ICRA2026 Late Breaking Results Poster presented at
2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

TeNet: Text-to-Network for Compact Policy Synthesis

Ariyan Bighashdel[1] and Kevin Sebastian Luck[1]

***Abstract*— Robots that follow natural-language instructions typically rely on either high-level planners with hand-designed interfaces or large end-to-end models that are difficult to deploy for real-time control. We propose TeNet (Text-to-Network), a framework that instantiates compact, task-specific policies directly from natural language. TeNet conditions a hypernetwork on embeddings from a pretrained language model to generate a fully executable policy, which operates solely on low-dimensional state inputs at high control frequencies. By using language only once at policy instantiation, TeNet combines the expressiveness of large language models with efficient execution. To improve generalization, we optionally ground language in behavior during training, without requiring demonstrations at inference. Experiments on MuJoCo and Meta-World show that TeNet produces policies that are orders of magnitude smaller than sequence-based baselines, while achieving strong performance in both multi-task and meta-learning settings and enabling high-frequency control. These results demonstrate that text-conditioned hypernetworks provide a practical approach for compact, language-driven robot control.**

I. Introduction

Recent advances in large language models (LLMs), such as GPT [1] and LLaMA [2], have enabled natural language to serve as a flexible interface for robotic control. Vision–language–action systems show how language can specify complex behaviors, but typically rely on large end-to-end architectures that are difficult to deploy in real-time or resource-constrained settings.

At the other extreme, compact models such as Decision Transformers (DT) and Prompt-DT [3], [4] enable efficient policy learning, but lack direct language conditioning and often require demonstrations at inference time. Hypernetworks [5] provide a mechanism for task-conditioned parameter generation, but are typically driven by structured inputs rather than natural language.

This creates a gap between expressive language-conditioned systems and efficient, deployable policies. We ask whether natural language can directly parameterize executable policies.

We propose **TeNet (Text-to-Network)**, a framework that conditions a hypernetwork on LLM-based text embeddings to generate compact, task-specific policies. Language is used only once at policy instantiation, after which the resulting controller operates solely on low-dimensional state inputs, enabling high-frequency control without requiring demonstrations at inference.

To improve generalization, we optionally ground language in behavior during training by aligning text embeddings with expert trajectories. This grounding enriches linguistic representations with behavioral semantics, while remaining unnecessary at inference time.

We evaluate TeNet on MuJoCo and Meta-World benchmarks, covering both multi-task and meta-learning settings. Results show that TeNet achieves strong performance while producing policies that are orders of magnitude smaller and significantly faster than sequence-based baselines.

Contributions.

- **Text-to-Network Policy Generation.** A framework that maps natural language to executable policies via hypernetworks.
- **Language Grounding for Generalization.** Aligning language with behavior improves generalization without requiring demonstrations at inference.
- **Efficient Language-Driven Control.** Compact policies with high control frequency across standard benchmarks.

II. Problem Statement

We consider a distribution over tasks, each modeled as a language-augmented MDP (LA-MDP), i.e., a standard MDP paired with natural-language descriptions specifying the task. Each task is associated with expert trajectories and corresponding descriptions, potentially including paraphrases of the same objective.

We study the offline setting, where the learner is given a dataset of demonstrations and descriptions from training tasks and must generalize to unseen tasks. The goal is to learn a model that performs well in both multi-task and meta-learning settings.

Unlike prior approaches such as Prompt-DT, which rely on trajectory prompts at inference, we aim to instantiate policies directly from natural-language descriptions without requiring demonstrations at test time.

III. Method

TeNet (Text-to-Network) instantiates task-specific policies directly from natural language by conditioning a hypernetwork on text embeddings (Figure 1). Given a task description d, a pretrained text encoder produces an embedding z_d, which is projected and used by a hypernetwork to generate the parameters θ_π of a policy π_{θ_π}: $\theta_\pi = h\big(g(f_{\text{text}}(d))\big)$. The resulting policy operates solely on low-dimensional state inputs, enabling efficient execution at high control frequencies.

Training is performed offline using expert demonstrations with a behavior cloning objective. To improve generalization, we optionally introduce **grounding**, where text embeddings

[1]Ariyan Bighashdel and Kevin Sebastian Luck are with the Vrije Universiteit Amsterdam, `a.bighashdel@vu.nl, k.s.luck@vu.nl`

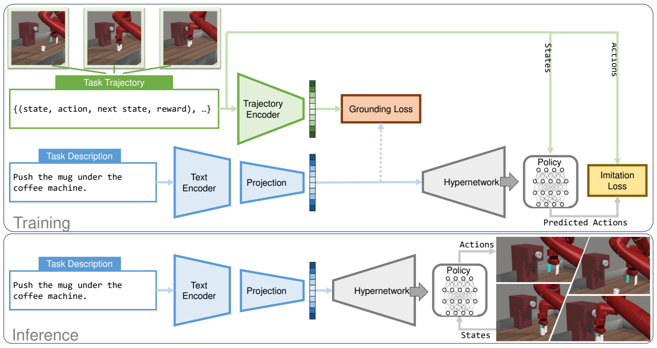

Fig. 1: TeNet overview. A task description is encoded and used by a hypernetwork to generate a task-specific policy. During training, optional grounding aligns text with trajectories; at inference, only text is required.

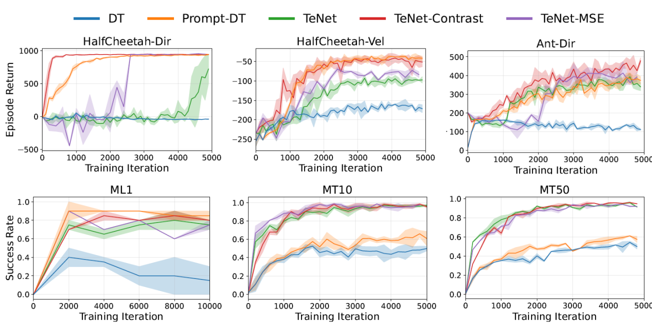

Fig. 2: Performance across MuJoCo and Meta-World benchmarks.

are aligned with trajectory embeddings produced by a trajectory encoder. This alignment (e.g., via contrastive objectives) enriches language representations with behavioral semantics.

At inference time, only a task description is required: the hypernetwork generates a policy in a single forward pass, without access to demonstrations. This contrasts with prior approaches that rely on trajectory prompts at test time.

IV. Experiments

Setup. We evaluate TeNet on MuJoCo locomotion (HalfCheetah-Dir, HalfCheetah-Vel, Ant-Dir) and Meta-World manipulation (ML1, MT10, MT50) benchmarks, covering both multi-task and meta-learning settings. We compare against Decision Transformer (DT) and Prompt-DT [3], [4]. Performance is measured using episodic return (MuJoCo) and success rate (Meta-World), averaged over 3 seeds.

Main Results. Figure 2 summarizes performance across all benchmarks.

TeNet outperforms DT and matches or exceeds Prompt-DT across benchmarks, with large gains in multi-task settings (MT10, MT50) and competitive performance in meta-learning.

Key Insights. (i) **Language as task signal.** Direct TeNet already improves over DT, showing that language provides an effective task representation.

(ii) **Grounding improves generalization.** Aligning text with trajectories improves performance on unseen tasks, with contrastive alignment performing best.

TABLE I: Performance and efficiency on MT10/MT50.

Model	MT10	MT50	Size	Freq.
Prompt-DT	0.73	0.61	1M	557 Hz
Prompt-DT-HN	0.99	0.97	5M	462 Hz
TeNet	**0.99**	**0.98**	**40K**	**9kHz**

(iii) **Task-specific parameterization is critical.** Prompt-DT struggles on diverse multi-task benchmarks, while hypernetwork-based parameter generation enables TeNet to scale effectively.

(iv) **Efficiency.** TeNet produces compact policies ($\sim$40K parameters) that run at high control frequencies (>9kHz), compared to 1M–39M parameters and sub-kHz rates for Prompt-DT (see Table I).

V. Conclusion

We introduced TeNet, a framework for instantiating compact, task-specific policies directly from natural language via hypernetworks. By leveraging language as a conditioning signal and optionally grounding it in behavior during training, TeNet enables policies that generalize across tasks without requiring demonstrations at inference. Experiments show that TeNet achieves strong performance while producing lightweight controllers that operate at high control frequencies. These results highlight text-conditioned hypernetworks as a practical approach for efficient and deployable robot control, suggesting a path to bridging expressive language interfaces with efficient low-level control.

References

[1] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, and Others, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[2] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, and Others, "LLaMA: Open and Efficient Foundation Language Models," *arXiv preprint arXiv:2302.13971*, 2023.

[3] L. Chen, K. Lu, A. Rajeswaran, K. Lee, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, and I. Mordatch, "Decision transformer: Reinforcement learning via sequence modeling," *Advances in neural information processing systems*, vol. 34, pp. 15 084–15 097, 2021.

[4] M. Xu, Y. Shen, S. Zhang, Y. Lu, D. Zhao, J. Tenenbaum, and C. Gan, "Prompting decision transformer for few-shot policy generalization," in *international conference on machine learning*. PMLR, 2022, pp. 24 631–24 645.

[5] D. Ha, A. Dai, and Q. V. Le, "Hypernetworks," *arXiv preprint arXiv:1609.09106*, 2016.