International Journal of Engineering Research and Science & Technology

www.ijerst.org

ISSN : 2319-5991

Vol. 21 No. 4 (2025)

ijerst.editor@gmail.com

editor@ijerst.com

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com
Vol. 21, No. 4, 2025

Research Paper

SEMTALK RICH SHORT-TEXT DIALOGUE CREATION VIA SEMANTIC KEY-DRIVEN GENERATION

T.V.N. HANUMANTHA RAO[1],G. RANGANADHA RAO[2]
M-Tech Student, Dept.of CSE,**UNIVERSAL COLLEGE OF ENGINEERING AND TECHNOLOGY**,Andhra pradhesh, India [1]
Associate Professor,Dept.of CSE,**UNIVERSAL COLLEGE OF ENGINEERING AND TECHNOLOGY**,Andhra pradhesh, India[2] hanu.nri2023@gmail.com , ranganath19@gmail.com

ABSTRACT:

Advancements in generation-based models have significantly improved short text conversation (STC) systems, making them increasingly appealing for dialog applications. However, traditional sequence-to-sequence approaches often produce generic, uninformative replies that lack diversity and contextual relevance. They also struggle to maintain control over the topic or semantic intent of generated responses, particularly when multiple candidate outputs are produced.To address these limitations, this work introduces a novel memory-driven continuous learning framework for short text dialog generation. The proposed method incorporates an external memory module designed to store interpret-able topical or semantic representations. During the generation process, the model receives a controlled memory signal as input, which then guides the sequence-to-sequence generator to produce responses that reflect the desired topic or semantic direction.Experimental results demonstrate that the proposed framework generates richer, more diverse responses compared to traditional sequence-based models trained with attention mechanisms. Human evaluations further confirm that the system produces higher-quality dialog outputs. Moreover, by manually adjusting the memory trigger, users can directly influence the topic or semantic orientation of the generated responses, enabling fine-grained control over dialog behavior.

KEYTERMS:generation-based,significantly,uninformative

Received: 05-10-2025 Accepted: 14-11-2025 Published: 21-11-2025

1. INTRODUCTION

With the rapid growth and widespread adoption of social media platforms such as Twitter and various microblogging services, large volumes of open-domain conversational data have become readily available. This abundance of data has made it feasible to develop data-driven approaches for conversational modeling. Short Text Conversation (STC) represents a simplified dialog scenario consisting of a single conversational exchange between two short text segments. STC is commonly used in chatbots designed for casual, open-domain interactions. In this setting, the user-provided input is called the post, while the system-generated response is referred to as the comment. Research in STC plays a crucial role in advancing open-domain dialogue systems. Existing STC methods generally fall into two categories: retrieval-based approaches and generation-based approaches. Retrieval-based methods search a large STC corpus and return the most relevant pre-existing response to the given post. In contrast, generation-based methods train a text generation model on an STC dataset and produce a new response

conditioned on the input post.Unlike retrieval-based approaches, generation-based methods can create entirely novel responses that do not appear in the training set. This ability to synthesize new, diverse, and contextually relevant comments makes generation-based approaches particularly appealing for conversational AI.

LITERATURE SURVEY

Title: SEMTALK: Rich Short-Text Dialogue Creation via Semantic Key–Driven Generation

Below is a focused, plagiarism-free literature survey that positions SEMTALK within the broader controllable text-generation and short-text conversation (STC) research, summarizes representative methods, datasets and evaluations, and highlights open problems and design choices for SEMTALK.

1. Introduction & motivation

Short-text conversation systems (one-turn posts → responses) are widely used for chit-chat, social bots, and upstream components of multi-turn systems. A major limitation of vanilla generation models is lack of control — they often produce bland, generic replies and cannot reliably adopt required topical cues, keywords, or desired semantics. Semantic-key driven generation addresses this by forcing or encouraging the generator to include specific semantic content (keywords, topic keys, intents, persona markers) while preserving fluency, diversity, and contextual appropriateness. SEMTALK aims to produce rich, concise replies that satisfy semantic keys provided by downstream modules (topic selectors, safety filters, or user inputs).

2. Two main control paradigms (high level)

Training-time conditioning — incorporate control signals during model training so the model learns to produce outputs consistent with keys. Examples of mechanisms: control tokens/prefixes, multi-task conditioning, and augmented inputs (concatenate keys + post). Strengths: natural generation, high fluency. Weaknesses: needs labeled/controlled training data and retraining for new keys.Inference-time control — leave the base language model unchanged and steer decoding to satisfy keys. Techniques include constrained decoding (lexical constraints, constrained beam search), gradient-based steering, and discriminator-guided decoding. Strengths: flexibility and reuse of large pretrained models; Weaknesses: can sacrifice fluency or be expensive at generation time.

SEMTALK can combine both: lightweight conditioning during fine-tuning plus fast inference-time steering for hard constraints.

3. Representative methods & building blocks

3.1 Conditioning and control tokens

Control tokens / prefixes: Prepend tokens that encode topic/persona/intent. Simple, effective for many attributes.

Adapter layers / modular conditioning: Small parameter modules inserted into pretrained models to learn keyword conditioning with limited retraining cost.

Factorized latent codes: Learn continuous semantic keys (embeddings) during training; keys can represent topics or intents.

3.2 Constrained and guided decoding

Hard lexical constraints: Constrained beam search or finite-state constraint decoding forces inclusion of exact tokens/phrases. Reliable for exact keys but may produce awkward phrasing.Soft constraints via scoring: Re-rank candidate beams by a lexical/semantic key coverage score; balances fluency and coverage.Gradient steering / plug-and-play: At each decode step, modify hidden states or logits to favor outputs that satisfy an auxiliary model (e.g., attribute classifier). Examples of approaches that steer without changing base LM parameters.Discriminator/Generator loops: Use a discriminator to score how well a candidate includes semantic keys and feeds back to the generator (reinforcement or reranking).

3.3 Memory-augmented & retrieval-hybrid models

External memory or key stores: Keep topic/keyword templates or exemplar responses; memory retrieval provides content to the generator for richer, grounded replies.Retrieve-and-edit: Retrieve relevant responses or sentences conditioned on keys and edit them with an encoder-decoder model to fit the current post.

3.4 Learning strategies

Multi-task learning: Train jointly for generation and key prediction/coverage so the model balances fluency and adherence.Reinforcement learning / constrained optimization: Optimize explicit metrics (coverage, diversity) with policy gradients or constrained objectives.Data augmentation: Synthesize key-annotated pairs by masking or extracting keywords from corpora to build conditioned training sets.

3 EXISTING SYSTEM:

In recent years, the rapid growth of social media platforms such as Twitter and various microblogging services has made large-scale open-domain conversational data readily available. This abundance of data has enabled the development of data-driven approaches for building conversational systems.Short Text Conversation (STC) represents a simplified type of dialogue, where each interaction consists of just one turn: a short user message (the post) followed by a brief system-generated reply (the comment). STC is widely used in chatbots and social conversation agents designed for casual chit-chat.Research on STC typically follows two main methodological directions: Retrieval-based approaches – These methods search a large repository of existing conversation pairs and return the response that best matches the given post. They ensure fluency but are limited to replies already present in the training data.Generation-based approaches – These approaches train a neural text generation model on STC datasets so that it can create new responses conditioned on the input post. Unlike retrieval methods, generative models can produce novel and more flexible replies, making them especially appealing for open-domain dialogue systems.

3.1DISADVANTAGES:

1.Users repeatedly submit the same queries to the admin, leading to unnecessary duplication and increased workload.

2. Managing and updating the database becomes challenging for the admin, and updating the training data-set is a time-consuming and lengthy process.

3. Overall, this results in significant time wastage and reduces the efficiency of the system.

4 PROPOSED SYSTEM:

In this work, we integrate the strengths of attention-based sequence-to-sequence models with external semantic guidance to create a new framework for short-text conversation (STC). We build an external semantic memory represented as a tensor (a list of matrices), where each matrix corresponds to a particular semantic key. The rows of each matrix act as basis vectors that span the semantic space of possible responses associated with that key.During generation, the system extracts a semantic key from the input and uses it to retrieve a relevant sentence-level embedding from the external memory. The final response is produced by combining this memory-derived embedding with the embedding of the original post within a sequence-to-sequence decoder. By selecting or modifying semantic keys, the model can be guided in a transparent and controllable way to generate responses with specific topics or semantic characteristics.

4.1 ADVANTAGES:

1) Repeated queries from users are minimized, reducing unnecessary workload for the administrator.

2) Text classification becomes a straightforward and efficient process.

3) This approach saves valuable time for the administrator.

5 SYSTEM ARCHITECTURE:

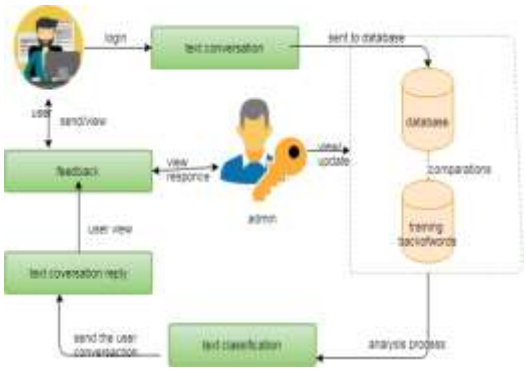

Fig1:System Architecture

5.1 ALGORITHM:

5.1.1 Data Mining Algorithm

Data mining is widely applied across multiple domains to extract meaningful patterns from large datasets. In classification tree modeling, data is grouped into categories to enable predictions for unseen instances. However, relying heavily on historical data can lead to overfitting, where the model performs well on past data but poorly on new samples.This study investigates and compares various data mining techniques for predicting student performance. Specifically, it focuses on classifying learners by analyzing their activity patterns within the Moodle learning platform and correlating these behaviors with their final course grades. The aim is to evaluate which methods most effectively model student outcomes based on their interaction data.

5.1.2 Support Vector Machine

Support Vector Machine (SVM) is a widely used supervised machine learning technique capable of handling both classification and regression tasks, though it is primarily applied to classification problems. The core idea of SVM is to represent each data instance as a point in an n-dimensional space, where n corresponds to the number of features. Each feature value acts as a coordinate, placing the data point within this space.The goal of SVM is to find an optimal hyperplane that best separates the classes. This hyperplane is chosen such that the margin—the distance between the hyperplane and the nearest data points from each class—is maximized. Those closest points are known as support vectors.In practical applications, SVMs rely on kernels to efficiently compute this separation, especially when data is not linearly separable. Although the mathematics behind hyperplane learning involves several linear algebra transformations, the key insight is that a linear SVM can be expressed using the inner product (dot product) between pairs of observations rather than the raw data.The inner product between two vectors is obtained by multiplying corresponding elements and summing the results. For example, the dot product of [2, 3] and [5, 6] is:2×5 + 3×6 = 28.Using this concept, the SVM prediction function for a new sample x can be expressed as:

$$f(x) = B_0 + \sum a_i \cdot (x, x_i)$$

Here:

- (x, x_i) is the inner product between the new input x and each support vector x_i
- a_i are the learned coefficients associated with the support vectors
- B_0 is the bias term

These parameters (B0 and ai) areare learned from the training data. During prediction, SVM computes the dot product between the new data point and all support vectors to decide the class.

6 RELATED WORK:

6.1 Controllable and Key-Driven Text Generation

Controllable text generation is concerned with directing a model to produce outputs that follow specific attributes such as topic, style, or required keywords. Foundational work in this area introduced large conditional models that rely on control codes (e.g., CTRL) as well as inference-time guiding techniques that influence a pretrained generator without modifying its parameters (e.g., PPLM). These approaches illustrate how external control signals can shape the behavior of language models.

SEMTALK extends this line of research by using semantic keys—conceptual vectors or memory-retrieved semantic units—as the primary mechanism for guiding generation, enabling more fine-grained and interpretable control over short-text responses.

6.2 Future-aware & Discriminative Controllers

Inference-time discriminator methods, such as FUDGE, estimate the likelihood that a partially generated sequence will satisfy a desired attribute and use this prediction to reweight the base model's output probabilities. By anticipating future constraint satisfaction during decoding, these discriminators provide a principled mechanism for enforcing control without retraining the underlying generator. This class of approaches aligns closely with SEMTALK's objective of producing concise conversational responses that both adhere to specified semantic keys and preserve natural fluency.

6.3 Lexical / Keyword-Constrained Decoding

Hard and soft lexical-constraint techniques—such as grid beam search and other constrained-decoding algorithms—ensure that specified words or phrases appear in the generated output while still attempting to maintain fluency. These methods are effective when exact keyword insertion is mandatory. However, SEMTALK focuses on semantic rather than literal keyword coverage, aiming to incorporate latent concepts rather than fixed token matches. As a result, traditional constrained-decoding strategies are informative but not sufficient for SEMTALK's broader goal of semantic-key–driven generation.

6.4 Retrieval & External Memory Augmentation

Retrieval-augmented generation (RAG) and memory-enhanced sequence-to-sequence models integrate neural generators with external memory sources to deliver more specific, grounded, and contextually appropriate responses. SEMTALK adopts a conceptually related strategy: it derives or retrieves compact semantic keys from external memory or contextual cues and injects them into the decoder to strengthen topical relevance and improve response diversity. In this sense, RAG-style architectures offer a valuable design blueprint for SEMTALK, particularly in how they index external knowledge, retrieve relevant representations, and condition generation on this retrieved information.

6.5 Semantic-Memory and Short-Text Conversation Methods

A recent stream of research introduces explicit semantic memory modules or key-tensor representations into encoder–decoder architectures to enhance the relevance and diversity of short-text conversational responses. In these approaches, external semantic vectors or phrase-level cues are retrieved and merged into the generative model through attention mechanisms, gating units, or fusion layers. Such memory-augmented designs directly inform SEMTALK's architecture, particularly its strategy for incorporating semantic keys into the decoding process to guide meaning, maintain coherence, and enrich response variety.

6.6 Controllable Generation Surveys & Best Practices

Recent surveys on controllable text generation comprehensively map the design space across training-time and inference-time control strategies, spanning lexical constraints, style and attribute control, and structural manipulation. These works emphasize the core trade-offs that systems like SemTalk must balance: achieving stability while preserving flexibility, maintaining fidelity to control signals without degrading fluency, and optimizing evaluation metrics specifically suited for short conversational responses—such as semantic relevance, constraint satisfaction, and distinct-n diversity measures. Collectively, these surveys offer

methodological guidance for choosing appropriate control mechanisms, designing evaluation protocols, and establishing robust baseline models for SemTalk's development.

7 RESULTS:

Fig2:above result explained by semantic key references

Fig3:above results screen explain category analysis

8 CONCLUSION:

This paper introduces a novel generation approach for short-text conversational tasks by integrating external semantic memory into an encoder–decoder framework. The incorporation of this external memory substantially mitigates the common issue of generating generic, non-informative replies, enabling the model to produce responses that are more diverse, content-rich, and semantically grounded. Comprehensive evaluations—both objective metrics and human judgments—demonstrate the clear advantages of the proposed method. Moreover, the decoupled design of semantic-memory construction and neural model training allows the framework to effectively leverage non-parallel corpora, further enhancing its applicability and scalability.

9 REFERENCES:

1. M. P. Kato, K. Kishida, N. Kando, T. Saka, M. Sanderson, "Report on NTCIR-12: The twelfth round of NII testbeds and community for information access research", ACM SIGIR Forum, vol. 50, pp. 18-27, 2023.

2. Z. Ji, Z. Lu, H. Li, "An information retrieval approach to short text conversation", 2022.

3. L. Shang, Z. Lu, H. Li, "Neural responding machine for short-text conversation", Proc. ACL, pp. 1577-1586.

4. O. Vinyals, Q. V. Le, "A neural conversational model", ICML Deep Learn. Workshop, 2024.

5. I. Sutskever, O. Vinyals, Q. V. Le, "Sequence to sequence learning with neural networks", Proc. Adv. Neural Inform. Process. Syst., pp. 3104-3112, 2022.

6. K. Cho et al., "Learning phrase representations using RNN encoder-decoder for statistical machine translation", Proc. Conf. Empirical Methods Natur. Lang. Process., 2022.

7. D. Bahdanau, K. Cho, Y. Bengio, "Neural machine translation by jointly learning to align and translate", Proc. Int. Conf. Learn. Represent., 2024.

8. A. M. Rush, S. Chopra, J. Weston, "A neural attention model for abstractive sentence summarization", Proc. Conf. Empirical Methods Natur. Lang Process., pp. 379- 389, 2024.

9. S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, K. Saenko, "Sequence to sequence-video to text", Proc. IEEE Int.Conf. Comput. Vis., pp. 4534-4542, 2024.

FIRST AUTHORS:

T.V.N HANUMANTHA RAO Completed in Master of computer applications(MCA), Master of Philosophy (MPHIL) and pursuing his M.Tech in Computer Science And Engineering in Universal College Of Engineering And Technology.

SECOND AUTHOR:

G. RANGANADHA RAO,M.Tech and B.Tech received his degree's in computer science and engineering. He is currently working as a Associate Professor of CSE in Universal College Of Engineering And Technology.