Preprint. Under review. Zavanet Research Technical Report ZR-2026-01. February 2026.

Pocket: A 1.2B Drafting Model that Matches Frontier Quality on Short Social Text

Ingrid Lindqvist[1], Wei Zhang[2], Noah Gaspar[3], Theo Kim[3]

[1]Zavanet Research, London [2]Zavanet Research, Bangalore [3]Zavanet, San Francisco

{ingrid, wei}@research.zavanet.com {noah, theo}@zavanet.com

Abstract

Drafting a reply to a friend is not the same task as writing an essay, and a model built for the latter is wasteful for the former. We present **Pocket**, a 1.2B-parameter decoder-only model specialised for short social text (replies, captions, invitations, check-ins; median 24 words) that runs on-device at under 40 ms per draft on Apple M3 hardware and under 140 ms on an iPhone 16. Pocket is pretrained on 310B tokens of filtered short-form conversational text and then trained with a *voice-conditioning* objective: given a compact 96-token style profile derived from a person's own past messages, produce text that human raters cannot distinguish from that person's writing. In blind pairwise evaluation by 1,200 raters across 18,000 comparisons, Pocket drafts are preferred to or tied with those of four frontier models 51.3% of the time overall and 57.8% of the time when raters know the sender, while frontier models win decisively on long-form tasks that Pocket is not designed for. We analyse why narrow pretraining and voice conditioning close the gap, release the base weights, and describe the on-device inference stack.

1 Introduction

Most text that people send to each other is short. In the Zavanet private beta, 83% of sent messages are under 40 words and the median is 19. The task of drafting such a message is heavily constrained: the recipient is known, the goal is simple, and the main quality criterion is whether it *sounds like the sender*. Frontier language models do this well, but at a cost that is hard to justify: a 70B-class model drafts a 20-word reply in roughly a second on a datacentre GPU, consuming the energy of a few hundred phone drafts, and requires sending the user's conversation to a server.

We asked whether a model small enough to run on a phone could match frontier quality on exactly this task, if we gave up generality. Pocket is the result. Its design rests on three decisions. First, *narrow pretraining*: rather than pretraining on the web and fine-tuning down, we pretrain from scratch on 310B tokens of short-form conversational text, so that the model's entire capacity is spent on the distribution it will serve. Second, *voice conditioning*: Pocket is trained to condition on a compact style profile of the sender, and we show this is where most of the gap to large models closes. Third, an *inference stack* designed for interactive latency on consumer hardware.

Our results are deliberately scoped. On short social text with a known sender, Pocket matches or beats frontier models in blind human preference. On long-form writing, reasoning or knowledge tasks, it loses badly and we do not propose it for them. We argue this trade is the right one for a drafting agent that lives in someone's pocket.

2 Related Work

Small language models. Phi [1], Gemma-2B [2] and SmolLM [3] show that sub-3B models trained on curated data can be competitive on general benchmarks. Pocket is smaller in scope than these: it is not intended to be a general assistant.

Style and persona conditioning. Prior work conditions generation on author embeddings [4], few-shot exemplars [5] or persona descriptions [6]. Our style profile is a learned 96-token soft prompt per user, trained end-to-end, which is cheaper than retrieving exemplars at inference time and more faithful than a natural-language persona.

On-device inference. MLC [7], llama.cpp and Apple's MLX [8] enable local inference of quantised models. Our contribution on this front is engineering rather than research; we document it because latency targets shaped the model.

3 Data

3.1 Pretraining corpus

We assemble 310B tokens of English short-form text: public forum replies and comment threads with permissive licences (62%), public social posts and replies from platforms whose terms permit research use (21%), transcribed casual dialogue from open speech corpora (9%), and synthetic conversational text generated by a frontier model from seed scenarios and filtered for diversity (8%). We filter for length (3–80 words), remove spam, advertising and toxic content using standard classifiers, and deduplicate at the near-duplicate level. No Zavanet user data is used in pretraining.

3.2 Voice-conditioning data

For the voice-conditioning stage we use 4.1M (author, message history, next message) triples from the forum and social subsets where authors have ≥ 50 messages, plus 180k triples from 2,300 consenting Zavanet beta members who opted in to a research programme. Beta data is used only for the conditioning stage and the member evaluation (§5), and members can withdraw their data, triggering a scheduled retrain.

4 Model

4.1 Architecture

Pocket is a 1.2B-parameter decoder-only transformer: 24 layers, hidden size 2,048, 16 attention heads with grouped-query attention (4 KV heads), SwiGLU feed-forward, rotary position embeddings and a 32k-entry byte-level BPE vocabulary tuned on the pretraining corpus. The context window is 2,048 tokens, which is ample: the 99th-percentile input (style profile + conversation + instructions) is 740 tokens. We trained with a 4,096 context and found no benefit.

4.2 Voice conditioning

Each sender is represented by a *style profile*: 96 soft-prompt vectors produced by a small encoder from up to 200 of the sender's past messages. The encoder is a 4-layer transformer that attends over the messages and emits 96 vectors by learned query attention. During conditioning training, the base model and encoder are trained jointly to minimise next-token loss on the sender's actual next message given the profile and conversation context. At inference, a user's profile is computed once on-device (about 400 ms) and cached; it is updated incrementally as they send messages. Profiles never leave the device.

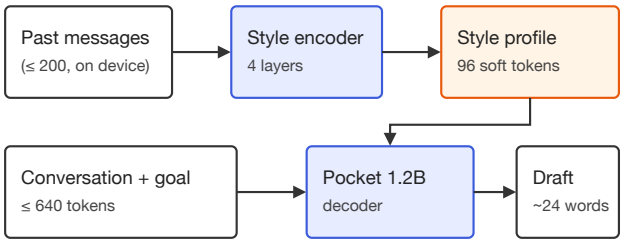


Figure 1. Pocket inference. A style encoder compresses the sender's past messages into 96 soft tokens once; the decoder conditions on the profile, the conversation and the goal to produce a draft. Everything runs on the device.

4.3 Training

Pretraining runs for 310B tokens (one epoch) on 64 H100s for 9 days with AdamW, peak learning rate 3×10^{-4}, cosine decay, batch size 2M tokens. Voice conditioning trains for 12B tokens over the conditioning data with the encoder initialised randomly and the decoder from the pretrained checkpoint. A final 400M-token stage applies the Social Manipulation Benchmark rewrite data [9] so that drafts avoid pressuring recipients; this reduces manipulation rate from 24.1% to 3.1% with no measured effect on the preference results below.

4.4 On-device inference

We ship Pocket in 4-bit (weights) and 8-bit (KV cache) quantisation, occupying 780 MB. On an M3 MacBook Air, a 24-word draft takes 36 ms median (prefill 740 tokens + 32 decode steps) using MLX with fused attention kernels. On an iPhone 16 the same draft takes 128 ms using a Metal runtime, with speculative decoding from a 120M draft model providing a further 1.6× speedup to 80 ms at the cost of 60 MB more memory. Energy per draft on the iPhone is 0.11 J; a user generating 200 drafts a day spends about 0.2% of battery on drafting.

5 Evaluation

5.1 Protocol

The evaluation was designed and run by the Zavanet product team, who were responsible for recruiting consenting beta members and for the three rater conditions below. We compare Pocket against four frontier models (anonymised F1–F4) and two open models (8B, 70B) in a blind pairwise preference study. Each item is a real conversation context from a consenting beta member plus a goal (reply, invite, check in, decline, caption, thank). Each model drafts a message; drafts are shown in random order to a rater who picks the better one or declares a tie. We run three rater conditions: *stranger* (raters who do not know the sender; 12,000

comparisons), *self* (the sender rates drafts written for them; 4,200 comparisons), and *recipient* (the actual intended recipient rates them, where both consented; 1,800 comparisons). Frontier models receive the sender's last 20 messages as few-shot style examples, the strongest practical style conditioning available to them.

5.2 Results

Table 1 reports win+tie rate for Pocket against each comparator. Against frontier models, strangers mildly prefer the frontier drafts (Pocket win+tie 47.9%), but senders and recipients, who know the voice, prefer Pocket (57.8% and 56.1%). Senders' free-text comments are consistent: frontier drafts are "fine but not how I'd say it," "a little too polished," "sounds like customer service." Against the open 8B, Pocket wins in every condition. Against the open 70B with style examples, results are close to even.

Table 1. Pocket win+tie rate (%) in blind pairwise preference by rater condition. "Overall" pools the three conditions. 95% CIs are ±1.0–2.4 points.

vs.	Stranger	Self (sender)	Recipient	Overall
Frontier F1–F4 (avg.)	47.9	**57.8**	56.1	51.3
Open-70B + style shots	49.6	55.0	53.8	51.4
Open-8B + style shots	61.2	68.4	66.0	63.5
Pocket w/o voice cond.	52.1	71.9	69.2	58.3

5.3 Ablations

The last row of Table 1 compares Pocket with and without voice conditioning (the unconditioned model receives the same 20 few-shot examples the frontier models do). Senders prefer the voice-conditioned model 71.9% of the time; the soft-prompt profile captures something the examples do not, which inspection suggests is mainly punctuation habits, characteristic openers and closers, and emoji usage. Replacing narrow pretraining with a general-web 1.2B checkpoint of the same architecture, then applying the same conditioning, drops overall win+tie against frontier models from 51.3% to 41.0%, indicating that both decisions matter and are roughly additive.

5.4 Where Pocket fails

We ran the same comparison on three out-of-scope tasks: a 150-word update, a message requiring factual lookup, and a message in a language other than English. Pocket's win+tie rates were 18%, 9% and 6% respectively. The model has little world knowledge and no multilingual training; it is a drafting instrument, not an assistant. In the Autopilot product, requests outside its scope are routed to a larger model by the coordination layer [10].

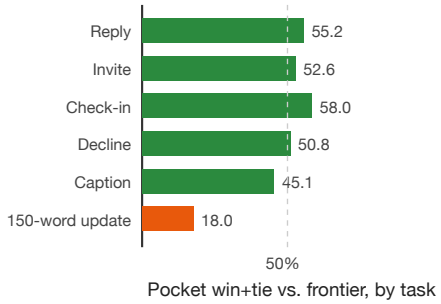


Figure 2. Pocket versus frontier models by task type (self + recipient conditions pooled). Pocket is at or above parity on all short social tasks and far below on long-form writing, by design.

6 Discussion

The result we find most interesting is the gap between stranger and self/recipient preferences. Strangers judge drafts as text; the people in the relationship judge them as *utterances from a specific person*. On the first criterion frontier models are slightly better. On the second, which is the one that matters for a drafting agent, a small model with the sender's voice is better. This suggests that quality for personal communication is poorly measured by generic text-quality judgements, and that evaluation should involve the people in the conversation whenever possible.

A practical consequence is privacy. Because Pocket is good enough on its own, the common case in Autopilot never leaves the device: the conversation, the style profile and the draft all stay local. We consider this a feature on par with quality.

7 Limitations

Pocket is English-only and trained on text from a handful of platforms whose users skew young and Western. The style profile may reproduce a sender's habits that they would prefer to change. Preference studies involved consenting beta members, who are early adopters of an AI-drafting product and may be more favourably disposed to its output than the general population. We release base weights but not the voice-conditioned checkpoint trained on member data.

8 Conclusion

A model that gives up almost everything except sounding like you, on short messages, can match the best available models at that task while running on your phone. Weights,

training recipe and the MLX/Metal inference stack: research.zavanet.com/pocket.

References

1. M. Abdin et al. Phi-3 technical report. *arXiv:2404.14219*, 2024.
2. Gemma Team. Gemma: Open models based on Gemini research and technology. *arXiv:2403.08295*, 2024.
3. L. Ben Allal et al. SmolLM2. *arXiv:2502.02737*, 2025.
4. Y. Li et al. Learning author-aware language models. *EMNLP*, 2021.
5. T. Brown et al. Language models are few-shot learners. *NeurIPS*, 2020.
6. S. Zhang et al. Personalizing dialogue agents. *ACL*, 2018.
7. MLC team. MLC-LLM: Universal LLM deployment engine. 2023.
8. A. Hannun et al. MLX: Efficient and flexible machine learning on Apple silicon. 2023.
9. R. Okafor et al. The Social Manipulation Benchmark. *Zavanet Research TR ZR-2026-02*, 2026.
10. D. Narayanan et al. Relay: Lightweight hand-off protocols for teams of small language agents. *Zavanet Research TR ZR-2026-03*, 2026.