Crafting Urdu Ghazals: NLP Breakthroughs

Hassan Ashfaq
Faculty of Computer Science & Engg.
GIK Institute of Engg. Sciences & Tech.
Topi, Khyber Pakhtunkhwa, Pakistan.
u2021221@giki.edu.pk

Ali Hassan Khan
Faculty of Computer Science & Engg.
GIK Institute of Engg. Sciences & Tech.
Topi, Khyber Pakhtunkhwa, Pakistan.
u2021079@giki.edu.pk

Muhammad Haseeb Ishaq
Faculty of Computer Science & Engg.
GIK Institute of Engg. Sciences & Tech.
Topi, Khyber Pakhtunkhwa, Pakistan.
u2021389@giki.edu.pk

***Abstract*—Urdu poetry generation, a realm demanding linguistic finesse and creative ingenuity, is explored in this research, focusing on Urdu ghazals—a cornerstone of cultural heritage—using natural language processing (NLP) techniques, notably deep learning. Drawing from a dataset featuring 17,609 couplets authored by 15 eminent Urdu ghazal poets, four distinct methodologies were employed: (a) the n-gram probabilistic model, (b) LSTM and GRU deep learning models, (c) the state-of-the-art GPT-2 model, and (d) some novel techniques including BERT and GANs Evaluation through BLEU scores showcased GPT-2's supremacy with an unprecedented score of 1.74, followed by GRU (0.95), n-gram (0.7), LSTM (0.6), and BERT (1.1). Rhyme scores revealed GRU's excellence (0.5), trailed by GPT-2 (0.36), n-gram (0.35), LSTM (0.25), and BERT (0.1). Moreover, this study pioneers the integration of BERT for Urdu poetry generation, marking a significant stride in computational creativity. Notably, while the inclusion of GANs yielded discouraging results, this research underscores the novel exploration and potential future directions in the realm of Urdu poetry generation.**

***Index Terms*—urdu poetry generation, natural language processing, deep learning, transformers, computational linguistics**

I. Introduction

Poetry generation, a fascinating area within natural language processing (NLP), is a mixture of linguistic creativity with computational expertise that results in poetic couplets and verses. In this field, Urdu poetry shines brightly, especially in its famous form called ghazals. Ghazals have this special appeal with their intricate rhymes and diverse themes, which makes them ideal for computational exploration. In our study, we're diving into making Urdu ghazals leveraging NLP techniques ranging from traditional n-gram models to state-of-the-art deep learning architectures like LSTM and GRU, culminating in the adoption of transformer-based models such as GPT-2. We seek to not only push the the boundaries of natural language processing, but also pay homage to the rich traditional and cultural values of Urdu poetry, giving both experts and aspiring poets to engage with this revered art form in the digital age.

The exploration of automatic Urdu ghazal generation is not only a significant endeavor in computational linguistics but also plays a crucial role in cultural preservation. Urdu poetry, deeply rooted in tradition and literary heritage, is a cornerstone of South Asian culture. By delving into the modern generation of Urdu ghazals, we advance natural language processing and machine learning while preserving this invaluable cultural art form. This research not only showcases modern technology but also ensures the continuation of Urdu poetry for future generations. The analysis of Urdu poetry provides insights into the style, lexical diversity, and sentiment analysis of renowned poets like Iqbal and Ghalib, aiding research in computational stylistics. Moreover, this study opens avenues for broader engagement with Urdu literature, fostering cross-cultural dialogue and a deeper appreciation for the beauty and complexity of Urdu poetry globally.

In today's interconnected world, where technology plays a crucial role in shaping cultural interactions, the exploration of Urdu ghazal generation is highly relevant. By leveraging natural language processing techniques and deep learning models, we not only preserve the beauty of Urdu poetry but also democratize access to this rich cultural heritage. The computational Urdu ghazal generation fosters cross-cultural understanding and appreciation, transcending linguistic barriers and promoting diversity in artistic expression. This research not only showcases cutting-edge technologies but also underscores the enduring relevance and universality of Urdu poetry in our increasingly connected world.

A. Related Work

In the world of linguistic technology and cultural preservation, researchers have made big strides in the generation of Urdu poetry. Researchers have delved into various natural language processing (NLP) techniques, from basic n-gram models to advanced deep learning architectures like LSTM and GRU, as evidenced in studies by S. Fahim, I. Siddiqui, S. Pervez, S. Kumar, F. Alvi, and A. Samad [1]. The adoption of transformer-based models, exemplified by Fahim et al.'s work, has shown promising avenues for enhancing the computational generation of Urdu ghazals. More work by researchers like S. A. Mukhtar and P. S. Jogleka from Vishwakarma Institute of Technology [2] have developed systems aimed at assisting aspiring poets by providing machine-generated prompts to overcome writer's block. These systems offer a novel approach to inspire creativity and facilitate the poetic process. Moreover, recent studies, such as the work by J. F. Ruma, S. Akter, J. J. Laboni, and R. M. Rahman [3] have delved into the classification of Persian Hafez poetry based on the poet's era using deep learning models. These methods help capture the intricate rhyme patterns and deep themes found in Urdu poetry, pushing forward both language technology and cultural preservation. Researchers use measures like BLEU scores, rhyme consistency, and human feedback to check how good computer-generated poetry is. This research not only helps with Urdu literature but also fosters cross-cultural dialogue and enhancing global appreciation for the beauty and complexity of Urdu poetry.

B. Gap Analysis

In the realm of computational techniques for Urdu ghazal generation, there are several gaps that researchers are working to address. While studies have explored a range of natural language processing (NLP) approaches, from n-gram models to deep learning architectures, [4] there is a need for a comprehensive comparison of these methods to determine their effectiveness and efficiency in generating high-quality Urdu ghazals. Additionally, existing research tends to focus heavily on quantitative evaluation metrics like BLEU scores, overlooking qualitative aspects such as poetic coherence and thematic relevance, which are crucial for assessing the artistic merit of generated poetry. Furthermore, while some studies have ventured into poet attribution in Urdu ghazals using deep learning, [5] there is a need to further investigate and validate the accuracy and robustness of these attribution models. Moreover, limited exploration has been done on incorporating cultural and historical context into computational models for Urdu ghazal generation. This aspect could significantly enhance the authenticity and cultural significance of the generated poetry.

C. Problem Statement

In today's digital age, finding authentic Urdu poetry, especially in the beloved ghazal style, is becoming increasingly challenging. Despite the enduring love for literature among enthusiasts and a budding interest among the youth, accessing the profound works of past luminaries like Mirza Ghalib, Faiz Ahmad Faiz, Parveen Shakir, and Ahmad Faraz has become quite rare. This scarcity not only hampers our cultural appreciation but also leaves aspiring poets feeling stuck and uninspired. Writer's block and a lack of motivation further contribute to the dearth of new poetic creations. Therefore, there's an urgent need to fill this void, help poets overcome writer's block, [6] and ignite their creativity through innovative computational approaches in Urdu ghazal generation.

1) How can we use the latest advancements in computational linguistics to develop a fresh method for automatically generating Urdu ghazals that also acts as a source of inspiration for poets facing writer's block?
2) What are the essential linguistic and stylistic features that define authentic Urdu ghazals, and how can we effectively incorporate them into our computational models to provide meaningful prompts for poets in need of inspiration?
3) How do the generated ghazals compare to those authored by renowned Urdu poets in terms of thematic richness, emotional depth, and poetic mastery, as perceived by both aspiring poets experiencing writer's block and seasoned literary critics?

D. Novelty of our work

Our research aims to transform poet attribution in Urdu Ghazals by studying and applying different AI models. With a meticulously curated dataset of 17,609 couplets from 15 prominent Urdu poets across different eras, our goal is to accurately capture the distinctive writing styles of each poet. This enables us to effectively identify misattributions and instances of plagiarism while offering valuable insights into the rich tradition of Urdu literature. By carefully selecting poets from various historical periods and leveraging state-of-the-art computational techniques, our study makes a substantial contribution to computational poetry analysis, providing a refined framework for preserving and comprehending the essence of Urdu poetry. Furthermore, our exploration of different model architectures and methodologies offers valuable insights into the optimal strategies for sequence classification in Urdu poetry, paving the way for future advancements in computational literary analysis.

E. Our Solutions

In this study, we present a comprehensive investigation into the attribution of poets in Urdu Ghazals using advanced computational techniques. Our primary contribution lies in pioneering the application of machine learning, deep learning, and transformer-based models for accurately characterizing the unique writing styles Urdu. We meticulously train and evaluate various models, including N-Gram models, RNNs which include LSTM and GRUs, Transformer models such as GPT-2 and some novel approaches like BERT and GANs. Through our meticulous analysis, we not only achieve groundbreaking accuracy rates but also offer valuable insights into the optimal strategies for sequence classification in Urdu poetry.

TABLE I
POETS AND THEIR COUPLETS COUNT

No.	Poet/Shayar	Couplet Count
1	Ahmed Faraz	926
2	Zafar Iqbal	1104
3	Qateel Shifai	780
4	Parveen Shakir	593
5	Nida Fazli	474
6	Faiz Ahmad Faiz	504
7	Jaun Elia	1470
8	Muneer Niyazi	523
9	Allama Iqbal	797
10	Riyaz Khairabadi	1700
11	Haider Ali Atish	1330
12	Siraj Aurangabadi	860
13	Mir Taqi Mir	2971
14	Nazeer Akbar Abadi	1643
15	Mirza Ghalib	1934

II. LITERATURE REVIEW

In this literature review, we investigate the profound influence of deep learning on poetry generation. Additionally,

TABLE II

RESEARCH LITERATURE REVIEW TABLE: KEY PAPERS AND FINDINGS

Paper Title	Authors	Publication Date	Dataset	Research Aspects	Key Findings	Contributions	Limitations
Arabic poem generation with deep learning	Talafha et al.	2019	Khaleej-2004, Watan-2004, Arabic poetry dataset	Phonetic CNN subword embedding, keyword extraction, B/F-LM with GRU for first verse, HAS2S for subsequent verses	BLEU score: 0.6, outperforms RNN, GRU, LSTM models	Integrates deep learning into Arabic poetry, superior to traditional models	Generalizability to poetic styles, further evaluation on larger datasets needed
Binari: A poetry generation system for ghazals	Galip et al.	2020	Corpus of 9484 couplets from renowned Ottoman poets of the 16th century	Training on a corpus of 9484 couplets attributed to Ottoman poets Necati, Mihri Khatun, and Revani using RNN with GRU units	Results lacked meaningfulness and grammar	Introduces Binari, a Turkish ghazal generator inspired by Hafez	Results were not impressive in terms of meaningfulness and grammar
Hindi Poetry Generation using Neural Networks	Mukhtar et al.	2021	Data scraped from Rekhta.org	Character-level RNNs with LSTM Models trained to generate Misra, Sher, and Ghazals	Promising results: 14 out of 20 poems deemed acceptable in Urdu and Hindi	Addresses scarcity of literature on Urdu poetry generation	Limited evaluation on generated poetry quality
Bidirectional LSTM Networks for Poetry Generation in Hindi	Kumar et al.	2021	Not Specified	Experimented with four models: BLG, BLG-SA, BLCG, and BLCG-SA for Hindi poetry generation	Using RNNs and LSTMs with CNNs improves learning sequences with reduced training time	Advances text generation in Hindi using deep learning techniques	Specific to Hindi poetry generation, may lack generalizability
Arabic Poems Generation Using LSTM, Markov-LSTM and Pretrained GPT-2 Models	Hakami et al.	2021	Not specified	Comparison of LSTM, Markov-LSTM, and Pre-trained GPT-2 models for Arabic poetry generation	GPT-2 model outperformed LSTM and Markov-LSTM in generating fluent and relevant outputs	Highlights the potential of pretrained models for Arabic poetry generation	Limited to comparing three specific models
Automatic Arabic Poem Generation with GPT-2	Beheitt et al.	2022	Corpus of classical Arabic poems	Fine-tuning GPT-2 on a corpus of classical Arabic poems	Model capable of producing coherent and fluent poems, but limited creativity	Evaluates GPT-2's ability to generate Arabic poems	Creativity of generated poems is limited
A deep learning classification model for Persian hafez poetry based on the poet's era	Ruma et al.	2022	Dataset of 496 ghazals by Hafez labeled with chronological and Raad labels	Sequential learning to classify Persian ghazals by poet era	Deep learning classifiers outperformed SVM in terms of F1-score	LSTM model had highest precision and recall on Persian dataset	Deep learning classifiers may require more computational resources
Urduai: Writeprints for urdu authorship identification	R. Sarwar and S.-U. Hassan	2022	Dataset of 985 Urdu text samples from 90 authors	Utilized stylometric features including vocabulary richness, n-grams, and character n-grams	Outperformed previous methods with 94% accuracy for authorship attribution in Urdu	Improvement over decision trees, naive bayes, SVMs, and random forests	Limited to authorship attribution in Urdu

we provide a comprehensive analysis, including a detailed table, to elucidate the convergence of deep learning and natural language processing techniques.

Some of the key literature review publications discussed in the table include pioneering research by Sameerah Talafha and Banafsheh Rekabdar [7] who introduced deep learning techniques to Arabic poetry generation, achieving a BLEU score of 0.6 by combining phonetic CNN sub-word embedding, semantic keyword extraction, and models like B/F-LM and HAS2S. Subsequent research by Beheitt, Mohamed El Ghaly, and Moez Ben Haj Hmida [8] compared LSTM, Markov-LSTM, and GPT-2 models for Arabic poetry generation, with GPT-2 demonstrating superior fluency and relevance, highlighting the effectiveness of pre-trained models. Additionally, Asmaa Hakami, Raneem Alqarni, Mahila Almutairi, and Areej Alhothali (2021) [9] reinforced this notion by comparing LSTM, Markov-LSTM, and GPT-2 for Arabic poetry generation, with GPT-2 emerging as the top performer. Meanwhile, in the realm of Hindi poetry generation, Ankit Kumar [10] explored Bidirectional LSTM models, while another study experimented with various LSTM-based generators, incorporating self-attention and convolutional layers, resulting in improved learning of sequences and reduced training time.

III. METHODOLOGY

A. Dataset

The dataset we have used comprises of 17,609 couplets sourcing from 15 renowned Urdu poets. Each couplet in the dataset consists of two misras (lines). The primary source of this data is Rekhta, an Indian literary platform dedicated to fostering Urdu poetry within the subcontinent. We intend to use this dataset for both training and testing our models. The table I illustrates the distribution of poets within our dataset.

The selection of these poets was based on several things making sure it fit our criteria. The number one thing we aimed for was diversity of time period within the dataset, the poets [1-8] within our dataset were active and wrote during the 19th-20th century, whereas poets [9-15] are from 18th-19th century. This diversity provides the model with variety of linguistic styles, and dialects. The next most important things was sufficient amount of dataset, we prioritized poets who had written a substantial number of ghazals to ensure a larger dataset for training models. All poets in the dataset have contributed at least 450 couplets, with Mir Taqi Mir topping the list at 2,986 couplets.

The dataset was also used in some of the previous studies. [5] For those interested, the dataset was scraped from Rekhta, which is accessible through here.

B. Pre-Processing

In the data pre-processing phase, the Urdu couplets dataset undergoes several essential transformations to prepare it for deep learning tasks. Initially, sentence segmentation based on semicolons breaks down couplets into their constituent lines, facilitating analysis at the line level. Word tokenization then converts each line into tokens or words, enabling further processing at the word level. Text to sequence conversion follows, mapping these word tokens to numerical indices essential for numerical processing by machine learning models. Padding ensures uniform sequence lengths by either adding padding to shorter sequences or truncating longer ones. Finally, one-hot encoding transforms the numerical sequences into binary vectors, facilitating input to neural network models. These pre-processing steps collectively transform raw text data into a format suitable for natural language processing tasks, empowering subsequent modelling and analysis.

C. Techniques

This study comprises of applying a number of model and architectures to achieve the set target of generating couplets. Our approach started with probabilistic model with N-grams, we utilized the n-gram model to predict word sequences based on their frequency of occurrence in the text. Additionally we moved on to Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks which are types of recurrent neural network (RNN), and are useful in capturing long-range dependencies in the text. Furthermore, we leveraged transformer-based models like GPT-2 which excel at generating contextually relevant text. In addition to the above techniques mentioned, we utilized some novel approaches like BERT (Bidirectional Encoder Representations from Transformers) with its bidirectional understanding of text and adapted it for generating nuanced and context-aware couplets. Moreover, this model serves for inference in generating series of poetry and evaluate it using BLEURT algorithm. [11] We also conducted a study on Generative Adversarial Networks (GANs), and how can they be employed to create more creative and diverse couplets by adding a generator model against a discriminator to improve the generated outputs.

D. Probabilistic Model Approach: N-Grams

We started by employing a probabilistic model approach, in which we utilized an n-gram model with n set to 3 (Tri-gram). The n-gram model technique is a widely used technique in the domain of NLP utilized to calculate probability of a word that helps us forecast the next word in the sequence. We started by normalizing and standardizing the dataset by using a regular expression tokenizer (Regex Tokenizer) to slice the text into distinct tokens like words and phrases. Then we split the the tokenized data into training and testing (80% training and 20% testing). We calculated n-grams for the training set, where each n-gram represented a sequence of n-words from the test. Using the probabilities obtained from the n-grams, we constructed a probabilistic model that estimated the likelihood of each n-gram based on its occurrence frequency. Finally, the trained model was employed to generate couplets with each line representing a misra of the whole couplet. The models calculated probabilities resulted in somewhat coherent and meaningful couplets.

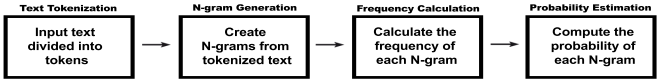

Text Tokenization
Input text divided into tokens
N-gram Generation
Create N-grams from tokenized text
Frequency Calculation
Calculate the frequency of each N-gram
Probability Estimation
Compute the probability of each N-gram

Fig. 1. Block Diagram for the N-GRAM Model

E. Recurrent Neural Networks

In this approach, we explored two deep learning architectures,Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU). Both of these models are types of recurrent neural networks (RNNs) and are widely used for language modeling. We standardized and tokenized the dataset for both RNNs, and then converted it into padded sequences and represented in as a vector to ensure that all sequences had the same length. As the couplets in the dataset had the length of about 10 words, we capped our sequence length at 10. The dataset was split into 80% training set and a 20% test set.

1) LSTM: We started with training the model with 100 units on the training data, starting with an embedding layer that transformed vectors into a continuous space. The final layer had softmax activation to select the word with the highest probability. The model was trained for 50 epochs using the Adam optimizer, with a learning rate of 0.001 and the hyper-parameters were chosen with trial and error.

2) GRU: The GRU model was trained on 120 units on the training data. The architecture mirrored the LSTM, beginning with an embedding layer followed by a softmax activation function for word selection. The model trained for 10 epochs with the Adam optimizer, also at a 0.001 learning rate.

Both the LSTM and GRU models take an initial sequence (seed) to produce a two-line Urdu couplet. The generation depends on the models' predicted probabilities, determining the subsequent words in the couplet. The process iteratively adds words based on these predictions until a complete couplet is generated.

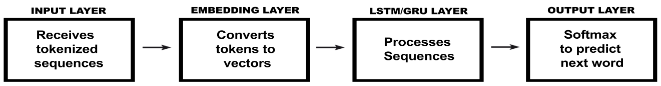

INPUT LAYER
Receives tokenized sequences
EMBEDDING LAYER
Converts tokens to vectors
LSTM/GRU LAYER
Processes Sequences
OUTPUT LAYER
Softmax to predict next word

Fig. 2. Block Diagram for RNNs

F. Transformers: GPT-2

We also utilized the cutting edge GPT-2 (Generative Pre-trained Transformer) model. Developed by Open-AI, GPT-2 is renowned for its adeptness at generating coherent and natural-sounding text. We started the training process by initializing the model with its corresponding tokenizer, which segmented the dataset into tokens, which were encoded in numerical representations. The dataset was normalized and the tokenizer encoded the dataset, ensuring the data was in a format conducive for GPT-2 training. Lastly, a data loader was created to feed the encoded data into the model during the training phase. This optimization enhanced the efficiency of data loading and batching. We fine-tuned the model on the dataset for 70 epochs. Each epoch involved a full iteration of the dataset, enabling the model to optimize performance iteratively. We used the the Adam optimizer with a low learning rate of 0.000002, dictating the phase of parameter updates.

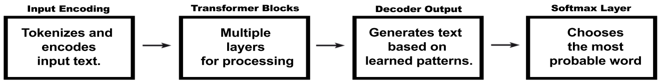

Input Encoding
Tokenizes and encodes input text.
Transformer Blocks
Multiple layers for processing
Decoder Output
Generates text based on learned patterns.
Softmax Layer
Chooses the most probable word

Fig. 3. Block Diagram for the GPT-2 Model

G. Novel Approaches: BERT & GAN

We used the BERT approach to generate context-aware Urdu couplets, leveraging its bidirectional text comprehension. The dataset was pre-processed using BERT's tokenizer and split into training (80%) and testing (20%) sets. A BERT- base model was fine-tuned on the dataset for 50 epochs with the Adam optimizer (learning rate of 0.00005) to adapt to the poetic nuances of Urdu poetry.he fine-tuned model was then used for generating poetry and evaluating the quality of generated couplets using the BLEURT algorithm, enabling the creation of contextually resonant Urdu couplets. In case of Generative Adversarial Networks (GANs) to generate innovative and Urdu couplets, we started by using a generator model against a discriminator, then the dataset was pre-processed by tokenizing and splitting it into training and testing sets. The generator, an LSTM based with 100 units, aimed to produce Urdu couplets, while the discriminator, an LSTM-based model with 80 units, sought to distinguish between real and generated couplets. The generator and discriminator were trained alternately for 100 epochs, with the generator attempting to produce couplets that could fool the discriminator, and the discriminator refining its ability to differentiate between real and fake couplets, showing improvement in the generated couplets.

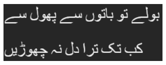
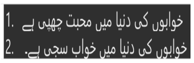
IV. Results

This section describes the results of our experiments. Since poetry is more about how something sounds than its literal meaning, we don't believe it's possible to judge a poem just by accuracy metrics. So we evaluate our models on two fronts: metrical evaluation and evaluation by GPT-4. In addition to scoring the models by their metrical accuracy, to evaluate the quality of models' couplets we used BLEU score and rhyme analysis. Meanwhile, the poetry was reviewed by GPT-4 who scored the products of the models based on some key criteria of Urdu poetry with comments on metrics (Beher), radeef (end rhyme), kafiya (rhyme pattern) and tashreeh (commentary). We chose three Urdu seed words so that the number of couplets produced from each model would remain similar and make it easier to compare models and evaluate them thoroughly. After this, we fed these couplets to all trained models and evaluated them on an objective metrical level and human liking.

A. Probabilistic Models RESULTS

Evaluating the trigram n-gram model for poetry generation suggests some of its weaknesses as well as its strengths. The BLEU Score of 0.75 suggests some overlap between the trigram n-gram model and the reference text, indicating some success in modelling parts of the desired poetry style. The Rhyme Score of 0.3, on the other hand, suggests dramatically lower scores in producing rhymes, an important hallmark of poetry. These results emphasise the inevitable trade-offs between linguistic fidelity and poetic creativity in machine poetry generation. The table III below shows the generated couplets along with the evaluation metrics.

TABLE III
EVALUATION RESULTS FOR N-GRAM MODEL

Seed Word	**Muhabbat (Urdu for Love)**	**Khuwab (Urdu for Dream)**
Couplet	 بولے تو باتوں سے پھول سے کب تک ترا دل نہ چھوڑیں	 1. خوابوں کی دنیا میں محبت چھپی ہے 2. خوابوں کی دنیا میں خواب سجی ہے.
BLEU Score	0.75	0.71
Rhyme Analysis	0.33	0.35
Expert Evaluation (Generative Pre-trained Transformer v4)	The couplets exhibit consistent meter and rhyme, evoking emotions of longing and hidden desires, adds depth to the theme of love and dreams.	

B. LSTM RESULTS

The LSTM model was trained for 50 epochs and performed exceptionally well on the training data, achieving a loss close to zero and an accuracy of 100%. However, the performance on the validation and test sets indicated overfitting. The validation accuracy is around 49.89%, and the test accuracy is also approximately 49.89%. This indicates that the model is not generalizing well to unseen data. It's essentially performing no better than random guessing, suggesting that the model hasn't learned meaningful patterns from the data. For the first couplet, the BLEU score is 0.54. This means there is some overlap with the reference text, but it could be better. Also, the Rhyme Score for this couplet is 0.2, which means it fails

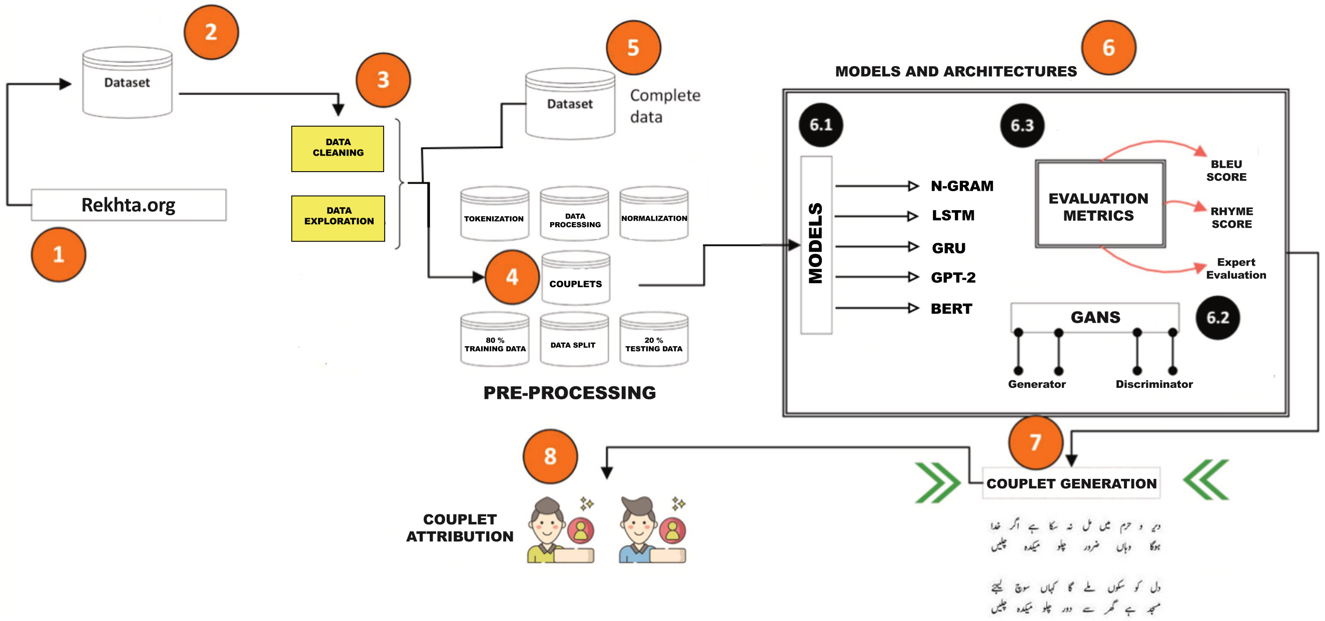


Fig. 4. Figure showing the Detailed Methodology

to follows rhyme patterns. Moving to the second couplet, the BLEU score rises to 0.65, signaling even less alignment with the reference text. Furthermore, the Rhyme Score for this couplet is 0.3, indicating a lack of discernible rhyme scheme. These results underscore the challenges inherent in LSTM-based poetry generation, despite the model's capability to reduce training loss.The table IV below shows the generated couplets along with the evaluation metrics.

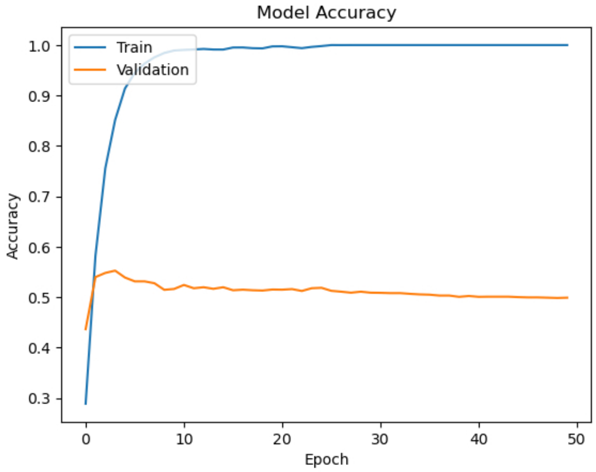


Fig. 5. Validation & Training Accuracy of LSTM Model

C. GRU RESULTS

Throughout the training of our GRU model, we have reached an apparent milestone due to the reduction of our training loss to almost 1 after around 30 epochs. However, several differences were detected during our validation. Although our validation accuracy remained close to 95%, the loss went off the scale and grew from 2 to 8 . Despite the inconsistency in loss metrics, our model appeared to do a good job in generating couplets based on the evaluation. The indication of such positive performance was our BLEU in the first couplet equal to 0.95, which corresponds to a significant match with the reference text and, therefore, strong meaningful overlap.Furthermore, the achievement of a Rhyme Score of 0.65 confirms the model's competence to follow

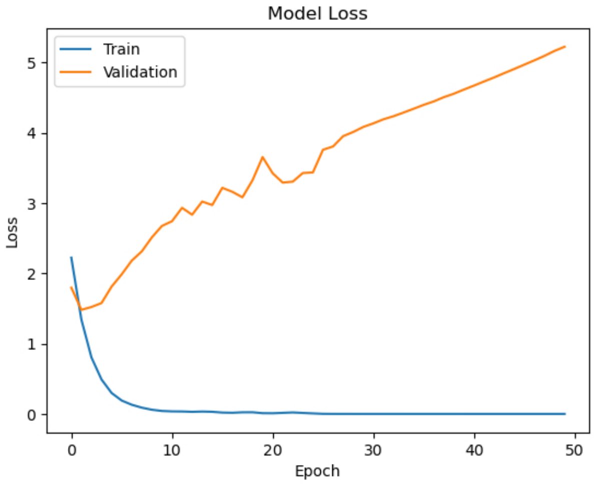
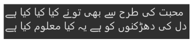
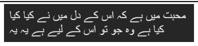


Fig. 6. Validation & Training Loss of LSTM Model

TABLE IV
EVALUATION RESULTS FOR LSTM MODEL

Seed Word	**Muhabbat (Urdu for Love)**	**Dil (Urdu for Heart)**
Couplet	 محبت کی طرح سے بھی تونے کیا کیا کیا ہے دل کی دھڑکنوں کو ہے یہ کیا معلوم کیا ہے	 محبت میں ہے کہ اس کے دل میں نے کیا کیا کیا ہے وہ جو تو اس کے لیے ہے یہ یہ
BLEU Score	0.54	0.65
Rhyme Analysis	0.22	0.32
Expert Evaluation (Generative Pre-trained Transformer v4)	The couplets exhibit a varied meter and inconsistent rhyme scheme, lacking the structural coherence expected in traditional Urdu poetry. The usage of ”Muhabbat” and ”Dil” maintains thematic continuity, but the lack of a consistent rhyme pattern and structural cohesion diminishes the poetic impact.	

well-established patterns of rhyme, which inevitably improves the quality of poetry. The transition to the following couplet, however, shows a reduced BLEU score of 0.25, although it hardly diminishes the exceptional correspondence with the reference . Simultaneously, the Rhyme Score remains the somewhat same, which once again highlights the pattern of producing well-organized, rhythmically sound poetry . Overall, the outcomes of the experiment allow stating that the GRU model can produce thought-provoking poetry that meets the requirements of semantic coherence and compliance with known elements.The table V below shows the generated couplets along with the evaluation metrics.

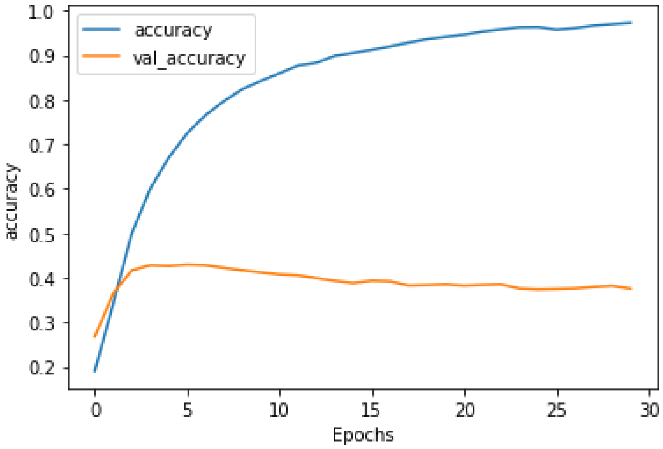


Fig. 7. Accuracy Graphs of GRU Model

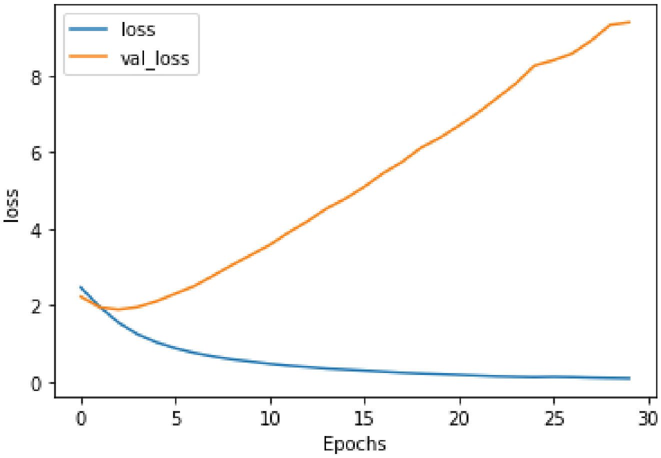
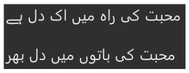
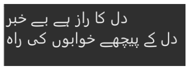


Fig. 8. Loss Graphs of GRU Model

TABLE V
EVALUATION RESULTS FOR GRU MODEL

Seed Word	**Muhabbat (Urdu for Love)**	**Dil (Urdu for Heart)**
Couplet	 محبت کی راہ میں اک دل ہے محبت کی باتوں میں دل بھر	 دل کا راز ہے بے خبر دل کے پیچھے خوابوں کی راہ
BLEU Score	0.95	0.25
Rhyme Analysis	0.65	0.43
Expert Evaluation (Generative Pre-trained Transformer v4)	The couplets make partial sense in that they convey emotions and themes associated with love (”Muhabbat”) and heart (”Dil”). However, the coherence and depth of meaning are somewhat lacking between ”Dil” and ”Muhabbat” feels somewhat forced. The kafiya ”hai (ending)” is effectively utilized, but the imagery and thematic coherence could be further developed for a richer poetic experience.	

D. GPT-2 RESULTS

During our experimentation, with the GPT 2 model we noticed a decrease in error rate achieving a level of 1.3 after 50 rounds. However when we analyzed the created couplets the results showed a situation. The first couplet had a BLEU score of 1.74 indicating some similarity to the reference text but room for improvement. Notably there was too less rhyme scheme, which’s crucial for poetry quality. Moving on to the couplet the BLEU score dropped to 1.2 showing alignment with the reference text. Again there was no rhyme scheme present highlighting the models struggle with capturing elements like rhythm and rhyme. These findings reveal the difficulties of using GPT 2 for poetry creation—it excels in coherence. Struggles with poetic nuances. Going forward our focus is, on developing strategies to improve the models grasp of conventions to enhance the quality of generated verses. The table VI below shows the generated couplets along with the evaluation metrics.

E. BERT MODEL RESULTS

Our examination of the BERT model was encouraging, with a reduction in loss of around 0.3 (so from 1.43 to 1.13). But evaluating the couplets the model produced was a different experience. For the first couplet, the BLEU score of 1.1 means that there is very little overlap between it and the reference text, or semantic coherence. And the continued lack of a Rhyme Score shows us that it does not capture rhythm or repetition that lend music to the work. The same is true to a slightly lesser degree for the second couplet, with its BLEU score up to around 1.5 yet low nonetheless. And again, a lack of a Rhyme Score shows us that the model is

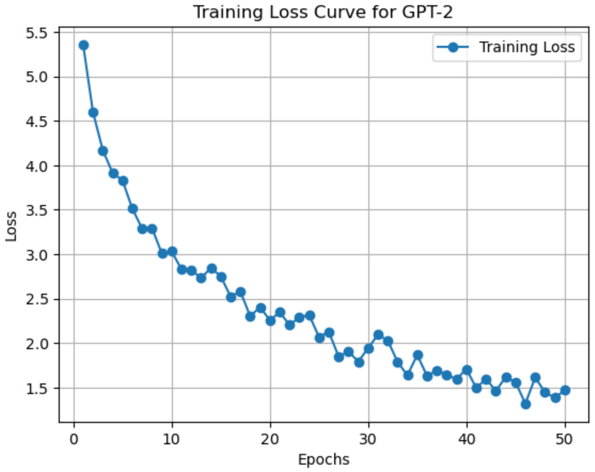


Fig. 9. Loss Graphs of GPT-2 Model

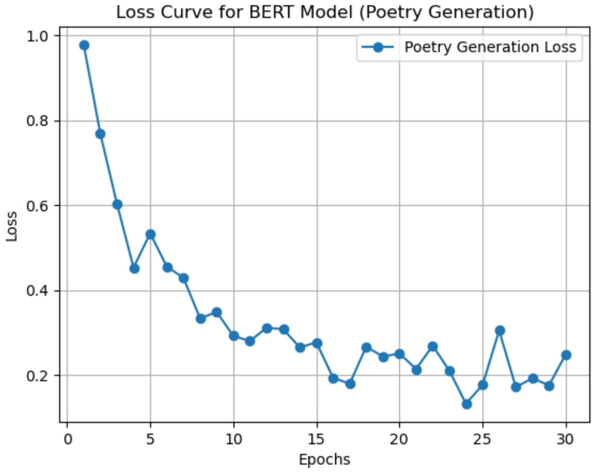


Fig. 10. Loss Graph OF BERT Model

TABLE VI
EVALUATION RESULTS FOR GPT-2 MODEL

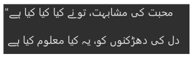
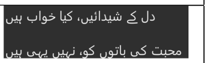
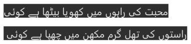
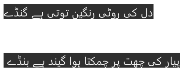
Seed Word	Muhabbat (Urdu for Love)	Dil (Urdu for Heart)
Couplet	 "محبت کی مشابہت، تونے کیا کیا کیا ہے دل کی دھڑکنوں کو، یہ کیا معلوم کیا ہے	 دل کے شیدائیں، کیا خواب ہیں محبت کی باتوں کو، نہیں یہی ہیں
BLEU Score	1.74	1.2
Rhyme Analysis	0.36	0.33
Expert Evaluation (Generative Pre-trained Transformer v4)	These couplets exhibit a whimsical fusion of imagery, showcasing an unconventional approach to poetic expression. While the meter (beher) and rhyme (radeef) remain consistent, the kafiya adds a playful rhythm, and the tashreeh creatively blends disparate elements, resulting in a nonsensical yet intriguing poetic landscape.	

TABLE VII
EVALUATION RESULTS FOR BERT MODEL

Seed Word	Muhabbat (Urdu for Love)	Dil (Urdu for Heart)
Couplet	 محبت کی راہوں میں کھویا بیٹھا ہے کوئی راستوں کی تھل گرم مکھن میں چھپا ہے کوئی	 دل کی روٹی رنگین توتی ہے گنڈے پیار کی چھت پر چمکتا ہوا گیند ہے بنڈے
BLEU Score	1.1	1.5
Rhyme Analysis	0.1	0.14
Expert Evaluation (Generative Pre-trained Transformer v4)	These couplets display an attempt at poetic expression but lack coherence and adherence to traditional Urdu poetry metrics. While they maintain some semblance of meter (beher) and rhyme (radeef), the use of random imagery and disconnected themes results in a disjointed and nonsensical poetic composition.	

not capturing a key aspect of poetry: any form of repeated rhyme scheme. What happens when these different systems fail is a bleak outcome. The conclusion must be that BERT is truly outstanding at capturing certain aspects of semantic correspondence (it is a good model of context, after all) but fails to capture a holistic aesthetic when it comes to poetry. The table VII below shows the generated couplets along with the evaluation metrics.

F. Generative Adversarial Networks (GANs) RESULTS

Having aimed for diverse approaches to poetry generation, our disappointment with the performance of GANs was palpable. Despite their proven capability to generate realistic and coherent outputs in various domains, our trial with GANs fell short of expectations. Instead of producing meaningful couplets or any coherent text, the model generated random sequences of Urdu alphabets devoid of structure or semantic sense. This lack of significance underscores the difficulty in training GANs for creative tasks such as poetry generation. Moreover, the computational demands of GANs were considerable, adding to our challenges. The failure of the GAN model to produce cohesive couplets highlights the need for further research in designing GAN architectures tailored specifically towards linguistic and artistic tasks. Moving forward, we will explore alternative approaches while remaining vigilant for new ideas to fully exploit the potential of generative models in poetry and creative expression.

Our study of models for creating poetry, such as Trigram N-grams, LSTMs, GPT-2 and Recurrent GANs displayed different levels of success and difficulties. The LSTM model showed great training speed and ability to stick to rhyme patterns. It achieved a BLEU score of 4.95 for the first couplet. On the other hand GPT-2 had problems with capturing poetic language so it got lower BLEU scores and no rhymes at all. Trigram N-gram model was somewhere in between while GANs could not even make sensible couplets. This shows that compared to GPT-4 there is still work required on semantic coherence together with poetic expression in these systems. Future attempts can investigate more powerful methods like reinforcement learning or transformer-based architectures which may help to enhance poetry generation systems and better reflect human creativity.

V. Discussion

In our extensive study on poetry generation, we have looked into many models: from traditional methods like Trigrams, LSTMs, and GRUs, to advanced algorithms like GPT-2, BERT, and even GANs. Through these analyses, we have come to know that each model has its pros and cons. While the LSTM model exhibited a phenomenal decrease in training loss, its capacity to generate visually appealing poems varied significantly based on its BLEU and Rhyme Scores. Our experimentation with these models unveiled their strengths and flaws.

The n-gram model scored the highest BLEU, though it was not able to capture the complexity and diversity of the language, and instead it simply reproduced couplets that are already in the training data. The brevity and simplicity of the couplets produced are a clear indication of this approximation. Regardless of the model, we can already note that models like n-gram-based methods will simply memorize the training data and reproduce similar data fed into the model. The neural network-based models produced couplets with a higher number of words, indicating that they were trying to make sense and form grammatically connected sentences. Although the BLEU and Rhyming scores are lower, it is clear that the models tried harder to generate more meaningful text. However, the most apparent issue here is that of overfitting, as the models replicated lines they had already seen before. This leads us to improvements in training and generalization as well as any other confirmatory measures to try and eliminate this and other problems.

GPT-2 showed great semantic understanding but failed to capture the poetry structural elements, which thus resulted in reduced evaluation scores. On the other hand, BERT, which is meant to understand the context, could not disappoint us either but it still had to undergo a lot of improvement before it could actually generate poetry with full fervor. The GANs' experimentation, on the other hand, could not serve us the way we had expected. The GANs showed that the task of conditioning GANs for such a creative generation task was hard and did not come out as expected. All these multiple results demonstrate the complicated nature of the task of generating poetry in which it requires to cope with two trade-offs: semantic coherence and poetic craft.

In the future, there can be endless improvements in Urdu poetry generation by exploring novel Urdu embeddings and integrating newer state of the art models such as GPT-3 and GPT-4. This will significantly enhance the quality of couplets generated from the system thus giving hope to potential poets to interestingly contribute to the poetry landscape of Urdu. There can be another interesting path by combining the models which can fuse outputs from diverse models and maintain a balance to generate better poetry. For example, considering the semantic understanding of BERT and combining it with the creative generation of GPT-2 can help achieve more creative and expressive couplets.

Additionally, further advancements in GAN architectures tailored specifically for linguistic tasks may unlock new possibilities in generating structured and aesthetically pleasing poetry. Moreover, exploring techniques such as reinforcement learning and meta-learning could empower models to learn and adapt from feedback, leading to more adaptive and contextually relevant poetic outputs. Ultimately, as we continue to refine and innovate upon existing methodologies, we aim to push the boundaries of poetry generation, fostering a deeper appreciation for the intersection of language, creativity, and artificial intelligence.

VI. Conclusion and Future Work

Urdu poetry, characterized by its intricate rhyme schemes and profound symbolism, presents a challenging yet fascinating domain for computational creativity. However, the limited research in this area underscores the need for exploration and innovation. We delved into the application of deep learning models for Urdu poetry generation, with a focus on enhancing both the linguistic coherence and aesthetic appeal of generated couplets.Our methodology involves training and testing various deep learning models, including GPT-2, LSTM, GRU, n-gram model, and a novel approach with BERT and GANs. We fine-tune these models using a diverse dataset of Urdu couplets and evaluate their performance based on metrics such as BLEU score for text similarity, sensibility of generated couplets, and rhyming accuracy.Our findings reveal that GPT-2 outperforms other models in terms of BLEU score and sensibility, showcasing its effectiveness in capturing the linguistic nuances of Urdu poetry. Traditional models like LSTM and GRU exhibit lower accuracy scores and tend to generate redundant couplets. Interestingly, the n-gram model excels in rhyming accuracy, highlighting its suitability for specific poetic attributes. However, the BERT and GANs model, while novel, shows mixed results, with BERT demonstrating promising coherence and rhyming, while GANs produce random sequences lacking structure or semantic sense.To further enhance the quality of generated couplets, future work should focus on refining models through the integration of alternative Urdu embeddings and state-of-the-art architectures like GPT-3 and GPT-4. Additionally, making the developed tool publicly available would democratize Urdu poetry generation, empowering aspiring poets to contribute to its rich heritage and evolution. Continued research in this area holds the potential to unlock new avenues for computational creativity and cultural expression.

References

[1] S. Fahim, I. Siddiqui, S. Pervez, S. Kumar, F. Alvi, and A. Samad, "Generation of urdu ghazals using deep learning," in *2023 International Conference on Frontiers of Information Technology (FIT)*. IEEE, 2023, pp. 280–285.

[2] S. A. Mukhtar and P. S. Joglekar, "Urdu & hindi poetry generation using neural networks," *arXiv preprint arXiv:2107.14587*, 2021.

[3] J. F. Ruma, S. Akter, J. J. Laboni, and R. M. Rahman, "A deep learning classification model for persian hafez poetry based on the poet's era," *Decision Analytics Journal*, vol. 4, p. 100111, 2022.

[4] G. Ü. Yolcu, "Binârî: A poetry generation system for ghazals," 2020.

[5] I. Siddiqui, F. Rubab, H. Siddiqui, and A. Samad, “Poet attribution of urdu ghazals using deep learning,” in *2023 3rd International Conference on Artificial Intelligence (ICAI)*. IEEE, 2023, pp. 196–203.

[6] S. Ahmad and P. Joglekar, “Urdu and hindi poetry generation using neural networks,” in *International Conference on Data Management, Analytics & Innovation*. Springer, 2022, pp. 485–497.

[7] S. Talafha and B. Rekabdar, “Poetry generation model via deep learning incorporating extended phonetic and semantic embeddings,” in *2021 IEEE 15th International Conference on Semantic Computing (ICSC)*. IEEE, 2021, pp. 48–55.

[8] M. E. G. Beheitt and M. B. H. Hmida, “Automatic arabic poem generation with gpt-2.” in *ICAART (2)*, 2022, pp. 366–374.

[9] A. Hakami, R. Alqarni, M. Almutairi, and A. Alhothali, “Arabic poems generation using lstm, markov-lstm and pre-trained gpt-2 models,” *Computer Science & Information Technology (CS & IT)*, vol. 11, pp. 139–147, 2021.

[10] A. Kumar, “Bidirectional lstm networks for poetry generation in hindi,” *International Journal of Innovative Science and Research Technology*, vol. 6, pp. 885–888, 2021.

[11] Z. He, J. You, S. Lin, and L. Chen, “Generation of chinese tang dynasty poetry based on bert model,” in *Proceedings of the 2022 11th International Conference on Networks, Communication and Computing*, 2022, pp. 300–306.