST Transactions on Software Engineering
http://vfast.org/journals/index.php/VTSE@ 2022, ISSN (e):2309-3978, ISSN(p): 2411-6246
Volume 10, Number 3, January-December 2022 *pp:24-36*

A Brief Survey on Natural Language Processing Based Text Generation and Evaluation Techniques

Philemon Philip[1] and Sidra Minhas[1]

[1]Computer Science Department, Forman Christian College, Lahore, Pakistan
Corresponding author email philemonphilip98@gmail.com

ABSTRACT
Text Generation is a pressing topic of Natural Language Processing that involves the prediction of upcoming text. Applications like auto-complete, chatbots, auto-correct, and many others use text generation to meet certain communicative requirements. However more accurate text generation methods are needed to encapsulate all possibilities of natural language communication. In this survey, we present cutting-edge methods being adopted for text generation. These methods are divided into three broad categories i.e. 1) Sequence-to-Sequence models (Seq2Seq), 2) Generative Adversarial Networks (GAN), and 3) Miscellaneous. Sequence-to-Sequence involves supervised methods, while GANs are unsupervised, aimed at reducing the dependence of models on training data. After this, we also list a few other text generation methods. We also summarize some evaluation metrics available for text generation and their Performance

KEYWORDS
Natural Language Processing, Text generation, GANs, Sequence-to-Sequence, Evaluation Matrices

JOURNAL INFO
HISTORY: Received: 26 July 20122 Accepted: Septeber 15, 2022 Published: September 27, 2022

1. INTRODUCTION

Natural Language Processing (NLP) is a field of Artificial Intelligence (AI) that focuses on teaching computers how to read and understand textual data [1]. NLP allows computers to be able to read textual data, understand sentiments, interpret it, and generate new text. NLP was first presented by Richard Bandler and John Grinder in the 1970s [2] as Neuro-Linguistic programming used for psychotherapy.

They attempted to recognize the patterns in the thinking and actions of successful personalities and to use them to teach others and published a series of eminent studies. Neuro-Linguistic Programming later evolved as NLP and replaced the term Text Mining to suit more general-purpose tasks.

The field of NLP can be divided into two categories shown in Figure 1. There are two main kinds of NLP-based tasks: 1) Analyzing and 2) Generating [3]. The first is the analysis task which focuses on analyzing the text (sentences, paragraphs, etc). An example of these tasks can be determining whether a given sentence is grammatically correct or not, if it is a positive or a negative statement, etc. The second NLP-based task is the generation task. Generation tasks are used to create a brand-new text with or without some former input data. Examples of these can be summarization, generating answers to questions, etc. Two popular models to perform generation tasks, as of 2022 are, Sequence-to-Sequence and GANs (explained below).

Over the previous decade, analysis of text using NLP models has reached good accuracy. While text analysis remains an interesting task, text generation is considered more challenging and intensive. NLP is being widely used in multiple applications where programs study the interaction between computers and human language. Systems like autocorrect, autocomplete, google translation, grammar checking, chatbots, social media monitoring, email filtering, smart assistants, language translators, etc [4], all of them are based on NLP. These NLP models are very precise and refrain from unwanted information and their accuracy increases as it learns their users writing and thinking styles. It can answer questions about any subject in different languages and is much faster than a human response. NLP is all around us yet still to be perfected as fields remain still that are far from being perfect. This field is not only interesting but also is growing rapidly. However, training a new NLP model and finding the most accurate one is time-consuming as the model must match a user's thinking and writing style. In this paper, we present a survey of recent cutting-edge works done in the domain of text generation. First, we explain the theory behind texr generation and later we explain the most commonly used techniques in this field.

2. TEXT GENERATION PIPELINE

Figure 2 shows a general life cycle of the text generation models in NLP. Firstly the data is collected (datasets) according to the requirements of our task at hand. Then the given datasets are preprocessed and/or cleaned. In the preprocessing phase things like removing stop words, removing punctuation, tokenization, lemmatization, word parsing, stemming, lower casing, etc could be done based on

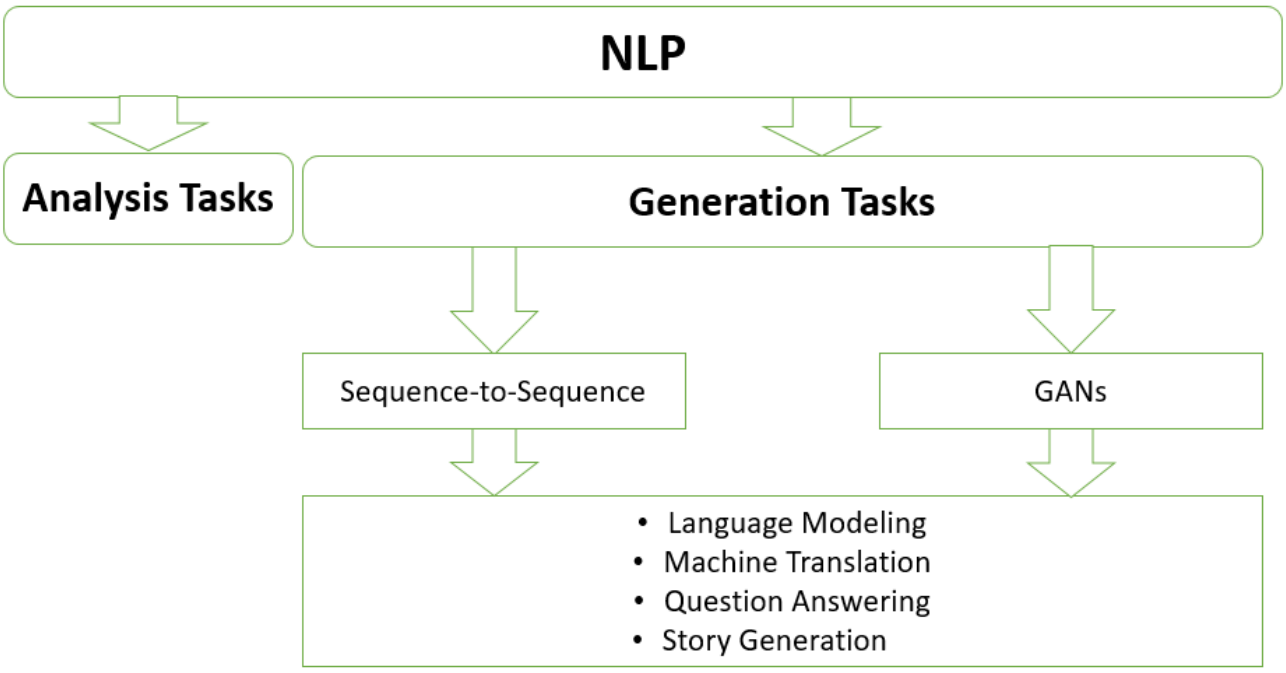

Figure 1: NLP Based Tasks

the task [5]. After preprocessing the feature extraction is done. Some popular methods of feature are term frequency (tf: how many times a word occurs in a document) , inverse document frequency(idf: measure to check if a word is common or rare in a corpus [6]) and term frequency-inverse document frequency (tf-idf: which is a product of both tf and idf) [7]. These can be expressed in a bag-of-words. It can be expressed as a model that converts text into fixed length vectors. This conversion can be done with the help of tf, idf and tf-idf [8]. After that, the text generation model is created and starts to generate text, based on some input. Then the text generation models are evaluated. Based on the evaluations of the model it is improved. The model is improved until the desired result is achieved.

Text generation is a task that leverages information from a certain corpus of textual data to generate a Language Model (LM) [9]. An LM determines the probability of a given sequence of words occurring in a sentence using statistical and probabilistic techniques to generate text as an output [10]. Several simple LMs exist in the literature including N-gram, Unigram, bidirectional, exponential, etc, however, these techniques were limited due to their context and application size. Various neural networks were employed to cater to the exponential growth of digital device users over the past decade and the increased demand for accurate text generators

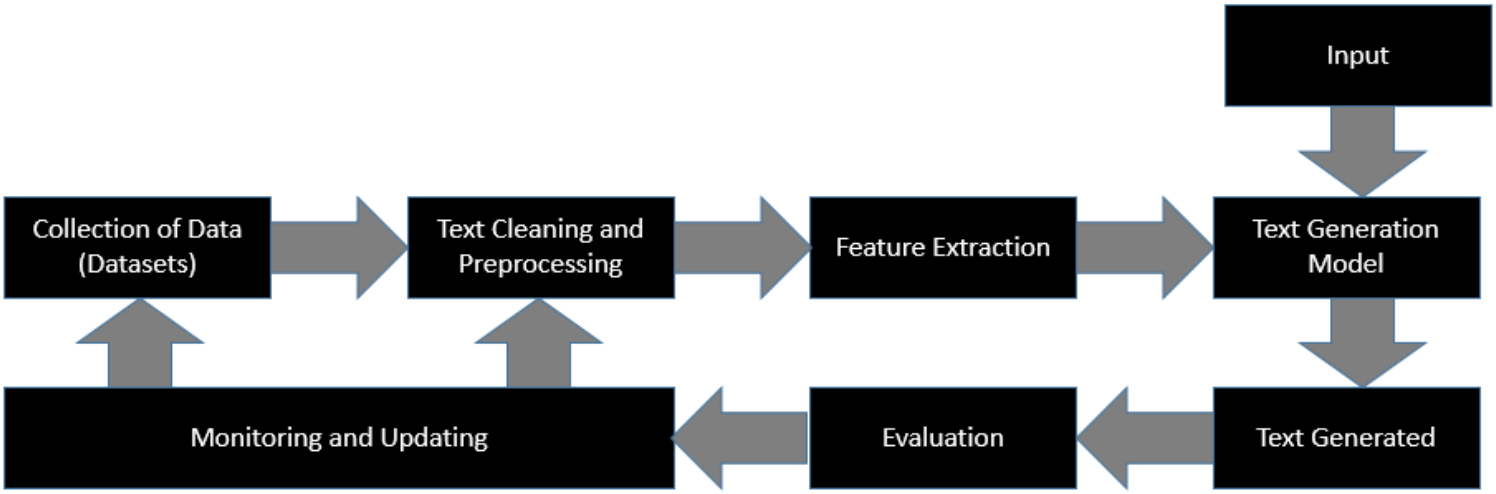

Figure 2:Text Generation Life-Cycle

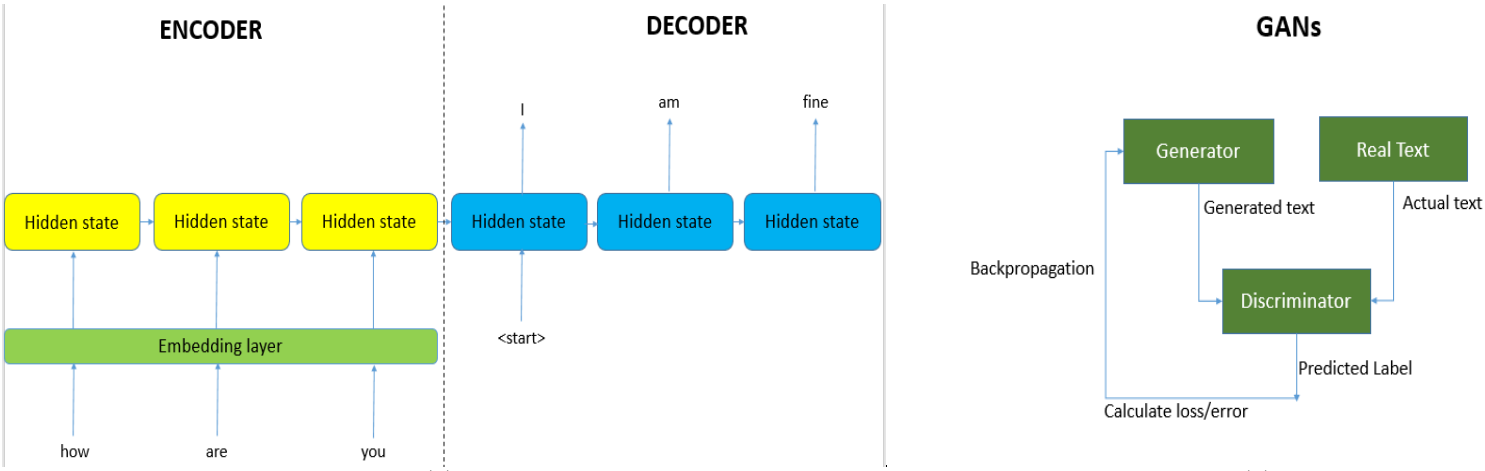

Figure3: (a) Sequence-to-Sequence, (b) GANs

[11]. Two major types of deep learning-based text generators recently being excessively studied are the Sequence-to-Sequence (Seq2Seq) Models and the Generative Adversarial Networks (GAN).

Sequence to Sequence (seq2seq) models are a class of Recurrent Neural Networks (RNN) that take human text as input and produce an output of a new sequence of textual data. In other words this model takes a sequence of items as inputs and produces another sequence as output [12]. Many popular systems around us use Seq2Seq models. These include google translate and many online chatbots. It is a type of encoder-decoder deep learning network using neural networks like Long Short Term Memory (LSTM) to generate output [13]. There are two components in this i.e. an encoder and a decoder. The encoder reads the input text and sends it to the hidden state (or a context vector). At the hidden state, the LSTM reads all the data one by one in a loop. Each hidden state is calculated keeping by checking the current word and the previous hidden state. Then the decoder reads the data from the encoders. The output or the result from the encoder is going to be the input for the decoder. Based on these inputs the decoder will start to generate the output sequence. LSTM is used in the prediction of these words. Like with the encoder, the decoder takes the previous hidden state and the current input in context to find the current hidden state. The output is calculated using the current hidden state with a certain weight assigned to it. The most common activation function used is softmax. It converts number vectors into probabilities [14].Softmax is used to find out the final output based on the probabilities of each token. The output will have a start and end token added to it word tokens as well. These tokens will indicate the start and end of a line. After the output is calculated, the next step is to find the loss. After the loss is found the backpropagation is done and the weights of the model are updated. It is recommended to run the model with large amounts of data over long periods, to achieve accurate results. Figure 3(a) shows the basic architecture of a Sequence to Sequence model.

On the contrary, GANs employ deep learning models like the Convolutional Neural Network (CNN) [15]. A major difference between Seq2Seq and GANs is that GANs are unsupervised and can automatically detect patterns and irregularities in the input data. They are being used in many text generation tasks and are very accurate since they can be used to train models to generate text that is very similar to the text generated by humans. GANs have two major components, one is a generator and the other is a discriminator. The generator is trained to produce brand-new data from the sample domain. It takes random vectors as its input. After training the points in a vector space they will map to the points in the domain space [15]. At the same time, a discriminator (critic) is trained to distinguish between the data generated by the generator and the real-world data. It is a simple classification model that assigns the given input binary input label, i.e. 0 or 1. Now after this the output of the generator and some real-world data from the problem domain are sent to the discriminator as input. Now the discriminator decides if the given input is given by the generator or is real-world data. Reinforcement Learning (RL) [16] [17] is used as a reward system for GANs. RL rewards wanted outputs and reprimand the unwanted ones [18]. Based on the reward the models are updated and become more accurate. The generator is eventually trained to fool the discriminator, hence being able to generate text similar to humans [19]. Figure 3(b) shows the basic architecture of a GANs model. The discriminator receives the text and determines whether it is sent by the generator or if it is real text. The reward function sees the loss/error and generates a reward signal accordingly. The discriminator is deleted at the end, as after training as it is not needed post-training. The model is evaluated and carefully monitored. Based on the evaluation, updates are made to the system to achieve better results.

Table 1: Studies on Sequence-to-Sequence Models Reviewed

Article	Generation Model	Data Set	Performance
Jiawen et al, 2021 [11]	1) ESN MODEL 2)'DA-PN' MODEL 3)'DA-PN + COVER' MODEL 4)'DA-PN + COVER + MLO' MODEL	1) LCSTS 2) TTnews	ROUGE = 32.76
Habib et al, 2021 [20]	1)LSTM 2)BiLSTM 3)CONV1D 4)LSTM-CONV1D	1)Altibbi Dataset	Accuracy = 70.2% Accuracy = 73.8% Accuracy = 78.0% Accuracy = 68.3%
Haung et al, 2021 [21]	1)Vector Regression Model	1) CH818 dataset	BLEU (Pre-trained) = 0.51 BLEU (Fine Tuned) = 0.12
Yang et al, 2020 [22]	1)Sequence-to-Sequence models 2)GANs	1) THE CNN/DAILY MAIL DATASET	ROUGE-L = 39.14
Yu et al, 2020 [23]	1) Unified Generative Adversarial Networks (UGAN)	1) YELP dataset 2)AMAZON dataset 3) CAPTION dataset	Accuracy (YELP) = 88.2% Accuracy (AMAZON) = 74.6% Accuracy (CAPTION) = 85.4%
Zhao et al, 2020 [24]	1)Encoder-Decoder Architecture	1) CSL 2)RWTH-PHOENIX-Weather 2014T	BLEU-4 = 0.769 ROUGE = 0.901
Huang et al, 2020 [25]	1)Sequence-to-Sequence model 2)PTGEN	1) THE CNN/DAILY dataset	ROUGE-L = 30.0
Tomer et al, 2019 [26]	1) Sequence-to-Sequence model 2) FLSTM	1)CNN/Daily Mail dataset 2)DUC 2004 dataset	ROUGE-L (LCS) = 0.464
Zeng et al, 2019 [27]	1)Sequence-to-Sequence model 2)A keyword controlled network (KCN)	1) MSCOCO dataset 2)PARANMT50N 3)Quora	BLEU = 48.5 ROUGE = 39.1

Table 2: Studies on GANs Models Reviewed

Article	Generation Model	Data Set	Performance
Kim et al, 2020 [17]	1)RL 2)GANs	1)COCO Image Caption dataset 2)Stanford Natural Language Inference 3)EMNLP 2017 WMT News	B-BLEU-2(COCO) = 0.806 B-BLEU-2(SNLI) = 0.756 B-BLEU-2(COCO) = 0.817
Yang et al, 202 [16]	1) Feature-Guiding Generative Adversarial Networks (FGGAN)	1)COCO dataset 2) Chinese poetry dataset.	BLEU (COCO) = 0.773 BLEU(Chinese poetry)= 0.764
Yang et al, 2020 [22]	1)Sequence-to-Sequence models 2)GANs	1) THE CNN/DAILY MAIL DATASET	ROUGE-L = 39.14
Yu et al, 2020 [23]	1) Unified Generative Adversarial Networks (UGAN)	1) YELP dataset 2)AMAZON dataset 3) CAPTION dataset	Accuracy (YELP) = 88.2% Accuracy (AMAZON) = 74.6% Accuracy (CAPTION) = 85.4%
Wu et al, 2020 [28]	1) GANs	1) MS COCO Dataset	Embedding Similarity=-0.0157
Guan, 2019 [29]	1) Medical Text Generative Adversarial Network (mtGANs)	1) EMR text dataset	Accuracy = 76.35
Shi et al, 2018 [30]	1) Inverse Reinforcement Learning 2) GANs	1)The synthetic oracle dataset 2)COCO image caption dataset 3)IMDB movie review dataset	BLEU-2(COCO) = 0.868 BLEU-2(IMDB) = 0.755
Li et al, 2018 [31]	1) Category sentence generative adversarial network (CS-GAN)	1)Amazon Review Dataset 2)Yelp review dataset 3) Stanford sentiment tree bank 4)NYTimes, NEWS dataset 5)USAToday 6)Emotion dataset	Accuracy(Amazon Review) = 89.67% Accuracy(Emotional) = 39.32% Accuracy(NY Times) = 72.31%

Table 3: Studies on some miscellaneous text generation Models Reviewed

Article	Generation Model	Data Set	Performance
Bayer et al, 2022 [32]	1) Generative Pre-trained Transformer -2 model	1) Stanford Sentiment Treebank datasets (11 datasets in total)	Accuracy = 93.85% F1-score = 84.73%
Chen et al, 2021 [33]	1)Control-and-Edit Transformer framework 2)BERT	1) A corpus of movie plots from Wikipedia	Goal Achievement Rate=94.2% Perplexity = 5.8
Su et al, 2021 [34]	1)Stylistic response generator model	1) Gender-Specific Dialogue Dataset 2) Emotion-Specific Dialogue Dataset 3) Sentiment-Specific Dialogue Dataset	BLEU% (Gender) = 2.58 BLEU% (Emotion) = 1.88 BLEU% (Sentiment) = 1.72
Cao et al, 2020 [35]	1) NLDT	1)WEATHERGOV 2)WIKIBIO 3)WIKITABLE 4) WIKIBIOCN	BLEU = 62.89 BLEU = 45.77 BLEU = 38.71 BLEU = 38.87
Gumaste et al, 2020 [36]	1) POS Tagging 2)Name Entity Recognition	No datasets mentioned	Accuracy = 70.46%

3. TEXT GENERATION MODELS EVALUATION

In this survey, we first describe the various studies published using Seq2Seq model-based text generators and then detail those who employed GANs for the said task. Along with presenting various improvements and additions in the text generation models, it is important to document different algorithm performance evaluation metrics.
First, we calculate the precision score and recall for textual data.

- Precision takes the intersection between the target and generated sentences and calculates its size and divides it by the size of the generated sentence. In simple words, we can say that it takes the number of words that are present in both the sentences and divides it by the number of words in the generated sentence. Below is the formula for calculating the precision score (P) [37]:

$$P = \frac{|\ target\ sentence\ \cap generated\ sentence\ |}{|\ generated\ sentence\ |}$$

- Recall, on the other hand, divides the length of the target sentence while keeping the numerator the same as precision. Below is the formula for calculating the recall score (R):

$$R = \frac{|\ target\ sentence\ \cap generated\ sentence\ |}{|\ target\ sentence\ |}$$

The most popular measures to evaluate the text generation models are the Bilingual Evaluation Understudy (BLEU) score and the Recall Oriented Understudy for Gisting Evaluation (ROUGE) score [38].

- BLEU score calculation is word-position independent. Below is the formula for calculating BLEU [37]:

$$BLEU\ score = \min\left(1 - \frac{r}{c}\right) + \sum_{1}^{n} \frac{\log p_n}{n}$$

Here r is the length of the reference of the sentence, c is the generated sentence's length, and p is the precision.
Here r is the length of the reference of the sentence, c is the generated sentence's length, and p is the precision.

- This can be achieved by considering n consecutive words as a single unit rather than taking a single word as a unit. Another popular version uses the longest common subsequence. ROUGE can be calculated by the following formula [39]:

$$ROUGE = \frac{|similarity\ between\ target\ and\ generated\ sentences|}{|\ target\ sentence\ |}$$

- METEOR, apart from these is also widely used to score the generation models. METEOR calculates the precision and recall the same way as in BLEU and

Table 4: Text Generation Models

Sequence-to-Sequence	Generative Adversarial Networks
Hierarchical Human-like deep learning model	ConcreteGAN
Keyword Controlled Network (KCN)	Feature-Guiding GANs (FGGANs)
Enhanced Semantic Network (ESN)	Inverse Reinforcement Learning-based GANs
DA-PN (attention distribution with pointer network)	Category Sentence GANs (CS-GAN)
Unified GANs (UGAN)	Truth-Guided SeqGAN (TGSeqGAN)
Fuzzy Logic with LSTM (FLSTM)	Medical Text Generative Adversarial Network (mtGAN)
1D CNN with LSTM (LSTM-CONV1D)	Unified Generative Adversarial Networks (UGAN)
	DD-GANs (2 discriminators)

ROUGE. Then uses the following formulae to get the score [40]

$$F_{mean} = \frac{10PR}{R + 9P}$$

$$p = 0.5\left(\frac{c}{|generated\ sentence|}\right)^3$$

$$METEOR\ score = F_{mean}(1 - p)$$

Here P is precision, R is the recall, and c is the number of all common subsequences in both sentences. However, these models do lack correlation with references.

- Human evaluation is known to be the most accurate way to evaluate text generated by the model. But once again this will also cost a lot of time and money to accomplish [41]. It requires hiring capable staff to evaluate the text manually.

4. PROPOSED TEXT GENERATION MODELS

In this section, we discuss some of the techniques or variants of Sequence-to-Sequence models, Generative Adversarial Networks, and some other text generation methods which will be briefly discussed in the following pages.

4.1. SEQUENCE-TO- SEQUENCE MODEL BASED TEXT GENERATORS

Zeng et al [27] performs paraphrasing of sentences. The model is called Keyword Controlled Network (KCN). The model focuses on paraphrasing sentences using keywords. The given models will generate different paraphrases for a sentence given different keywords. Two separate encoders transform both the input sentence and keywords into vectors. These vectors are then combined together and passed by the decoder. The decoder has an attention-based mechanism. The decoder will copy words from keywords or generate new words from its vocabulary. The datasets used in the research are PARANMT50N, Quora, and MS COCO datasets. The MSCOCO dataset is used in training but has source and target sentences only hence the keywords had to be extracted. The keyword extraction was done via the use of the Positional Rank and Random Sample. BLEU, METEOR, and TER are used to perform automated evaluations (an absolute improvement of 0.06 is reported for the BLEU score). All the evaluation matrices have shown considerable improvement over the previous models. The human evaluation shows that the model also meets the user's expectations..

Fuzzy logic rules have been used by Tomer et al [26] in order to generate summaries of some given text. The researcher adds the fuzzy logic rules to the bidirectional LSTM to create the model FLSTM. The model has self-attention and an adam optimizer. There are four phases of the execution of this model. Firstly, all of the stop words (words that do not hold any have meaning i.e. the, is, are, etc.) are removed from the source text, and then the stemming is applied to the text (all the variants of the root word are converted to the root word i.e. words like runs, ran, running etc. are turned into run). Secondly, feature extraction is performed on the preprocessed text. Nine features are extracted (Sentence Position, Bigrams, Trigrams, TF-IDF, Cosine Similarity, Thematic number, Sentence length, Proper Noun Score, and Numeric Token). After this in the third phase, it uses fuzzy logic to find the most relevant sentences in the given text. Lastly, the relevant sentences are sent to the Bi-directional LSTM to generate a summary. The datasets used in this research are CNN/Daily Mail, DUC 2003-2004, and DUC 2006-2007datasets. The DUC 2003-2004 and DUC 2006-2007 datasets were merged into a single dataset with the aim to get better results. Rouge Evaluation was used to evaluate the generated summaries. FLSTM has shown improvement over the state-of-the-art model.

In another research Huang et al [25] also use a Sequence-to-Sequence to summarize legal public opinion news that can help the reader to get the main ideas of the news. The proposed model counters two tasks, one is the selection of the right domain and the second is to integrate the knowledge into the summaries. To counter these problems, pre-training topic information is selected from the legal news domain, which is later integrated into the model. The given model uses a bidirectional LSTM with self-attention. First, the topic words are encoded as they represent the main aspects of the documents. These encodings help the decoding process. By backpropagation, the output will be made with the same topic probability distribution as the source data. The used dataset is THE CNN/DAILY News corpus. Using the Rouge evaluation metric the results have been shown to outperform existing baseline models.

Apart from textual input, text generation tasks can take video inputs as well instead of just words. Zhao et al [24] aimed to generate real text output given the sign language videos. The introduced model has three modules, word existence verification, sentence generation, and cross-model re-ranking. Word existence verification checks whether the word exists in the vocabulary via a series of binary classifications. Convolutional Neural Network (CNN) is used in this phase. CNN alongside an encoder is used to get the video features. Logistic regression is learned for all the words in the video to confirm their presence. Sentence generation gathers all the words found to create multiple sentences via a pre-trained text generator. Cross-Model re-ranking selects one sentence that is semantically correct and most similar to the input video. The datasets used in this research are CSL and RWTH-PHOENIX-Weather 2014T. The amount of data available for this study is considerably small and hence makes it hard for the model to be trained accurately. Multiple evaluation matrices were used for this research including METEOR, BLEU, and ROUGE and all of them showed that the given SLT model is better than the previous works done.

Haung et al [21] has worked on generating lyrics. The proposed system is a Chinese lyrics generator. They report that the present lyrics generation system does not give a worthy output. The result lyrics do not have any relation with the music and the melody emotions. In the initial phase, a regression

model is trained that detects the melody emotions. In the second phase, a sequence-to-sequence model is used. This seq2seq model takes the notes and melody as inputs and generates lyrics. Lastly, the detected emotions and new lyrics are evaluated. The CH818 dataset is used is this research, it has 818 Chinese pop songs released between 1987 to 2010. The final results show that the lyrics generated in the end are fluent and the sentences are connected to each other.

Another research by Habib et al [20] works on the procedure of writing medical recommendations in Arabic. Different deep learning models are used and enhanced to predict the next word in the context. Unidirectional and bidirectional LSTM, one-dimensional convolutional neural network (CONV1D), and a pairing of both these models (LSTM-CONV1D) were implemented in this research. This model was created by using Altibbi databases which contain medical recommendations for gynecology, dermatology, psychiatric diseases, urology, and internist diseases. Two versions of this dataset were passed through this model. A 3-gram and 4-gram version of the dataset is used to train the final model. As this model predicted the next word therefore the training accuracy was used as a measure for evaluation. Out of all the four used models CONV1D performed the best for 4 datasets including gynecology, dermatology, psychiatric diseases, and internist diseases, for both version of the dataset. While unidirectional LSTM was the best for urology. 4-gram data gave slightly better results as compared to 3-gram data.

While JIANG et al [11] propose four Sequence-to-Sequence models for text summarization tasks by capturing key features of long documents. Supervised machine learning previously was widely used for this task but since it depended heavily on the quality of the features of the text so they did not end up being very accurate and the performance and speed were also not satisfactory according to the current needs. The proposed model uses a bidirectional LSTM with self-attention, which enhances so that the generated text and source text will be similar while handling any words that are being encountered for the first time by the model (out of vocabulary words). Enhanced Semantic Network (ESN) model focuses on adding semantic similarity into the loss function so that the correlation between source and generated text is very strong, DA-PN (attention distribution with pointer network) model aims at solving the issue of encountering vocabulary words, DA-PN + COVER model adds multi-attention to the DA-PN model, DA-PN + COVER + MLO is used to prevent the errors in the outputs by adding a mixed learning objective (MLO) to the previous model. The datasets used are the LCSTS dataset and the TTnews are used in the research and the ROUGE evaluation metric has been used for evaluation. The achieved results show that the proposed approach gives better performance as compared to the baselines and few state-of-the-art models.

4.2. GAN BASED TEXT GENERATORS

Generative Adversarial Networks have been a big step forward in the field of deep learning and opened doors for better generative models. NLP has also made use of these models to create text. It is based on a generator and a discriminator/critic, the generator produces text and the discriminator/critic determines whether the given text is generated by the generator or a human while the generator tries to fool the discriminator/ critic by generating human-like text.

Li et al [31] introduced a Category Sentence Generative Adversarial Network (CS-GAN). The model uses RL, RNNs, and GAN to generate categorical sentences that expand the dataset and improve its training during supervised learning. Based on the present state the generator will choose the next token and there is no reward straight away after this action. A rollout technique is used and the sentences are fed into the descriptors so that the maximum possible reward signal could be gained. The actions will cause the generation of category sentences and also improved the supervised learning by increasing the datasets as well. The used datasets include Amazon Review Dataset, Yelp review dataset, and Stanford sentiment treebank dataset. Results were generated for the generated text by performing sentiment analysis on them. CS-GAN produced an accuracy of 83%.

Shi et al [30] propose an IRL (Inverse Reinforcement Learning) framework. This model is based on GANs and learns a reward function (based on IRL) and an optimal policy to have maximum reward during training. While the reward function helps produce a reward signal, the policy encourages the generator to produce better text. Both of them are updated alternatively. The words are predicted one by one (the next word will be based on the previous words generated) using LSTM-based neural networks. The datasets used are the synthetic oracle dataset, COCO image caption dataset, and IMDB movie review dataset. The majority of evaluation matrices have shown that the IRL-based GANs model produces results closer to the ground truth, hence has been better.

In the field of medicine Guan et al [29] has developed a model called Medical Text Generative Adversarial Network (mtGAN). It uses RNNs, Maximum Likelihood Estimate (MLE), RL (Reinforcement Learning), and a bidirectional LSTM to generate the model. The basic GANs system is used for this task with a generator and a discriminator architecture. Disease tags are taken as input and output the Electronic Medical Records (EM) in text form for the disease. The disease tag can also be referred to as the keywords. The model makes sure that the resultant text is by these tags. It can avoid the leaking of patients' information and still produce good data that people can understand. A Chinese EMR text dataset is used for training the model. Evaluation of the model has shown that it can output realistic and diverse EMR text samples better than the baseline MLE and SeqGans.

A text generation model based on Truth-Guided SeqGAN has been worked on by Wu et al [28]. He lays emphasis on the loss function and has added the truth-guided method (based on Reinforcement Learning) so that the generated text would have a strong correlation with the real text, and the discriminator has a one-directional recurrent neural network layer with self-attention applied to it so that the final result may be semantically accurate. CNN and attention extract features of the text. This can cause semantic loss and many

mistakes in the attention mechanism, so therefore both of them have to work in parallel. The Dataset used was the MS COCO dataset TG-SeqGANs model has an improved speed for generating text as compared to the regular SeqGANs and the text quality has improved after being stabilized on NLL-test loss and Embedding Similarity, as well.

In another study, YANG et al [16] proposes a text generation model named Feature-Guiding GANs (FGGANs) to deal with the weak feedback (Reward Function) from the discriminator, as the reward signal is scalar and guidance is weak. FGGAN extracts the features from the discriminator and transforms them into vectors for feature guidance and feeds them to the generator. Sampling is needed before the features are passed to the generator but since sampling is random so it may cause the text to end up being of poor quality, now this is countered by introducing text semantic rules that remove the unreasonable entries in the generated text. The COCO Image Captioning dataset and Chinese poetry dataset have been used and using the BLEU score as an evaluation method the results show this model as better than the previous works done.

In [17] Kim et al present GAN-based architecture known as ConcreteGAN to generate text in both continuous and discrete space. The proposed Generative Adversarial Networks (GANs) architecture has four important components, RNN-based autoencoder, Code-generator, Code- discriminator, and Text-discriminator. The autoencoder is constructed via RNN and Gated Recurrent Unit (GRU), trained using the cross-entropy loss function. The code-generator and code-discriminator are trained for text in continuous space (encoded text) and finally the code-generator and text-discriminator are trained for text in discrete space (real-world textual data). Both the encoder transforms the input noises into an encoded version which is then decoded by their respective decoder. The decoded data is the generated text. The generated text is then sent to the discriminator which causes a reward signal to be generated and sent to the encoder. The datasets used here were the COCO Image Caption, Stanford Natural Language Inference (SNLI) corpus, and EMNLP 2017 WMT News datasets. The results were promising for SNLI and EMNLP datasets but COCO Image Captioning could not perform as well as the past models have done. Instead of using GRU, LSTM (Long Short Term Memory) could have been used as it keeps all the previous words in the context of the calculation of the next word and does not lose them as the words increase thus enabling the processing of larger data.

4.3. HYBIRD: SEQUENCE-TO- SEQUENCE + GENERATIVE ADVERSARIAL NETWORKS

Some studies have used both the Sequence-to-Sequence framework and GANs in their text generation models. These joint models have done very well in improving their accuracy and performance.

Earlier this year Yu et al [23] created a model named Unified Generative Adversarial Networks (UGAN). This model unifies both the sequence-to-sequence and GANs architectures. This research aims to perform TST with multidirectional transformation (train multiple attributes with one singular network). The given model takes a general GANs architecture and replaces its generator with a sequence-to-sequence model. Now, this model takes a sentence as an input and passes it to the sequence-to-sequence architecture. The sentence is initially embedded via an embedding layer. Then passed by the LSTM layer. The values from the final hidden state are converted back to words from embedding. The generated output is then sent to the discriminator which decides whether the received text is typed by a human or not. There are 2 LSTM layers and an embedding layer in the neural networks. It also uses Ranking and Attributes Classification in order to calculate the quality of the language and the accuracy of the style. YELP, AMAZON, and CAPTION datasets were used to train the model. The model used BLEU and Human Evaluation to test the model. The final model showed better performance as compared to the previous baseline models and was about 13% faster in terms of training time.

Yang et al [22] in another study performs text summarization using a Hierarchical Human-like deep learning model. It is inspired by how humans read and summarize some given text. It is divided into 3 components i.e. a knowledge-aware hierarchical attention module, a multi-task learning module, and a dual discriminator generative adversarial network (DD-GANs). Knowledge-aware hierarchical attention module, has an LSTM layer and first of all will decide the essential features of the input document and produce a knowledge-aware document. The multi-task learning module then uses a sequence-to-sequence framework to perform text categorization and syntax annotation (express the important components of the sentence). Then thirdly, DD-GANs improve the final results of the model. DD-GANs have one generator and two discriminators. The binary discriminator differs between human-generated and generator-based text. The ranking calculates the similarity between the source and generated text and uses it to generate an optimal reward. The dataset used is the CNN/DAILY MAIL and Gigaword dataset. The Rouge library was used to evaluate the model and consistently has produced better results than the compared models.

4.4. MISCELLANEOUS

Apart from them Gumaste et al [36] also creates a question generation system that creates questions from a given paragraph, passed as an input to the model. Parts of Speech (POS) tagging is very significant as it helps to give a proper tag to the text. NLP libraries like Spacy and NLTK are also used. The input text is broken down into tokens via Named Entity Recognition (NER) and these tokens are then analyzed to produce text. Both semantically and syntactically the sentences are examined. Syntactic analysis is done by POS tagging while semantic analysis is done through NER. The results show 70.46% accuracy for this model.

In another research Cao et al [35] generates short text that explains a given table. A neural generative architecture known as NLDT was proposed by the researcher. A two-level neural model is used to fully describe the connection between the data given in the table. The phases of this model include

embedding, encoding, decoding, and word conversion. A table can be represented by word embedding, field embedding, and position embedding. They represent a value, similar content, and the position of the word and are presented as a tuple. Now the embedded tuple is passed by a two-layer LSTM-RNN encoder-decoder. Now in the final phase the out-of-vocabulary words are handled in this study by replacing them with common field words that can imitate the information at hand. Four dataset have ben used in this study i.e. WEATHERGOV, WIKIBIO, WIKITABLE and WIKIBIOCN. The final results show that the proposed model was better then the state of art model.

While on the other hand, CHEN et al [33] aim to generate a story plot with a Language Model (LM). LM has some limitations which cause the generated plot to not make sense or may lack coherence (sentences may lack logic and be inconsistent). A control-and-edit transformer technique has been presented which uses edit distance (Dynamic programming algorithm) for simulated learning of support deleting and inserting policy. A reward function is also trained for this model as well. The story's topic and the desired goal are given as an input to the model. BERT model is used to generate a story plot from the input. The delete policy read the plot sequence and based on it decides whether to delete or the word or not (binary). While inserting a placeholder is placed after deciding number of tokens needed for a slot and then the word is decided to replace the placeholders. A corpus of movie plots from Wikipedia is used as the dataset for this research. Both automatic and human evaluation has been performed on the generated story plot. The automatic evaluation shows 94% accuracy for the generated story plot with the goal story plot while human evaluation focuses on grammar, reasonable story, logic, unique plot if the story is interesting, quality and whether it is worth sharing. The given model manages to produce great results in all the given fields consistently. The previous model in one or two fields did show some better results but failed greatly in other fields thus none of the previous models were consistent enough.

Su et al [34] has created a dialog generation system that has two parts: an Information Retrieval system and a TST model. This system uses an Information Retrieval (IR) system that receives a query sentence as input and from its database outputs an appropriate reply sentence. Then a Text Style Transfer (TST) system is used that takes the output from the IR system and a target style as its input and generates a new sentence as an output using a prototype-to style model. In order to control the output using a style embedding. All the styles have a style embedding. The embedding is lastly changed into text. The output has the same meaning as input but with the wording that satisfies the target style. This task is accomplished with GPT as well. The Gender-Specific, Emotion-Specific, and Sentiment-Specific Dialogue Datasets are used for training. The given model has outdone many robust baseline models by a significant margin.

Generative Pre-trained Transformer (GPT) models have been very good at text generation tasks. Bayer et al [32] perform a data argumentation task using GPT, the aim here was to perform text transformation to create new patterns of the language that can be used as training data for machine learning tasks. The model handles both long and short texts, lays emphasis on quality preservation, and makes sure that the model has seen all the text patterns before. First of all specific class data is extracted from the training data and are given a prefix and suffix token. If the data is too large some numeric fields will be removed. Then the model is trained.

Sentence-BERT [42] is used to create embedding. If the data is far away from the correct data then they are deleted. If many values are to be deleted then it depends on how many

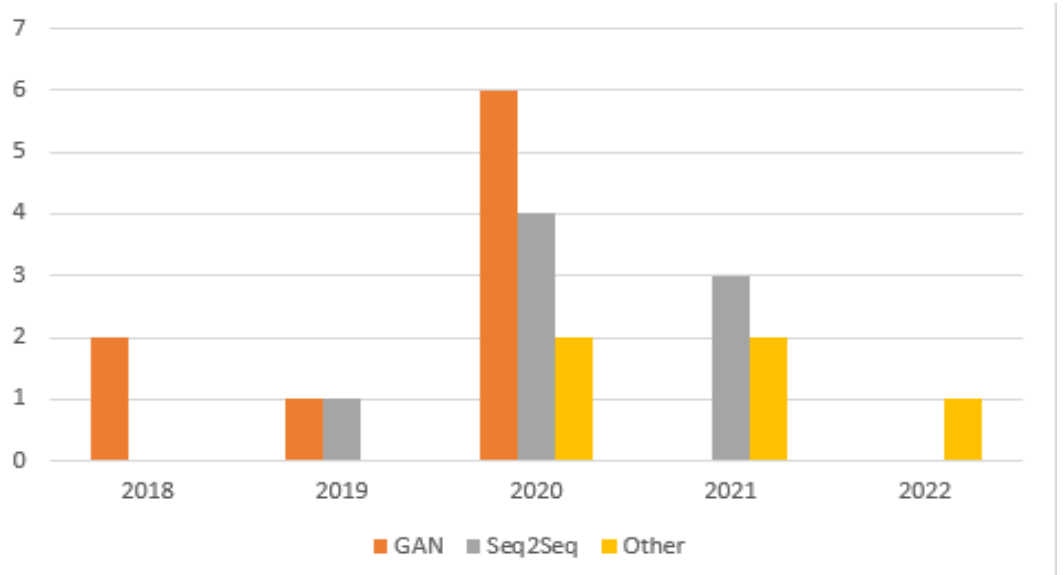

Figure 4:research models based on the year of their Publications

are deleted the threshold is changed to increase. And if the deletion is very less then the threshold decreases. Only the best generated data is given as output.11 different datasets were used for this task and the final model showed better accuracy than the previous models. GPT models are very good for NLP-based tasks and the newer version with more power and better accuracy is being developed therefore GPT has the potential of becoming a major part of NLP tasks in the near future.

Figure 4 shows research models based on the year of their Publications

5. EVALUATION MATRICES:

Text generation models have gained huge progress in the past few years. After the text generation models are created the next important step is to evaluate the model. If the given model is not evaluated there is no way of knowing if the model performs the way it is required to do so. Evaluation metrics need to be capable of consistently generating optimal results. Many different evaluation methods are being used for evaluating generated text. However, these evaluations have failed to have any correlation with human evaluations whatsoever.

Sellam et al [43] proposes a BLEURT evaluation matrix. It is trained to be similar to human judgments, with help of BERT (Bidirectional Encoder Representations from Transformers). The model is pre-trained over millions of examples for its generalization. The model is trained with many lexical and semantic supervision signal. The technique is tested with sentences that are large and diverse, with a wide range of words, sentence structures, and meanings (to cope with different variations of data). BLEURT learns to identify them effectively. The training is done in three steps, a regular pre-training of BERT, synthetic data's pre-training, and lastly fine-tune on task specific rating. It was shown great results against some recent state-of-the-art metrics.

Apart from this Khashabi et al [41] introduces the reader to GENIE. GENIE is a leaderboard that works with human evaluation of text generation. The text generations are posted on the leaderboard which automatically sends them to crowdsourcing (human evaluation) platforms. Correctness, conciseness, fluency, etc, are evaluated and compared to different automatic evaluations. There are four core challenges in text generation that GENIE represents, machine translation, summarization, commonsense reasoning, and comprehension. All of these are evaluated on some pre-defined templates by crowdsourcing. Evaluations on different datasets show that there are many evaluation metrics that produce results far from human evaluation. GENIE is publically available for the NLP community to use and hopes that it can contribute to improvements in text evaluation metrics in the foreseeable future.

In another study, Marchenko et al [44] introduces text generation and its evaluation methods. This study introduces coherence metrics for the evaluation of the generated text. Clustering is used to group the sentences that are nearest to each other. The distance between the sentences is calculated by the formula for relatedness. And calculates the coherence between the 2 sentences. The result should have similar sentences in the same cluster and with high coherence. The ROCStories corpus is used as a dataset for this study. A story generation model was trained on the basis of this evaluation metric. 50 fluent English speakers were asked to evaluate the stories generated based on Readability, Likability, and appropriateness. All the parameters were to be evaluated on a scale of 1-10. The correlation between the human evaluation and the proposed method was 0.86.

ROUGE has been a popular evaluation metric and Liu et al [45] conducted research to find whether ROUGE has a good correlation with human evaluation. Different domain-specific tasks and factors are studied to find what kind of impact they have on the correlation. The research was conducted on the ICSI meeting dataset. The experiment has shown that both human and ROUGE evaluation have a weak correlation but for some specific tasks, a better correlation can be achieved (especially for the summaries). In addition to this other Rouge metrics like ROUGE-2 and ROUGE-SU4 have better correlations as well. Disfluencies and stop words also have an impact on the results. Some other automatic evaluation metrics were also compared to ROUGE as well.

On the other hand, Zhao et al [46] present a research in which he presents the MoverScore metric. The aim of the research was to inspect approaches used in creating a metric with a high correlation with human evaluation. The metric was evaluated on several generation tasks, like summarization (TAC-2008 and TAC-2009 dataset), translation (CNN/DailyMail), image captioning (MS COCO Image Caption dataset), and data-to-text (BAGEL and SFHOTEL dataset) models. The research suggests that contextualized embedding representation and earth mover distance measures together present the best results. The metric has two variations the word mover distance and the sentence mover distance. Word mover distance is used with a semantic metric with unigram-based (read a single word at a time without the words before and after being in context) word embedding and is fine-tuned on the datasets. While sentence mover metric uses two-sentence embedding and uses their weighted sum. The results have shown a good positive correlation between the metric and human evaluation, where the word mover has a higher correlation than the sentence mover. This metric is also available online for public use.

Apart from this Goel et al [47] present a Robustness Gym (RG) for evaluation of a text. It unifies four standard evaluation patterns i.e. subpopulations, transformations, evaluation sets, and adversarial attacks. It gives three steps for the performing evaluation. First is contemplating, what evaluation to use, with help of key decision variables. Secondly creating slices of data using RG. And lastly, consolidate (combine) the findings and shared test benches and report in RG. RG makes it simple to evaluate models. The quality of the RG toolkit was tested on 7 state-of-the-art models on the CNN/DailyMail dataset.

Furthermore in another work Yang et al [48] build a Dual-based Translation Evaluation (DTE) metrics. The main

aim of this study was to evaluate the Ancient-Modern Chinese translations. Initially, the model was used to translate inputs in Ancient Chinese into Modern Chinese outputs, and then use a symmetric model is used to do vice versa. Then the BLEU score is calculated between the source input Ancient Chinese and the output from the symmetric model. Now a formula DualBLEU is introduced which does the accuracy of the model. After this, it is checked if the generated sentence is clear and well formatted (fluent). For checking the fluency first we get the average embedding similarity. All the words are already converted into the embeddings during the training, now these word embeddings are used to find the average ending of each sentence. Now we used the average embeddings to calculate the cosine distance between both sentences (source and output from the symmetric model output sentences). Then average BLEU is calculated using bigrams (instead of using a single word as an attribute a combination of two words is used). The translated sentences and references are in the test set. Lastly, a z-score is used to standardize the results. Now to get the final result of the DTE score both accuracy and fluency scores (both are standardized) are added. The Ancient-Modern Chinese corpus was used in the research as a dataset. The experiments have shown that DTE is very close to human evaluation.

6. CONCLUSION

In this survey, NLP-based text generation and evaluation methods are reviewed. Studies presented on the respective topic are surveyed. Text generation models are divided into three categories i.e. Sequence-to-Sequence, GANs, and miscellaneous. GANs has been very popular and almost half of the studies presented a model based on them. According to the results communicated by the authors, the models that are created by using both GANs and Sequence-to-Sequence architectures tend to achieve great results. While for the aim of evaluation coherence metrics have reaped the most promising results. However, currently, the models being used are not fully capable of generating meta results for text generation. Recent text generation models are consistently being improved daily and have given great results. But these results have been far from perfect. The majority of the models have been just under 90% accurate. On the other hand, most of the papers claim the popular evaluation metrics are having no or very small correlation with human evaluation. The best way to evaluate text generation models is that it is evaluated by humans. And there seems to be no evaluation technique that can simulate human evaluation. Whereas human evaluation is extremely time-consuming and cost-effective. It is clear that evaluation metrics need an extremely large amount of time and works to make significant progress. The evaluation metrics for text generation are nowhere close to the ground truth either.

Sequence-to-Sequence generates meaningful text and GANs discriminator module has the great ability to improve the performance of the model. But both still have a huge room for improvement. Among the reviewed studies the hybrid models that use both Seq2Seq and GANs models produced the best outcomes. Combining both approaches has had a great effect on the overall results of the generated text. Yu et al [23] not only report better results but also show a decrease in the training time. While Yang et al [22] also reported a ROUGE-L score of 39.14 which is the best one among the reviewed studies. Both these models have combined both of the approaches and have among the best results.

In recent years NLP based text generation tasks have been a red hot topic. Concepts like LSTM and self-attention layers have also been doing well to push the accuracies of the models as well. Newer and better models are being created every day. Sequence-to-Sequence models are commonly used over the past years and then the introduction of GANs has created a huge improvement among them. Researchers have been introducing the idea of more and more robust models. Recently GPT-3 has also shown great potential in this field and could end up producing much better results. There is a huge room for improvement in this field and there is a lot of being done. Hence in the upcoming years, these models can be seen, being improved exponentially and producing more and more accurate results. A lot of improvement is needed in both fields. Nowadays newer researchers are surfacing that are producing more accurate results. In the next few years, we can expect to reach greater results in the field as it is a very hot topic with a lot of researchers working on it.

CREDIT AUTHOR STATEMENT

Philemon Philip: Conceptualization, Methodolo-gy, Writing- Original draft preparation, Visualization, Investigation., Validation., Writing- Reviewing and Editing. **Sidra Minhas:** Supervision.

COMPLIANCE WITH ETHICAL STANDARDS

It is declare that all authors don't have any conflict of interest. Furthermore, informed consent was obtained from all individual participants included in the study.

7. REFERENCES

[1] W. Xiao-jie, B. Zi-wei, L. Ke and Y. Cai-xia, "A Survey on Machine Reading Comprehension," *Journal of Beijing University of Posts and Telecommunications,* vol. 8, no. 21, pp. 55170-55180, 2020.

[2] T. J.Legg, "What is NLP and what is it used for?," Medical News Today, 27 May 2022. [Online]. Available: https://www.medicalnewstoday.com/articles/what-is-nlp-and-what-is-it-used-for.

[3] J. A. Gulla, "New Language Models in NorwAI," NorwAI, [Online]. Available: https://www.ntnu.edu/norwai/new-language-models-in-norwai.

[4] A. Sharma, "Top 10 Applications of Natural Language Processing (NLP)," Analytics Vidya, 8 July 2020. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/07/top-10-applications-of-natural-language-processing-nlp/.

[5] Harshith, "Text Preprocessing in Natural Language Processing," Towards Data Science, 21 November 2019. [Online]. Available: http//:towardsdatascience.com/text-preprocessing-in-natural-language-processing-using-python-6113ff5decd8.

[6] D. Nettleton, "Inverse Document Frequency," Science Direct, 2014. [Online]. Available: https://www.sciencedirect.com/topics/computer-sscience/inverse-document-frequency.

[7] "Feature Extraction Techniques – NLP," Geeks for Geeks, 3 June 2022. [Online]. Available: https://www.geeksforgeeks.org/feature-extraction-techniques-nlp/.

[8] V. Zhou, "A Simple Explanation of the Bag-of-Words Model," Towrds Data Science, 11 December 2019. [Online]. Available: https://towardsdatascience.com/a-simple-explanation-of-the-bag-of-words-model-b88fc4f4971.

[9] R. LAKSHMANAMOORTHY, "Beginners Guide To Text Generation With RNNs," Analytics India Magzine, 30 May 2021. [Online]. Available: https://analyticsindiamag.com/beginners-guide-to-text-generation-with-rnns/.

[10] B. Lutkevich, "Language Modeling," TechTarget, March 2020. [Online]. Available: www.techtarget.com/searchenterpriseai/definition/language-modeling.

[11] J. JIANG, H. ZHANG,C. DAI,Q. ZHAO, H. FENG, Z. JI, AND I. GANCHEV, "Enhancements of Attention-Based Bidirectional LSTM for Hybrid Automatic Text Summarization," *IEEE Access,* vol. 9, pp. 123660-123671, 2021.

[12] P. Dugar, "Attention — Seq2Seq Models," Towards Data Science, 13 July 2019. [Online]. Available: https://towardsdatascience.com/day-1-2-attention-seq2seq-models-65df3f49e263.

[13] P. Singh, "A Simple Introduction to Sequence to Sequence Models," Analytics Vidya, 31 August 2020. [Online]. Available: www.analyticsvidhya.com/blog/2020/08/a-simple-introduction-to-sequence-to-sequence-models/.

[14] J. Brownlee, "Softmax Activation Function with Python," Machine Learning Mastery, 19 October 2020. [Online]. Available: https://machinelearningmastery.com/softmax-activation-function-with-python/.

[15] J. Brownlee, "A Gentle Introduction to Generative Adversarial Networks," Machine Learning Mastery, 19 July 2019. [Online]. Available: https://machinelearningmastery.com/what-are-generative-adversarial-networks-gans/.

[16] Y. YANG , X. DAN , X. QIU , AND Z. GAO, "FGGAN: Feature-Guiding Generative Adversarial Networks for Text Generation," *IEEE Access,* vol. 8, pp. 105217-105225, 2020.

[17] Y. KIM ,S. WON ,S. YOON AND K. JUNG, "Collaborative Training of Gans in Continuous and Discrete Spaces for Text Generation," *IEEE Access,* vol. 8, pp. 226515-226523, 2020.

[18] J. M. Carew, "What is reinforcement learning?," Search Enterprise AI, March 2021. [Online]. Available: https://www.techtarget.com/searchenterpriseai/definition/reinforcement-learning.

[19] Di Jin,Zhijing Jin,Zhiting Hu,Olga Vechtomova and Rada Mihalcea, "Deep Learning for Text Style Transfer," *MIT Press Direct,* vol. 48, no. 1, p. 155–205, 2022.

[20] M. HABIB, M. FARIS, R. QADDOURA, A. ALOMAR AND H. FARIS, "A Predictive Text System for Medical Recommendations in Telemedicine: A Deep Learning Approach in the Arabic Context," *IEEE Access,* vol. 9, pp. 85690 - 85708, 2021.

[21] Y, HUANG AND K.YOU, "Automated Generation of Chinese Lyrics Based on Melody Emotions," *IEEE Access,* vol. 9, pp. 98060 - 98071, 2021.

[22] M. Yang, C. Li, Y. Shen, Q. Wu, Z. Zhao, and X. Chen, "Hierarchical Human-Like Deep Neural Networks for Abstractive Text Summarization," *IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS,* vol. 32, no. 6, pp. 2744 - 2757, 2020.

[23] W. YU, T. CHANG, X. GUO, X. WANG, B. LIU, AND Y. HE, "UGAN: Unified Generative Adversarial Networks," *IEEE Access,* vol. 8, pp. 55170-55180, 2020.

[24] Jian Zhao, Weizhen Qi*, Wengang Zhou, Nan Duan, Ming Zhou, and Houqiang Li, Senior Member, IEEE, "Conditional Sentence Generation and Cross-modal Reranking for Sign Language Translation," *IEEE Transactions on Multimedia,* vol. 24, pp. 2662 - 2672, 2020.

[25] Y. Huang, Z. Yu, J. Guo,Z. Yu, and Y. Xian, "Legal public opinion news abstractive summarization by incorporating topic information," *International Journal of Machine Learning and Cybernetics,* vol. 11, p. 2039–2050, 2020.

[26] M. Tomer, M. Kumar, "Improving Text Summarization using Ensembled Approach based on Fuzzy with LSTM," *Arabian Journal for Science and Engineering,* vol. 45, p. 10743 – 10754, 2019.

[27] D. ZENG,H. ZHANG, L. XIANG, J. WANG, AND G. JI, "User-Oriented Paraphrase Generation With Keywords Controlled Network," *IEEE Access,* vol. 7, pp. 80542-80551, 2019.

[28] Y. WU AND J. WANG, "Text Generation Service Model Based on Truth-Guided SeqGAN," *IEEE Access,* vol. 8, pp. 11880-11886, 2020.

[29] J. Guan, R. Li, S. Yu and Y. Zhang, "A Method for Generating Synthetic Electronic Medical Record Text," *IEEE/ACM Transactions on Computational Biology and Bioinformatics,* vol. 18, no. 1, pp. 173-182, 2019.

[30] Z. Shi, "Towards Diverse Text Generation with Inverse Reinforcement Learning," *arXiv,* 2018.

[31] Y .Li, Q. Pan, S. Wang, T. Yang and E. Cambria, "A Generative Model for Category Text Generation," *Information Sciences,* vol. 450, pp. 301-315, 2018.

[32] M. Bayer, M. Kaufhold, B. Buchhold, M. Keller, J. Dallmeyer and C. Reuter , "Data augmentation in natural language processing: a novel text generation approach for long and short text classifers," *International Journal of Machine Learning and Cybernetics,* 2022.

[33] J. CHEN, G. XIAO , X. HAN, AND H. CHEN, "Controllable and Editable Neural Story Plot Generation via Control-and-Edit Transformer," *IEEE Access,* vol. 9, pp. 96692-96699, 2021.

[34] Y. Su , Y. Wang , D. Cai , S. Baker, A. Korhonen, and N. Collier, "PROTOTYPE-TO-STYLE: Dialogue Generation With Style-Aware Editing on Retrieval Memory," *IEEE/ACM Transactions on Audio, Speech, and Language Processing,* vol. 29, pp. 2152-2161, 2021.

[35] J. CAO, "Generating Natural Language Descriptions From Tables," *IEEE Access,* vol. 8, pp. 46206 - 46216, 2020.

[36] P. Gumaste, S. Joshi, S. Khadpekar and S. Mali, "Automated Question Generator System Using NLP Libraries," *IRJET,* vol. 7, no. 2, pp. 4568-4572, 2020.

[37] K. Doshi, "Foundations of NLP Explained — Bleu Score and WER Metrics," Toward Data Science, 9 May 2021. [Online].

Available: https://towardsdatascience.com/foundations-of-nlp-explained-bleu-score-and-wer-metrics-1a5ba06d812b.

[38] Z. Fu, X. Tan,N. Peng,D. Zhao and R. Yan, "Style Transfer in Text: Exploration and Evaluation," in *Vol. 32 No. 1 (2018): Thirty-Second AAAI Conference on Artificial Intelligence*, Southern California, 2018.

[39] "An intro to ROUGE, and how to use it to evaluate summaries," Free Code Camp, 26 January 2017. [Online]. Available:https://www.freecodecamp.org/news/what-is-rouge-and-how-it-works-for-evaluation-of-summaries-e059fb8ac840/.

[40] "METEOR metric for machine translation," Machine Learning Interviews, 2 November 2021. [Online]. Available: https://machinelearninginterview.com/topics/machine-learning/meteor-for-machine-translation/.

[41] D. Khashabi, G. Stanovsky, J. Bragg, N. Lourie, J. Kasai, Y. Choi, N. A. Smith, and D.S. Weld, "GENIE: A Leaderboard for Human-in-the-Loop Evaluation of Text Generation," *arXiv,* 2021.

[42] N. Reimers, I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," *arXiv,* 2019.

[43] T. Sellam, D. Das, A.P. Parikh, "BLEURT: Learning Robust Metrics for Text Generation," *arXiv,* 2020.

[44] O. O. Marchenko, S. Radyvonenko, T. S. Ignatova,P. V. Titarchuk,and D. V. Zhelezniakov, "IMPROVING TEXT GENERATION THROUGH INTRODUCING COHERENCE METRICS," *Cybernetics and Systems Analysis,* vol. 56, pp. 13-21, 2020.

[45] F. Liu, and Y. Liu, "Exploring Correlation Between ROUGE and Human Evaluation on Meeting Summaries," *IEEE Transactions on Audio, Speech, and Language Processing,* vol. 10, pp. 187-196, 2010.

[46] W. Zhao, M. Peyrard, F. Liu, Y. Gao, C.M. Meyer, and S. Eger, "MoverScore: Text Generation Evaluating with Contextualized Embeddings and Earth Mover Distance," *arXiv,* 2019.

[47] K. Goel, "Robustness Gym: Unifying the NLP Evaluation Landscape," *arXiv,* 2021.

[48] K. Yang,D. Liu,Q. Qu,Y. Sang and J. Lv, "An automatic evaluation metric for Ancient-Modern Chinese," *Neural Computing and Applications,* vol. 33, p. 3855–3867, 2020.