LISTENING BY THE RIGHT AMOUNT: CALIBRATED RESPONSES TO USER REQUESTS IN TEXT-PROFILE RECOMMENDERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Natural-language user profiles make recommenders scrutable: users can read what the system believes about them and correct it in words. A correction has a direction and an amount, yet evaluations and training objectives address only the direction, and nothing specifies how much a recommendation list should change. We measure thousands of requests, adding, withdrawing or negating a preference, on users the model never saw, against a principled yardstick for the amount: a user who states a preference should be treated like the *reference users* who already hold it. Fine-tuned rankers ignore requests (8% of a negated genre's share removed on MovieLens-1M, 0% on tag-level attributes held out from training). Training them to obey fixes the direction but not the amount: after "*I also love* $\langle$*genre*$\rangle$ *movies*", 98% of the top 20 carry that genre where reference users get 31%, and a withdrawal is executed as a negation; we show that an objective rewarding only the direction never prefers the right amount to a larger one, an error that success metrics cannot see. We propose *calibrated edit distillation* (CED). During training, a teacher given each request's kind and attribute shifts the logits of the items it concerns by the log of the population odds ratio, which under a softmax choice model moves a neutral user's odds exactly to those of the reference users; the model learns to produce this shift from the request's text alone, and at the optimum only the requested items move. A 125M-parameter CED model removes negated genres completely and 85% of the share of held-out tags it never saw edited, and with a scale selected on validation users it lands within three points of the reference users on added and withdrawn genres and matches its teacher on genres. It follows 36 independently written phrasings once trained on 60 per kind and costs no measurable accuracy, without a separate parser or a generation step.

1 INTRODUCTION

Recommenders built on large language models increasingly describe each user by a short natural-language profile (Wu et al., 2024). The appeal is scrutability: the user can read what the system believes about them and correct it in words (Balog et al., 2019; Radlinski et al., 2022), in the way a concept bottleneck lets a person intervene on an interpretable intermediate representation (Koh et al., 2020). A growing line of text-profile recommenders builds on this promise: they learn to write better profiles (Zhou et al., 2024; Gao et al., 2025), align profiles with collaborative embeddings (Penaloza et al., 2025) or recommend from them directly (Mysore et al., 2023; Ramos et al., 2024; Öncel et al., 2026). They rest on a premise that is rarely measured: after the user says "*I no longer enjoy Action movies*", the recommendations change the way the user intended. Controllability evaluations check whether an edited profile moves the list in the requested direction (Mysore et al., 2023; Penaloza et al., 2025), and the two studies that measured the response of fine-tuned or pretrained rankers found it weak, on one hand-edited profile (Mamié & van den Bergh, 2026) and without fine-tuning (Zhou et al., 2026).

Responding at all is only half of the requirement: a recommender that responds must also respond by the right amount. Instructable and controllable recommenders (Zhang et al., 2023; Lu et al., 2024; Paischer et al., 2025) and critiquing-based recommenders (Chen & Pu, 2011; Wu et al., 2019; Luo et al., 2020) train and evaluate whether the requested attribute rises or falls, and CTRL-Rec (Carroll

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

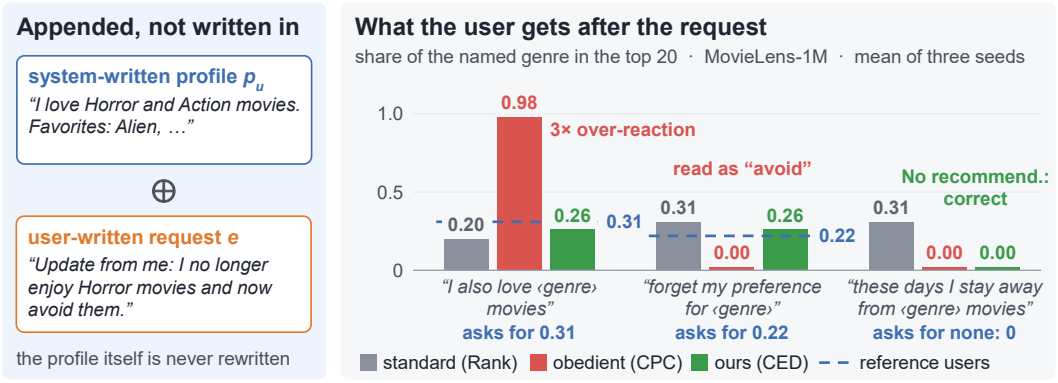


Figure 1: Left: the profile and the request are kept apart. Right: share of the named genre in the top 20 after three kinds of request (MovieLens-1M, three seeds) and the share each request asks for. A standard model ignores requests, an obedient one over-reacts and reads a withdrawal as an aversion, and ours (CED, $\alpha = 1$) lands near the *reference users*.

et al., 2025) adds a request signal to a recommender's scores with a weight set by hand. None of them says how far the list should move. A model that answers "*I also love Westerns*" with a list of Westerns scores perfectly on a success metric, much as a classifier can be accurate and badly calibrated (Guo et al., 2017). Calibrated recommendation (Steck, 2018) matches a list's composition to the user's history, but no criterion checks whether the change a request causes matches what it asks for. This paper asks how much a text-profile recommender should listen, measures how much it does, and trains it to listen by the right amount.

We apply thousands of requests, adding a preference, withdrawing one or turning one into an aversion, to users the model never saw, and measure what happens to the items a request is about and, separately, to the items it is not about (§2). The yardstick for the amount is the *reference users*: a user who now states a preference should be treated like the users whose own profile already states it. Three findings follow, the first and the last shown in Figure 1:

- **Text-profile recommenders do not listen.** A request appended in plain language removes $8 \pm 10\%$ of the negated genre's share of the top 20 on MovieLens-1M and nothing on tags held out from training. Rewriting the profile instead removes a fifth to a third, under rating regression and ranking losses alike (§4.2).
- **Written into the profile, obedience costs accuracy.** A contrastive objective makes the model obey rewritten profiles, but only by trusting every stated preference more, and NDCG@10 falls on every seed (§4.3).
- **Kept apart from the profile, obedience over-reacts.** After "*I also love* ⟨*genre*⟩ *movies*", 98% of the top 20 carry that genre where the reference users get 31%, and "*forget my preference for Horror*" empties the list of Horror exactly like a negation. Success metrics reward this, and so does any objective that rewards only the direction of a response (Proposition 1).

The last finding shows what is missing: a target for the amount. We derive one from the reference users: a request shifts the logits of the items it is about by the log odds ratio between users who state the preference and users who do not, which under a softmax choice model takes a neutral user exactly to the odds of users who state the preference while keeping every user's own starting point (Proposition 2). The shift can be computed only when the kind and attribute of the request and the attributes of the items are known, as they are for training requests but not for a request a user types. We therefore treat them as privileged information (Vapnik & Vashist, 2009): a teacher computes the calibrated response on the model's own scores, and *calibrated edit distillation* (CED) distils it into a model that reads only the text of the request (Lopez-Paz et al., 2016), as privileged features are distilled in industrial recommenders (Xu et al., 2020). The optimum reproduces the teacher and leaves the order of all other items unchanged, so locality needs no penalty of its own (Proposition 3). Applied to the standard ranker, the teacher is what a perfect parser followed by the shift would produce, so the gap between CED and its teacher measures the cost of reading a request from its text alone.

The gap turns out to be small. CED removes 100% of a negated genre and 85% of a negated tag never edited in training (against 0% for the baseline), reaches its teacher on genres, negates under unseen phrasings and, trained on 60 phrasings per kind, follows all three kinds under them, and costs no measurable accuracy (§4.4, §4.5). With the scale selected on validation users it lands within three points of the reference users on added and withdrawn genres, and on users whose taste really changed it puts a newly liked genre at 0.53 of the top 20, next to their own later share of 0.55. Baselines fall short: at the same scale, parsers fail on unseen phrasings or misread 13% of the requests, and CTRL-Rec's single request weight either over-reacts or fails to negate (§4.5).

Our contributions are as follows:

- **A systematic test** on attributes and phrasings unseen in training: fine-tuned text-profile recommenders do not follow requests, and training them to obey over-reacts.
- **A yardstick for the amount**, the reference users, which exposes over-reaction invisible to success metrics, with a proposition showing that an objective rewarding only the direction never prefers the right amount to a larger one.
- **Calibrated edit distillation**, which distils a privileged teacher built on this yardstick into a 125M-parameter model that reads the request in the same forward pass that scores the items, without a separate parser, and transfers to held-out tags and unseen phrasings.

2 Measuring Whether a Recommender Listens

Text-profile recommenders and requests. A user u is represented only by a natural-language profile p_u, written by the system from the user's interaction history; an item i by a short text x_i. A scorer $s_\theta(p, x_i) \in \mathbb{R}$ ranks candidates for any profile p; no user identifier or embedding enters the model, so the recommender is *inductive* and all our evaluations use users it was never trained on. A *request* $e = (k, a)$ has a kind $k \in \{\text{INSERT}, \text{DELETE}, \text{NEGATE}\}$ and an attribute a (a genre or a tag): INSERT adds a preference ("*I also love Horror movies*"), DELETE withdraws one and asks for a to be treated neutrally, not erased as in attribute unlearning (Chen et al., 2025) ("*forget my preference for Horror*"), NEGATE turns it into an aversion ("*I now avoid Horror movies*"). A request reaches the model either by *rewriting* the profile text or, with the profile as written, *appended* as a marked sentence ("*Update from me: I no longer enjoy Horror movies and now avoid them.*"); $p' = e(p)$ denotes the profile with the request applied in either form.

What a request should change. Each attribute a induces a target set G_a of candidate items that have it and a neutral set N_a of items that clearly lack it, excluding items the user already interacted with; $\pi_K(p, a)$ is the fraction of the top-K list under profile p that lies in G_a ($K = 20$). For the requests of one kind, let $\bar{\pi}$ and $\bar{\pi}^0$ be the mean of $\pi_K(p', a)$ and of $\pi_K(p, a)$, after and before the request. The *request success* of a NEGATE is the fraction of the attribute's share that disappears,

$$\text{succ} = 1 - \bar{\pi}_{\text{NEG}} \,/\, \bar{\pi}^0_{\text{NEG}}. \tag{1}$$

For INSERT and DELETE neither a saturated nor an emptied list is right, so the *reference users* set the amount: after INSERT, the user should receive the share π_a^{like} that users whose own profile states a receive; after DELETE, the share π_a^{neu} of users whose profile does not mention a; after NEGATE, none, since a negation is a constraint, not a preference to be weighed. The *response deviation* adds the two gaps,

$$\text{dev} = \left|\bar{\pi}_{\text{INS}} - \pi^{\text{like}}\right| + \left|\bar{\pi}_{\text{DEL}} - \pi^{\text{neu}}\right|, \tag{2}$$

computed per seed and averaged, with reference shares pooled over the reference user–attribute pairs of the evaluation set. Reference shares are computed once per seed from what the standard ranker (RANK, §4.1) shows to the reference users, so no method can move its own yardstick (Appendix C). This default is consistent, since a request counts as much as the same statement already counts in a profile, and it is measured rather than assumed; whether it matches where users who really change their taste end up is tested in §4.4. *Locality*, as in model editing (Meng et al., 2022; Mitchell et al., 2022), is the Spearman correlation between $s_\theta(p, \cdot)$ and $s_\theta(p', \cdot)$ on N_a (1: unrelated items untouched); we measure it directly because recommender rank lists can move under small changes of the input (Oh et al., 2022). *Accuracy* is NDCG@10 of held-out positives (ratings 4–5) under the profile as written, over a fixed pool of warm candidates (at least 20 training ratings); as a sampled metric (Krichene & Rendle, 2020) it is used only for paired comparisons (Recall@20: Appendix C).

To see whether obedience is bought by trusting profiles more (§4.3), we also report the *native lift*: the top-20 share of an attribute among users whose profile states it, divided by the attribute's prevalence.

Because a model could memorise one switch per attribute, a shortcut in the sense of Geirhos et al. (2020), a fixed subset of attributes is never a request target in training and the main metrics are reported separately for these *unseen* attributes; likewise, training and test request sentences come from disjoint sets (*unseen phrasings*).

3 METHOD: CALIBRATED EDIT DISTILLATION

CED rests on three decisions: the request is kept apart from the profile (§3.1), obedience is rejected as the target (§3.2), and the target is a data-sized shift that the model learns to produce from text (§3.3).

3.1 KEEP THE PROFILE AS WRITTEN; APPEND THE REQUEST

A system-written profile is *observational* evidence: how much "I love Horror movies" should count is an empirical question that training data answers. A user's request is an *instruction*, and should hold even when the rest of the profile suggests otherwise. Written into the profile, a request is indistinguishable from evidence, and a model trained to obey it does so by trusting every such statement more (§4.3). We therefore keep the two apart: the profile is left unchanged and each request is appended as a separate sentence prefixed by "*Update from me:*", a convention the model does not key on (Appendix O). Because a factual profile never contains such a sentence, following a request never requires weighing a factual statement differently; free-form rewrites of the profile are outside our scope.

3.2 OBEDIENCE ALONE IS THE WRONG TARGET

The obvious way to teach a model to obey is a contrastive term on minimally different inputs, the remedy for text encoders that read inputs as bags of words (Yuksekgonul et al., 2023): for a training positive i^+ of user u, pick an attribute a of i^+ and a request e that supports (INSERT) or contradicts (DELETE, NEGATE) it, and require the score of i^+ to move in the corresponding direction relative to unrelated items:

$$\mathcal{L}_{\mathrm{cpc}} = \mathrm{softplus}\big(-\varsigma_k\,(\Delta_{i^+} - \overline{\Delta}_U)\big) + \mu\,\mathrm{Var}_{j\in U}[\Delta_j], \tag{3}$$

where $\Delta_i = s_\theta(e(p_u), x_i) - s_\theta(p_u, x_i)$ is the score change of item i under the request, the sign ς_k is $+1$ for INSERT and -1 otherwise, U are the sampled negatives that lack a, $\overline{\Delta}_U$ is their mean change, and the second term penalises any non-uniform change among them ($\mu = 1$). We call this *counterfactual profile contrast* (CPC) and add it to $\mathcal{L}_{\mathrm{rank}}$ with weight λ. It makes the model obey, even on unseen attributes and phrasings (Table 2), but it cannot stop:

Proposition 1 (Direction-only objectives saturate)**.** *Let a request term depend on the response only through the signed relative response $x = \varsigma_k(\Delta_{i^+} - \overline{\Delta}_U)$, via a nonincreasing loss $\ell(x)$. (i) $\ell(x) \le \ell(x^\star)$ for every $x \ge x^\star$: no response, however large, costs more than a calibrated one $x^\star$; if ℓ is strictly decreasing, as the softplus of Eq. (3) is ($\ell'(x) = -\sigma(-x) < 0$), it has no minimiser. (ii) If a request adds d to the scores of the items of G_a and leaves the others, π_K is nondecreasing in d and equals $\min(1, |G_a|/K)$ for d large enough and 0 for d small enough (when at least K candidates lack a).*

Because the variance term of Eq. (3) vanishes whenever unrelated items move together, a larger x is a larger shift of G_a against the rest, and by (ii) the list saturates (proofs: Appendix A): an insertion is pulled towards a list made of a and a withdrawal towards none of it, exactly like a negation (§4.3). A success metric, maximised at the same point, cannot see this.

3.3 CALIBRATED EDIT DISTILLATION

Section 2 fixed the amount at the reference users' share; we now turn it into a logit shift. For a training attribute a, let ρ_a^{like} and ρ_a^{neu} be the fraction of held-out positives in G_a among training users

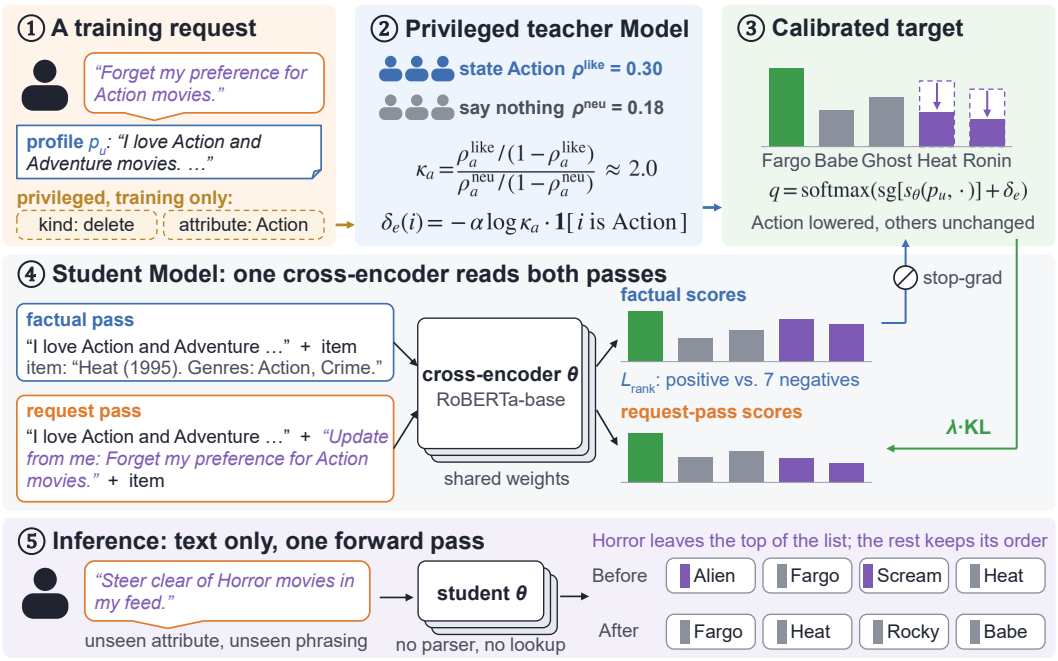


Figure 2: CED on one training example (a *delete* of Action) and one test-time request. (1) The request is appended to the unchanged profile; its kind and attribute are known only in training. (2) The privileged teacher sizes the shift by the odds ratio κ_a (Eq. (5)) and (3) applies it to the model's own factual scores, gradient stopped. (4) One cross-encoder scores the candidates with and without the request; the request pass is distilled towards the target (Eq. (6)). (5) At test time it reads a request on an unseen attribute, in an unseen phrasing.

whose profile states a and among those whose profile says nothing about it, and $\kappa_a = \frac{\rho_a^{\text{like}}/(1-\rho_a^{\text{like}})}{\rho_a^{\text{neu}}/(1-\rho_a^{\text{neu}})}$ their odds ratio (clipped to $[1.2, 20]$, Appendix C). The following identity turns κ_a into a logit shift.

Proposition 2 (A log-odds shift turns neutral odds into native odds)**.** *For scores s over a candidate set $\mathcal{C}$, let $q(s) = \sum_{i \in G_a} e^{s_i} / \sum_{j \in \mathcal{C}} e^{s_j}$ be the softmax probability of choosing an item of G_a. Adding d to the scores of G_a multiplies its odds by e^d:*

$$\frac{q(s + d\,\mathbf{1}_{G_a})}{1 - q(s + d\,\mathbf{1}_{G_a})} = e^d \frac{q(s)}{1 - q(s)}. \tag{4}$$

The unnormalised mass of G_a is multiplied by e^d, that of the rest is unchanged, and the normaliser cancels in the odds. With $d = \log \kappa_a$ (inside its clipping range) a user at the neutral odds is taken exactly to the native odds, whatever else the profile says. The shift acts on each user's own odds, so a requesting user keeps an individual starting point instead of being reset to the population mean, and shifts compose: an insertion followed by a withdrawal is the identity. We therefore *define* the intended effect of a request as a logit shift on the items it is about and nothing else,

$$\delta_e(i) = \mathbf{1}[i \in G_a] \cdot \begin{cases} +\alpha \log \kappa_a & e = \text{INSERT}(a), \\ -\alpha \log \kappa_a & e = \text{DELETE}(a), \\ -c & e = \text{NEGATE}(a), \end{cases} \tag{5}$$

where $\alpha = 1$ is the shift of Proposition 2 (the scale is discussed below) and $c = 8$ multiplies the odds of G_a by $e^{-8} \approx 3 \cdot 10^{-4}$ ($c = 4$ behaves identically, Appendix J): a user who says they now avoid an attribute asks for a constraint, so the target share of a negated attribute is zero. Computing δ_e requires the kind and attribute of the request and the attributes of the candidates. They are known for every training request, which is generated from them, but not for a request a user types, so the *privileged teacher* applies δ_e to the model's own factual scores and the model learns to produce the teacher's response from the text alone. For a training group $\mathcal{C}$ of candidates (the positive, seven

Table 1: Datasets. Users are split 80/10/10 into training, validation and test users; the first 80% of each user's ratings form the profile and the rest are targets. Main results use 200 test users per dataset and a fixed pool of warm candidates. Unseen attributes are never request targets in training.

Dataset	Users	Items	Attributes	Seen / unseen	Candidates	Item text
MovieLens-1M	6,040	3,686	18 genres	12 / 6	500	title, genres
MovieLens-20M + Tag Genome	8,000	6,718	638 tags	474 / 164	1,000	title, genres, 8 tags

sampled negatives and two unrated items of G_a), the teacher's distribution is the model's own factual distribution, gradient stopped ($\mathrm{sg}[\cdot]$), after the shift:

$$\mathcal{L} = \mathcal{L}_{\mathrm{rank}}(p_u) + \lambda\,\mathrm{KL}\Big(\mathrm{softmax}_{\mathcal{C}}\big(\mathrm{sg}[s_\theta(p_u,\cdot)] + \delta_e\big)\;\Big\|\;\mathrm{softmax}_{\mathcal{C}}\big(s_\theta(e(p_u),\cdot)\big)\Big). \tag{6}$$

We call this *calibrated edit distillation* (CED); Algorithm 1 (Appendix A) lists one training step. In form, the teacher is a logit adjustment (Menon et al., 2021) and the objective a rule distillation (Hu et al., 2016); what is new is the rule, a request-specific shift read off the population. One request in a training phrasing is sampled per group, independently of the positive (a DELETE or NEGATE of a stated attribute or an INSERT of an unmentioned one; Appendix D), and the two items of G_a enter only the distillation term (as negatives in $\mathcal{L}_{\mathrm{rank}}$ they cost accuracy, Appendix H).

Proposition 3 (The optimum of CED)**.** *On a group $\mathcal{C}$, the KL term of Eq. (6) vanishes if and only if $\Delta_i = \delta_e(i) + b$ for all $i \in \mathcal{C}$ and some constant b. Then (i) the items of G_a move by exactly δ_e relative to every other item, and (ii) all other items move by the same b, so their order is unchanged. The gradient with respect to the request-pass scores z is $\mathrm{softmax}(z) - \mathrm{softmax}(t)$, with t the shifted target: zero at the target and pointing back towards it beyond.*

Proposition 3 has three consequences (proofs in Appendix A): locality comes from the target, with no penalty like the variance term of Eq. (3); unlike the softplus, the distillation stops pulling once the shift is reached; and the fixed point is the privileged teacher itself, which a system without CED can only approximate with a separate parser (§4.5).

The scale of the shift. Proposition 2 is exact for the probability of choosing an item of G_a, whereas we evaluate its share of the top 20, which Proposition 1(ii) shows to be monotone in the shift but not equal to that probability. One scalar α on $\log \kappa_a$, chosen with λ on validation users (Appendix I), absorbs the difference; $\alpha = 1$ is reported alongside.

4 EXPERIMENTS

We ask whether fine-tuned text-profile recommenders follow requests (**RQ1**), what training them to obey does (**RQ2**), whether CED responds by the right amount (**RQ3**), and whether it transfers and what it takes otherwise (**RQ4**).

4.1 SETUP

Data. We use MovieLens (Harper & Konstan, 2015) because, unlike most public interaction datasets, it attaches reasonably complete attributes to every item, which the target items of a request and its reference users require (Table 1). The unseen tags of MovieLens-20M are whole correlation clusters, so none has a near-synonym among the edited ones (Appendices D and L). Profiles are rendered from a fixed template (stated likes, at most one or two aversions, five favourite titles) so that every statement has an unambiguous referent; LLM-written profiles are studied in §4.5.

Models and evaluation. The scorer is a RoBERTa-base cross-encoder (Liu et al., 2019) over the concatenated profile and item text with a scalar head, trained for three epochs of AdamW at 3×10^{-5} (RANK's validation-best $2\cdot10^{-5}$ is inside seed noise, Table 4) with 16 groups per batch and $\lambda = 1$ unless selected; CED's α and λ were chosen on validation users (Appendix I). Each test user receives every applicable request: DELETE and NEGATE of up to three stated attributes (two on MovieLens-20M) and one INSERT of an unmentioned one, on seen and unseen attributes, in training

Table 2: Requests appended to MovieLens-1M profiles (200 test users, 500 candidates). $\Delta\%$: paired NDCG@10 change against RANK (per seed and per kind: Table 4); deviation: Eq. (2). Seen / unseen: genres edited / never edited in training; "unseen phr.": negation success under the two held-out sentences per kind (‡under the six of the extended phrasing set). Privileged-teacher and parser rows add Eq. (5) to the seed-0 RANK scores; the teacher is told each request's kind and attribute ("given"). CTRL-Rec: Appendix G. †request rewritten into the profile (RANK under that protocol: 0.19 / 0.21). Trained rows: three seeds; teacher, parser and CTRL-Rec rows: seed 0.

			request success (NEGATE) ↑			response deviation ↓		
Method	NDCG@10	$\Delta\%$	seen	unseen	unseen phr.	seen	unseen	unrelated ↑
RANK	0.375 ±0.019	0.0	0.08 ±0.10	0.14 ±0.13	0.09 ±0.08	0.19 ±0.02	0.16 ±0.03	0.983 ±0.002
CPC, profile rewritten†	0.353 ±0.003	-5.9 ±5.0	0.96 ±0.03	0.52 ±0.09	–	–	–	0.997 ±0.000
CPC, request appended	0.376 ±0.012	+0.3 ±5.4	1.00 ±0.00	1.00 ±0.00	1.00 ±0.00	0.90 ±0.00	0.89 ±0.01	0.998 ±0.000
CPC bounded	0.368 ±0.009	-1.9 ±4.7	1.00 ±0.00	1.00 ±0.00	1.00 ±0.00	0.09 ±0.01	0.08 ±0.01	0.997 ±0.000
CED ($\alpha = 1$)	0.371 ±0.012	-1.1 ±4.9	1.00 ±0.00	1.00 ±0.00	0.92 ±0.12	0.09 ±0.02	0.09 ±0.01	0.994 ±0.000
CED ($\alpha = 2$, validation-selected)	0.371 ±0.011	-1.0 ±5.6	1.00 ±0.00	1.00 ±0.00	0.87 ±0.10	0.06 ±0.01	0.02 ±0.02	0.990 ±0.000
Privileged teacher ($\alpha = 2$; seed 0)	= RANK	0.0	1.00	1.00	given	0.06	0.04	1.000
Learned parser + shift ($\alpha = 2$; seed 0)	= RANK	0.0	1.00	1.00	0.76	0.06	0.04	1.000
Qwen3-14B parser + shift ($\alpha = 2$; seed 0)	= RANK	0.0	1.00	0.99	1.00‡	0.14	0.09	1.000
CTRL-Rec re-scoring (seed 0)	= RANK	0.0	0.94	0.74	0.90	0.44	0.25	0.745

and unseen phrasings (three training and two unseen sentences per kind unless stated otherwise; Appendix K), and on MovieLens-1M also as rewritten profiles. Numbers with a $\pm$ are means and SDs over three seeds; accuracy differences are paired per user within a seed, and RANK's own seed-to-seed SD (0.019 NDCG@10, about 5%) is the yardstick for a measurable difference.

Baselines. No published system trains the scorer of an inductive text-profile recommender to follow free-text requests: instructable recommenders identify users and items by IDs, critiquing and controllable recommenders act on user embeddings or need their own labelled requests, and text-profile systems leave the reader untouched. We therefore implement the mechanism of each family on the same ranker, users, candidates and requests, which isolates the mechanism from differences in data and backbone: standard training (REG, the rating regression of Ramos et al. (2024); REG+NEG, with one sampled negative per rating; and RANK, a sampled-softmax ranking loss with seven negatives (Jean et al., 2015; Rendle et al., 2009)), direction-only obedience training (CPC), our calibrated target imposed without distillation (a bounded margin, CPC bounded, and counterfactual augmentation, CFA), re-scoring a fixed ranker (CTRL-Rec (Carroll et al., 2025), which adds a learned request–item approval with one weight, and parse-then-shift pipelines with a keyword, learned or LLM parser) and prompting (a zero-shot LLM ranker (Hou et al., 2024) in the SteerEval setting (Zhou et al., 2026)). The privileged teacher, RANK plus Eq. (5) given each request's true kind and attribute, is what parse-then-shift produces with a perfect parser and the response CED is trained to reproduce.

4.2 RQ1: FINE-TUNED TEXT-PROFILE RECOMMENDERS DO NOT LISTEN

Takeaway 1. Under every standard objective, a request written into the profile removes at most a third of the negated genre, and an appended request almost nothing.

Appended requests are ignored (Table 2, RANK row): a negated genre loses $8 \pm 10\%$ of its share, an added one does not move, and on the held-out tags the response is 0% (Table 7). Standard training never contains a request, so the ranker reads one mostly as another mention of the genre: placed first, a negation even raises the negated genre's share (Appendix O). Rewriting the profile helps only a little, and not for the reason Mamié & van den Bergh (2026) proposed, that rating regression learns almost no user–item interaction: one sampled negative per rating quadruples the share of score variance due to that interaction, yet negation success stays at 0.29 (REG 0.32, RANK 0.23; Appendix E), and under all three objectives a negated genre keeps 68–77% of its share.

4.3 RQ2: DIRECTION-ONLY OBEDIENCE EITHER COSTS ACCURACY OR OVER-REACTS

Takeaway 2. Direction-only obedience training works only with the request kept apart from the profile, and then over-reacts by a factor of three, the saturation of Proposition 1.

Table 3: Users whose later ratings changed (MovieLens-1M test users, seed-0 checkpoints, seen genres). *Emergent*: a genre a absent from the profile makes up ≥ 0.25 of the later positives, ≥ 0.15 more than of the earlier ones (129 user–genre pairs; top-20 share before the request 0.32); request INSERT(a). *Lapsed*: a stated genre's later share is ≥ 0.15 below its earlier one (76 pairs; before 0.31); request DELETE(a). Gap: distance of the mean top-20 share to the users' mean later share (first row).

	emergent, INSERT		lapsed, DELETE	
Method	share	gap ↓	share	gap ↓
Users' own later share	0.55	–	0.14	–
RANK (ignores the request)	0.32	0.23	0.29	0.16
CPC, request appended (over-reacts)	1.00	0.45	0.00	0.14
Privileged teacher ($\alpha = 2$)	0.49	0.06	0.16	0.03
CED ($\alpha = 2$)	0.53	0.02	0.17	0.03

Written into the profile, requests make the model trust every statement more. Training CPC on rewritten profiles makes the model obey them (negation success from 0.19 to 0.96 ± 0.03 on seen genres and from 0.21 to 0.52 ± 0.09 on unseen ones; RANK, seed 0), but only by trusting stated genres more: without any request, the top-20 share of a stated genre rises from 0.35 to 0.48 (seed 0), the native lift from 2.32 ± 0.04 to 2.80 ± 0.09, and NDCG@10 falls on every seed (-10.3, -7.0, -0.4%; Table 4).

Appended requests keep the model's reading of the profile, and the model over-reacts. The same objective on appended requests reaches 1.00 success on seen and unseen genres and unseen phrasings, leaves unrelated items in place and keeps the native lift near the baseline (2.41 ± 0.07; NDCG@10 $+0.3 \pm 5.4\%$). It is still wrong. After "*I also love* ⟨*genre*⟩ *movies*" the genre's top-20 share is 0.98 where the reference users sit at 0.31, and after "*forget my preference for* ⟨*genre*⟩" it is 0.00, as after a negation, where users who say nothing sit at 0.22; the response deviation is 0.90, against 0.19 for the model that ignores requests (Table 2). A direction-only loss can only push a withdrawal away from a, like a negation; it has no way to say "neutral".

4.4 RQ3: CALIBRATED EDIT DISTILLATION LISTENS THE RIGHT AMOUNT

Takeaway 3. CED lands within three points of the reference users on added and withdrawn preferences, removes negated ones completely at no measurable accuracy cost, and, on genres, reaches its privileged teacher without being told what the request means.

With the scale selected on validation users ($\alpha = 2$, $\lambda = 0.3$; Appendix I; the first CED row of Table 2 uses $\alpha = \lambda = 1$) an added genre reaches 0.34 ± 0.01 (reference 0.31) and a withdrawn one 0.19 ± 0.01 (reference 0.22), on unseen genres 0.19 against 0.18 and 0.09 against 0.09: a response deviation of 0.06 and 0.02, against 0.90 for the obedient model and 0.19 for the baseline. At every scale at which it was trained, CED reaches its privileged teacher on genres (0.26 / 0.26 at $\alpha = 1$, 0.34 / 0.19 at $\alpha = 2$, within 0.004 at $\alpha = 1.5$; Appendix J), as at the training optimum of Proposition 3. Neither row costs measurable accuracy ($-1.1 \pm 4.9\%$ and $-1.0 \pm 5.6\%$ NDCG@10, within RANK's seed-to-seed spread), and both keep the model's reading of the factual profile (native lift 2.26 ± 0.09 and 2.30 ± 0.11 against 2.32 ± 0.04). The same target imposed as a bounded margin (CPC bounded) calibrates as well (0.09 / 0.08) but raises the native lift to 2.60 ± 0.06, and imposed through sampled positives (CFA, Table 4) it is seed-unstable on insertion (0.28–0.42); the distilled form avoids both.

Does the calibrated shift predict where users whose taste changed go? Table 3 looks at test users whose later ratings actually changed: 129 user–genre pairs in which a genre absent from the profile makes up 0.55 of the later positives (*emergent*), and 76 in which a stated genre drops to 0.14 (*lapsed*). Sent the matching request, the selected CED puts an emergent genre at 0.53 of the top 20 on average and a lapsed one at 0.17, within 0.02 and 0.03 of these users' later share, while the over-reacting model (0.45 and 0.14 away), the model that ignores the request (0.23 and 0.16) and the privileged teacher (0.06 and 0.03) are farther or equal. These users start above the population

share, and because the shift acts on each user's own odds (Proposition 2) it takes them to where users who really changed their taste go: the external check that §2 calls for.

4.5 RQ4: GENERALISATION, AND WHAT IT TAKES OTHERWISE

Takeaway 4. Negation transfers to held-out tags and unseen phrasings, all three kinds do with 60 training phrasings per kind, and negation and withdrawal transfer to LLM-written profiles and a decoder; parse-then-shift and re-scoring baselines fall short.

Tags never edited in training. On the 164 held-out MovieLens-20M tags (Table 7), a negated tag is removed with 0.85 ± 0.01 success (RANK 0.00 ± 0.02), a withdrawn tag settles at 0.061 ± 0.005 (reference 0.061), an added tag reaches 0.054 ± 0.003 against RANK's 0.037 (reference 0.070), and NDCG@10 changes by $+2.3 \pm 4.3\%$ ($\alpha = 1$). CPC over-reacts again (added tag 0.24 against 0.070; withdrawn tag wiped out).

Phrasings never seen in training. Trained on three sentences per kind, CED keeps 0.92 ± 0.12 negation success under its two unseen sentences, and the selected model also responds to added and withdrawn preferences under them (0.25 ± 0.01 and 0.19 ± 0.02; references 0.31 / 0.22). Trained on 60 sentences per kind (Appendix J), it reaches 0.29 ± 0.01 on added genres under unseen phrasings with negation success 1.00 ($-2.0 \pm 4.9\%$ NDCG@10), and under 36 further sentences written independently after training, sharing no content-word bigram with any training or test sentence, it negates at 1.00 and adds at 0.27 ± 0.02 (Table 11). A paraphrase of the genre name ("*scary movies*" for Horror, "*whodunits*" for Mystery) is negated with success 0.90 (single seed), and five behavioural probes show that the response is not string matching (Appendix O).

Other profiles and backbones. With free-text profiles written by Qwen3-14B (Yang et al., 2025, prompt in Appendix D), appended requests are again ignored. CED with the configuration selected on template profiles listens (negation 1.00; 0.93 under unseen phrasings) and puts an added genre at 0.27 and a withdrawn one at 0.19 (references 0.23 and 0.18) at -2.2% NDCG@10; a Qwen2.5-0.5B decoder with LoRA (Qwen Team, 2025; Hu et al., 2022) matches the encoder on negation and withdrawal (1.00; 0.27 against 0.22) at $+1.7\%$ (single seeds; Table 8).

Parse-then-shift, re-scoring and prompting. Without CED, a system must first recover what a request means: parse-then-shift pipelines read its kind and attribute with a separate parser and add Eq. (5) to RANK's scores, which reproduces the privileged teacher when the parse is right, as for a learned parser (RoBERTa, trained on the same phrasings) in the training phrasings (deviation 0.06 / 0.04 at the selected scale; Table 2, Appendix G). CED matches it there and keeps reading requests where parsers fail: a keyword parser written from the training phrasings parses none of the unseen negations, the learned parser negates 0.65 and 0.76 of the share under them at $\alpha = 1$ and 2, against 0.99 and 0.81 for CED on the same seed, and a zero-shot Qwen3-14B parser reads every phrasing but misreads 13% of the requests, mostly withdrawals, which leaves it at a response deviation of 0.14 against 0.06 for CED (Appendix G), at one 14B generation per request. CTRL-Rec (Carroll et al., 2025) adds a learned request–item approval to RANK's scores with one weight, and no weight serves all three kinds: at the weight chosen by CED's rule it negates (0.94) but over-fills an added genre (0.66 against 0.31) and disturbs unrelated items (0.75), and at the weight with the smallest deviation it negates 0.41 (Table 2; Appendix G). A 14B zero-shot *ranker* listens as well, at half the fine-tuned ranker's NDCG@10 (0.25 against 0.47 on the same users; Table 8).

5 CONCLUSION

A request to a recommender has a direction and an amount; fine-tuned text-profile rankers get neither, and obedience training gets the direction but overshoots the amount, since an objective that rewards only the direction never prefers the right amount to a larger one. The users who already hold a preference define the amount, and calibrated edit distillation learns it from a privileged teacher: a 125M-parameter model that reads the request in one forward pass responds by as much as the reference users receive, transfers to held-out tags and unseen phrasings, and costs no measurable accuracy. The same recipe applies wherever the effect a request should have can be computed during training.

AI USE STATEMENT

We used AI-based writing tools to help polish the text of this paper. The authors take full responsibility for its content.

REPRODUCIBILITY STATEMENT

The supplementary material contains runnable code for our proposed method CED. We will publicly release the code and all accompanying materials. Proofs are given in Appendix A; the preprocessing, attribute split and request phrasings in Appendices D, L and K; the hyperparameter selection rule in Appendix I; and compute, seeds and run times in Appendix N.

REFERENCES

Krisztian Balog, Filip Radlinski, and Shushan Arakelyan. Transparent, scrutable and explainable user models for personalized recommendation. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 265–274, 2019. doi: 10.1145/3331184.3331211.

Micah Carroll, Adeline Foote, Kevin Feng, Marcus Williams, Anca Dragan, W. Bradley Knox, and Smitha Milli. CTRL-Rec: Controlling recommender systems with natural language, 2025. URL https://arxiv.org/abs/2510.12742. Bi-Align workshop at ICLR 2025.

Chaochao Chen, Yizhao Zhang, Yuyuan Li, Jun Wang, Lianyong Qi, Xiaolong Xu, Xiaolin Zheng, and Jianwei Yin. Post-training attribute unlearning in recommender systems. *ACM Transactions on Information Systems*, 43(1):1–28, 2025. doi: 10.1145/3701987.

Li Chen and Pearl Pu. Critiquing-based recommenders: survey and emerging trends. *User Modeling and User-Adapted Interaction*, 22(1-2):125–150, 2011. doi: 10.1007/s11257-011-9108-6.

Zhaolin Gao, Joyce Zhou, Yijia Dai, and Thorsten Joachims. LangPTune: Optimizing language-based user profiles for recommendation. In *Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM)*, 2025. doi: 10.1145/3746252.3761369. URL https://arxiv.org/abs/2410.18870.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, November 2020. ISSN 2522-5839. doi: 10.1038/s42256-020-00257-z. URL http://dx.doi.org/10.1038/s42256-020-00257-z.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, pp. 1321–1330, 2017.

F. Maxwell Harper and Joseph A. Konstan. The MovieLens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems*, 5(4):1–19, December 2015. ISSN 2160-6463. doi: 10.1145/2827872. URL http://dx.doi.org/10.1145/2827872.

Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian McAuley, and Wayne Xin Zhao. Large language models are zero-shot rankers for recommender systems. In *Advances in Information Retrieval (ECIR)*, Lecture Notes in Computer Science, pp. 364–381, 2024. doi: 10.1007/978-3-031-56060-6_24.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://arxiv.org/abs/2106.09685.

Zhiting Hu, Xuezhe Ma, Zhengzhong Liu, Eduard Hovy, and Eric Xing. Harnessing deep neural networks with logic rules. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2016. doi: 10.18653/v1/P16-1228.

Sébastien Jean, Kyunghyun Cho, Roland Memisevic, and Yoshua Bengio. On using very large target vocabulary for neural machine translation, 2015. URL https://arxiv.org/abs/1412.2007.

Divyansh Kaushik, Eduard Hovy, and Zachary C. Lipton. Learning the difference that makes a difference with counterfactually-augmented data. In *International Conference on Learning Representations (ICLR)*, 2020. URL https://arxiv.org/abs/1909.12434.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *International Conference on Machine Learning (ICML)*, 2020. URL https://arxiv.org/abs/2007.04612.

Walid Krichene and Steffen Rendle. On sampled metrics for item recommendation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, pp. 1748–1757. ACM, August 2020. doi: 10.1145/3394486.3403226. URL http://dx.doi.org/10.1145/3394486.3403226.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A robustly optimized BERT pretraining approach, 2019. URL https://arxiv.org/abs/1907.11692.

David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik. Unifying distillation and privileged information. In *International Conference on Learning Representations (ICLR)*, 2016. URL https://arxiv.org/abs/1511.03643.

Wensheng Lu, Jianxun Lian, Wei Zhang, Guanghua Li, Mingyang Zhou, Hao Liao, and Xing Xie. Aligning large language models for controllable recommendations. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8159–8172, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.443. URL https://aclanthology.org/2024.acl-long.443/.

Kai Luo, Scott Sanner, Ga Wu, Hanze Li, and Hojin Yang. Latent linear critiquing for conversational recommender systems. In *Proceedings of The Web Conference 2020*, pp. 2535–2541, 2020. doi: 10.1145/3366423.3380003.

Noah Mamié and Laurin van den Bergh. Reproducing transparent and scrutable recommendations: Exploring open-weight models via natural-language user profiles, 2026. URL https://arxiv.org/abs/2609.19831. BlackBoxNLP workshop at EMNLP 2026.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. URL https://arxiv.org/abs/2202.05262.

Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. In *International Conference on Learning Representations (ICLR)*, 2021. URL https://arxiv.org/abs/2007.07314.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D. Manning. Fast model editing at scale. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://arxiv.org/abs/2110.11309.

Sheshera Mysore, Mahmood Jasim, Andrew Mccallum, and Hamed Zamani. Editable user profiles for controllable text recommendations. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, pp. 993–1003. ACM, 2023. doi: 10.1145/3539618.3591677. URL http://dx.doi.org/10.1145/3539618.3591677.

Sejoon Oh, Berk Ustun, Julian McAuley, and Srijan Kumar. Rank list sensitivity of recommender systems to interaction perturbations. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management (CIKM)*, pp. 1584–1594, 2022. doi: 10.1145/3511808.3557425.

Fırat Öncel, Jihoon Jeong, Emiliano Penaloza, Mirco Ravanelli, Laurent Charlin, and Cem Subakan. Controllable and content-based recommendations, 2026. URL https://arxiv.org/abs/2607.20938.

Fabian Paischer, Liu Yang, Linfeng Liu, Shuai Shao, Kaveh Hassani, Jiacheng Li, Ricky Chen, Zhang Gabriel Li, Xiaoli Gao, Wei Shao, Xue Feng, Nima Noorshams, Sem Park, Bo Long, and Hamid Eghbalzadeh. Preference discerning with llm-enhanced generative retrieval, 2025. URL https://arxiv.org/abs/2412.08604.

Emiliano Penaloza, Olivier Gouvert, Haolun Wu, and Laurent Charlin. TEARS: Text representations for scrutable recommendations. In *Proceedings of the ACM Web Conference (WWW)*, 2025. doi: 10.1145/3696410.3714948. URL https://arxiv.org/abs/2410.19302.

Qwen Team. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Filip Radlinski, Krisztian Balog, Fernando Diaz, Lucas Dixon, and Ben Wedin. On natural language user profiles for transparent and scrutable recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, pp. 2863–2874. ACM, 2022. doi: 10.1145/3477495.3531873. URL http://dx.doi.org/10.1145/3477495.3531873.

Jerome Ramos, Hossein A. Rahmani, Xi Wang, Xiao Fu, and Aldo Lipani. Transparent and scrutable recommendations using natural language user profiles. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13971–13984, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.753. URL https://aclanthology.org/2024.acl-long.753/.

Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. BPR: Bayesian personalized ranking from implicit feedback. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2009. URL https://arxiv.org/abs/1205.2618.

Harald Steck. Calibrated recommendations. In *Proceedings of the 12th ACM Conference on Recommender Systems*, RecSys '18, pp. 154–162. ACM, 2018. doi: 10.1145/3240323.3240372. URL http://dx.doi.org/10.1145/3240323.3240372.

Vladimir Vapnik and Akshay Vashist. A new learning paradigm: Learning using privileged information. *Neural Networks*, 22(5–6):544–557, 2009. doi: 10.1016/j.neunet.2009.06.042.

Ga Wu, Kai Luo, Scott Sanner, and Harold Soh. Deep language-based critiquing for recommender systems. In *Proceedings of the 13th ACM Conference on Recommender Systems*, pp. 137–145, 2019. doi: 10.1145/3298689.3347009.

Likang Wu, Zhi Zheng, Zhaopeng Qiu, Hao Wang, Hongchao Gu, Tingjia Shen, Chuan Qin, Chen Zhu, Hengshu Zhu, Qi Liu, Hui Xiong, and Enhong Chen. A survey on large language models for recommendation. *World Wide Web*, 27(5):60, 2024. doi: 10.1007/s11280-024-01291-2.

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. C-Pack: Packed resources for general Chinese embeddings. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 641–649, 2024. doi: 10.1145/3626772.3657878.

Chen Xu, Quan Li, Junfeng Ge, Jinyang Gao, Xiaoyong Yang, Changhua Pei, Fei Sun, Jian Wu, Hanxiao Sun, and Wenwu Ou. Privileged features distillation at Taobao recommendations. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 2590–2598, 2020. doi: 10.1145/3394486.3403309.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui

Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? In *International Conference on Learning Representations (ICLR)*, 2023. URL https://arxiv.org/abs/2210.01936.

Junjie Zhang, Ruobing Xie, Yupeng Hou, Wayne Xin Zhao, Leyu Lin, and Ji-Rong Wen. Recommendation as instruction following: A large language model empowered recommendation approach, 2023. URL https://arxiv.org/abs/2305.07001.

Joyce Zhou, Yijia Dai, and Thorsten Joachims. Language-based user profiles for recommendation, 2024. URL https://arxiv.org/abs/2402.15623. LLM-IGS workshop at WSDM 2024.

Joyce Zhou, Weijie Zhou, Doug Turnbull, and Thorsten Joachims. SteerEval: A framework for evaluating steerability with natural language profiles for recommendation, 2026. URL https://arxiv.org/abs/2601.21105.

A Proofs and the training step

Algorithm 1 One CED training step (Eq. (6)).

Require: batch of training users u with profile p_u and a held-out positive i^+; odds ratios κ_a; scale α; negation shift c; weight λ

for each user u in the batch **do**
  sample seven unrated negatives $\mathcal{N}$ and a request $e = (k, a)$: DELETE or NEGATE of a stated attribute, or INSERT of an unmentioned one, in a training phrasing
  sample two unrated items $A \subset G_a$; $\quad \mathcal{C} \leftarrow \{i^+\} \cup \mathcal{N} \cup A$
  $f \leftarrow s_\theta(p_u, \cdot)$ on $\mathcal{C}$ ▷ factual pass
  $\ell_{\text{rank}} \leftarrow -\log \operatorname{softmax}(f|_{\{i^+\}\cup\mathcal{N}})_{i^+}$ ▷ A stays out of the ranking loss
  $t \leftarrow \operatorname{sg}[f] + \delta_e$ ▷ Eq. (5)
  $z \leftarrow s_\theta(e(p_u), \cdot)$ on $\mathcal{C}$ ▷ request pass: profile + "Update from me: …"
  $\ell_u \leftarrow \ell_{\text{rank}} + \lambda \operatorname{KL}\big(\operatorname{softmax}(t) \,\|\, \operatorname{softmax}(z)\big)$
end for
update θ with AdamW on the batch mean of ℓ_u

Proposition 1. (i) For a nonincreasing ℓ and $x \geq x^\star$, $\ell(x) \leq \ell(x^\star)$ by definition; if ℓ is strictly decreasing, $\ell(x+1) < \ell(x)$ for every x, so no x attains the infimum. For the softplus, $\ell(x) = \log(1+e^{-x})$ and $\ell'(x) = -e^{-x}/(1+e^{-x}) = -\sigma(-x) < 0$. (ii) Write $s_i(d) = s_i + d\,\mathbf{1}[i \in G_a]$ and let $i \in G_a$ be in the top K at d, with $d' > d$ and ties broken by a fixed item order. An item j ranked above i at d' either lies in G_a, and then $s_j(d) - s_i(d) = s_j - s_i$ does not depend on d, so j was above i at d as well; or it lies outside G_a, and then $s_j \geq s_i + d' > s_i + d$, so again j was above i at d. The set of items above i does not grow, i stays in the top K, and $|\operatorname{top}_K \cap G_a|$ is nondecreasing in d. For $d > \max_{j\notin G_a} s_j - \min_{i\in G_a} s_i$ every item of G_a ranks above every other item and the top K holds $\min(K, |G_a|)$ of them; for $d < \min_{j\notin G_a} s_j - \max_{i\in G_a} s_i$ every item of G_a ranks below every other, and the top K holds none of them when at least K candidates lack a. □

Proposition 2. The odds are $q(s)/(1-q(s)) = \sum_{i\in G_a} e^{s_i} \big/ \sum_{j\in\mathcal{C}\setminus G_a} e^{s_j}$. Adding d to the scores of G_a multiplies the numerator by e^d and leaves the denominator unchanged. From the neutral odds $\rho_a^{\text{neu}}/(1-\rho_a^{\text{neu}})$, $d = \log\kappa_a$ gives $\kappa_a\,\rho_a^{\text{neu}}/(1-\rho_a^{\text{neu}}) = \rho_a^{\text{like}}/(1-\rho_a^{\text{like}})$ by the definition of κ_a. Shifts d_1 and d_2 multiply the odds by $e^{d_1+d_2}$, so INSERT(a) followed by DELETE(a) multiplies them by $e^{\alpha\log\kappa_a - \alpha\log\kappa_a} = 1$. □

Proposition 3. Let $t_i = \mathrm{sg}[s_\theta(p_u, x_i)] + \delta_e(i)$ and $z_i = s_\theta(e(p_u), x_i)$ for $i \in \mathcal{C}$, $P = \mathrm{softmax}(t)$ and $Q = \mathrm{softmax}(z)$. By Gibbs' inequality $\mathrm{KL}(P \,\|\, Q) \geq 0$ with equality if and only if $P = Q$. If $z = t + b\mathbf{1}$ the normaliser cancels and $P = Q$; conversely, $P_i = Q_i$ for all i gives $z_i - t_i = \log \sum_j e^{z_j} - \log \sum_j e^{t_j}$, which does not depend on i. With $\Delta_i = z_i - s_\theta(p_u, x_i)$ this reads $\Delta_i = \delta_e(i) + b$; (i) and (ii) follow because δ_e is zero off G_a and constant on it. Finally $\mathrm{KL}(P \,\|\, Q) = \sum_i P_i \log P_i - \sum_i P_i z_i + \log \sum_j e^{z_j}$, so $\partial \mathrm{KL}/\partial z_i = Q_i - P_i$: zero exactly at the target, and a gradient step lowers every item the student over-weights relative to the target and raises every item it under-weights. □

B OBEDIENCE IS NOT CALIBRATION, IN ONE FIGURE

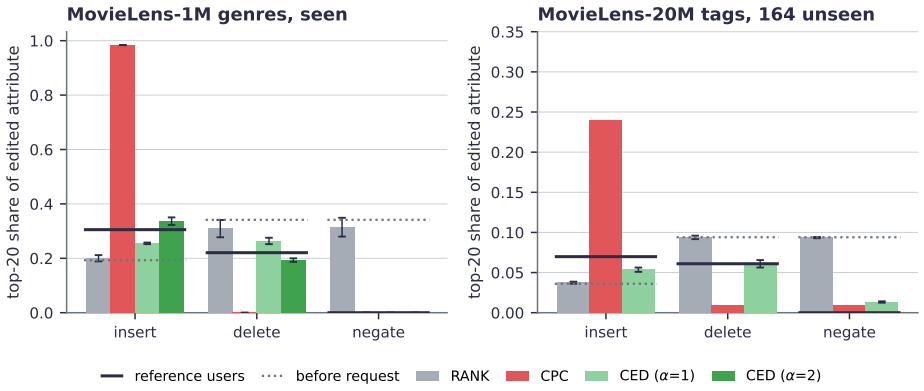


Figure 3: Top-20 share of the edited attribute after a request under RANK, CPC, CED ($\alpha = 1$) and, on MovieLens-1M, CED with the validation-selected scale ($\alpha = 2$); dotted lines mark the share before the request, solid lines the reference share (users whose profile states the attribute for INSERT, non-mentioners for DELETE, zero for NEGATE). RANK barely moves; CPC overshoots and treats a withdrawal like a negation; CED lands near the reference on DELETE and NEGATE, including on tags it was never trained to edit (right), and stays below it on INSERT at $\alpha = 1$ (left: 0.26 against 0.31; right: 0.054 against 0.070), a gap the selected scale closes on MovieLens-1M. Bars are means over seeds (three per method on MovieLens-1M; on MovieLens-20M three for RANK and CED, one for CPC), error bars one standard deviation; reference lines are the RANK model's reference shares, as in the main tables.

C CALIBRATION TARGETS ON MOVIELENS-1M

Table 5 lists, for each MovieLens-1M model, the top-20 share of the edited genre after an appended request next to that model's own reference share: the share among users who natively state the genre (INSERT) or say nothing about it (DELETE), and zero for NEGATE. κ_a is clipped to $[1.2, 20]$ before the logarithm, so that a noisy estimate never yields a negative or a runaway shift (one of the twelve MovieLens-1M training genres hits the lower bound), and an attribute is an INSERT target only if the training users who state it have at least 20 held-out positives in G_a. Targets differ between models because they are computed from each model's factual recommendations; a model that inflates its trust in stated likes (CPC bounded, native lift 2.60) raises its own target, which is why the tables in the main text use the RANK model's targets for every method. Against their own targets the two CED rows deviate by 0.088 ($\alpha = 1$) and 0.064 ($\alpha = 2$) on seen genres: the selected scale moves the model towards the RANK yardstick, which is the one a deployed system would measure against. A pooled reference and a pooled pre-request share need not coincide even for the same attribute set: an INSERT targets one unmentioned attribute per user, so rare attributes weigh more, while a DELETE applies to every stated one, so common attributes weigh more. Averaging the deviation per attribute before averaging over attributes, instead of over requests, leaves the comparison unchanged (seen / unseen genres, three seeds): RANK 0.19 / 0.15, CPC 0.87 / 0.88, CPC bounded 0.09 / 0.11, CFA 0.12 / 0.12, CED $\alpha = 1$ 0.10 / 0.09, CED $\alpha = 2$ 0.07 / 0.06. The NEGATE column also lists the

Table 4: Per-kind shares behind Table 2 (requests appended to MovieLens-1M profiles). NEGATE: request success; INSERT/DELETE: top-20 share of the genre after the request. Reference users: those whose own profile states the genre receive about 0.31 (seen) / 0.18 (unseen); those who do not mention it, 0.22 / 0.09. "per seed": the paired NDCG@10 difference against RANK for seeds 0, 1, 2 (the seed is the unit for any significance claim); "native lift": the top-20 share of a genre among users whose profile states it, divided by the genre's prevalence (§2), mean ± SD over seeds. The rewritten-profile CPC row is evaluated here on appended requests, which it was not trained on. CFA (counterfactual augmentation) trains on positives drawn from the reference users instead of a shift; it is seed-unstable on INSERT and keeps 0.77 negation success under the two unseen sentences per kind. The privileged-teacher rows are the seed-0 RANK checkpoint plus Eq. (5), told the kind and attribute of every request, at $\alpha = 1$ and $\alpha = 2$. Seeds: RANK 3, RANK (lr $2 \cdot 10^{-5}$) 3, CPC, profile rewritten 3, CPC, request appended 3, CPC bounded 3, CFA 3, CED ($\alpha = 1$) 3, CED ($\alpha = 2$, validation-selected) 3; single seed otherwise.

						seen genres			unseen genres			
Method	NDCG@10	Recall@20	Δ%	per seed	native lift	success	ins.	del.	success	ins.	del.	unrelated
RANK	0.375 ±0.019	0.382 ±0.010	0.0	–	2.32 ±0.04	0.08 ±0.10	0.20 ±0.01	0.31 ±0.03	0.14 ±0.13	0.10 ±0.01	0.17 ±0.03	0.983 ±0.002
RANK (lr $2 \cdot 10^{-5}$)	0.372 ±0.004	0.377 ±0.009	-0.7 ±4.2	-4.4 / -1.5 / +3.9	2.33 ±0.11	0.07 ±0.04	0.19 ±0.00	0.31 ±0.01	0.08 ±0.03	0.10 ±0.01	0.18 ±0.02	0.987 ±0.001
CPC, profile rewritten (on appended requests)	0.353 ±0.003	0.370 ±0.008	-5.9 ±5.0	-10.3 / -7.0 / -0.4	2.80 ±0.09	0.29 ±0.04	0.16 ±0.04	0.33 ±0.02	0.01 ±0.11	0.09 ±0.01	0.21 ±0.02	0.976 ±0.010
CPC, request appended	0.376 ±0.012	0.389 ±0.010	+0.3 ±5.4	-5.7 / +2.2 / +4.5	2.41 ±0.07	1.00 ±0.00	0.98 ±0.00	0.00 ±0.00	1.00 ±0.00	0.98 ±0.00	0.00 ±0.00	0.998 ±0.000
CPC bounded	0.368 ±0.009	0.380 ±0.003	-1.9 ±4.7	-7.0 / -1.0 / +2.2	2.60 ±0.06	1.00 ±0.00	0.28 ±0.00	0.29 ±0.01	1.00 ±0.00	0.16 ±0.01	0.16 ±0.01	0.997 ±0.000
CFA	0.378 ±0.006	0.391 ±0.008	+1.0 ±6.6	-4.6 / -0.7 / +8.3	2.32 ±0.06	1.00 ±0.00	0.33 ±0.08	0.29 ±0.01	1.00 ±0.00	0.18 ±0.07	0.16 ±0.00	0.998 ±0.001
CED ($\alpha = 1$)	0.371 ±0.012	0.382 ±0.004	-1.1 ±4.9	-1.6 / -5.7 / +4.1	2.26 ±0.09	1.00 ±0.00	0.26 ±0.00	0.26 ±0.01	1.00 ±0.00	0.13 ±0.00	0.14 ±0.01	0.994 ±0.000
CED ($\alpha = 2$, validation-selected)	0.371 ±0.011	0.382 ±0.005	-1.0 ±5.6	-2.1 / -6.0 / +5.1	2.30 ±0.11	1.00 ±0.00	0.34 ±0.01	0.19 ±0.01	1.00 ±0.00	0.19 ±0.01	0.09 ±0.00	0.990 ±0.000
Privileged teacher ($\alpha = 1$) (seed 0)	= RANK	= RANK	0.0	–	–	1.00	0.26	0.26	1.00	0.15	0.13	1.000
Privileged teacher ($\alpha = 2$) (seed 0)	= RANK	= RANK	0.0	–	–	1.00	0.34	0.19	1.00	0.21	0.08	1.000

Table 5: MovieLens-1M genres (appended requests, training phrasings, seen genres) with each model's *own* reference shares. "share" is the top-20 share of the edited genre after the request; "target" the share among users of the evaluation set who state (INSERT) or do not mention (DELETE) the genre under this model's factual recommendations; for NEGATE the target is 0 and the share among users whose profile states an aversion is given in parentheses. Δ% is paired per user over the seeds shared with RANK. Three seeds per row.

			INSERT		DELETE		NEGATE			
Method	NDCG@10	Δ%	share	ref.	share	ref.	share	ref.	success	unrelated
RANK	0.375 ±0.019	0.0	0.20 ±0.01	0.31	0.31 ±0.03	0.22	0.31 ±0.03	0 (0.20)	0.08 ±0.10	0.983 ±0.002
CPC bounded	0.368 ±0.009	-1.9 ±4.7	0.28 ±0.00	0.35	0.29 ±0.01	0.22	0.00 ±0.00	0 (0.19)	1.00 ±0.00	0.997 ±0.000
CFA	0.378 ±0.006	+1.0 ±6.6	0.33 ±0.08	0.31	0.29 ±0.01	0.22	0.00 ±0.00	0 (0.17)	1.00 ±0.00	0.998 ±0.001
CED ($\alpha = 1$)	0.371 ±0.012	-1.1 ±4.9	0.26 ±0.00	0.30	0.26 ±0.01	0.22	0.00 ±0.00	0 (0.18)	1.00 ±0.00	0.994 ±0.000
CED ($\alpha = 2$, validation-selected)	0.371 ±0.011	-1.0 ±5.6	0.34 ±0.01	0.30	0.19 ±0.01	0.22	0.00 ±0.00	0 (0.17)	1.00 ±0.00	0.990 ±0.000

share that users whose profile states an aversion receive (the "stated aversion" reference), which we do not use as a target.

D DATA DETAILS

MovieLens-20M users are sampled among those with 50–500 ratings; movies need at least 200 ratings; a tag is kept when at least 60 movies have genome relevance ≥ 0.6 for it and it covers at most 15% of the catalogue. An item has a tag when its relevance is at least 0.6 and clearly lacks it below 0.2. Table 6 uses a larger evaluation (300 test users, 600 candidates); all other tables use 200 test users with 500 candidates on MovieLens-1M and 1,000 on MovieLens-20M, which is why the same RANK configuration shows slightly different NDCG@10 across tables. On MovieLens-1M the six genres never used as request targets in training are Horror, Romance, Sci-Fi, Crime, War and Musical; at most three stated attributes per side are edited per user there, on MovieLens-20M two. Utility uses held-out positives against unrated candidates under the unedited profile.

The statement rule. A profile states the attributes that the user's own history supports. On MovieLens-1M, the *likes* are the three genres with the highest smoothed lift of the user's 4–5-star ratings over the training population's base rate (at least three positives in the genre; lift $= \frac{(n_g^+ + 5\, b_g)/(n^+ + 5)}{b_g}$ with b_g the base rate); the *aversion* is the genre with the lowest mean rating among genres rated at least five times with a mean of at most 3.0 (at most one); the *favourites* are the five highest-rated titles. On MovieLens-20M, likes are up to six tags with mean genome relevance ≥ 0.25 over the liked history and lift above 1 over the base rate, one per correlation cluster,

Table 6: Direct profile edits on MovieLens-1M under standard objectives (seen genres; 300 test users, 600 candidates, single seed). Success is the relative drop of the target genre's top-20 share; "shift" is the uniform score shift in units of the user's score spread. Interaction share is the variance of the score matrix not explained by additive user and item effects.

			DELETE		NEGATE		INSERT	
Objective	NDCG@10	inter.	success	shift	success	shift	share	shift
REG	0.171	0.054	0.24	0.10	0.32	0.51	0.14→0.19	0.10
REG+NEG	0.261	0.223	0.15	0.05	0.29	0.04	0.17→0.20	0.04
RANK	0.340	0.210	0.27	0.05	0.23	0.07	0.17→0.22	0.04
CPC, profile rewritten	0.301	0.267	0.60	0.05	0.98	0.04	0.13→0.43	0.06

and aversions up to two tags by the same rule over 1–2-star items from other clusters. Seeds shuffle the order in which the likes are listed; the sets themselves are deterministic. A rendered profile reads "*I love Horror, Action and Adventure movies. Some of my favorites are Goldfinger (1964); . . .*". The tag split, made over whole correlation clusters, is the stricter test of unseen attributes, since unseen genres can co-occur with seen ones (Horror with Thriller).

Training requests. One request per training group, sampled independently of the positive: with probability one half a stated training attribute with a kind drawn from $\{\text{DELETE}, \text{NEGATE}\}$, otherwise an INSERT of an unmentioned training attribute, in a training phrasing; DELETE and NEGATE are thus only ever applied to stated attributes, in training and at test time, and the DELETE target $-\log \kappa_a$ takes a stating user to the neutral rate.

Rewritten profiles, LLM-written profiles and prompts. A request *rewritten* into a templated profile changes the statement lists and re-renders the template: DELETE removes the attribute from the likes; NEGATE removes it from the likes and adds it to the aversion clause; INSERT removes it from the aversions if present and inserts it into the likes at a random position. LLM-written profiles (Table 8) are generated by Qwen3-14B from a sample of the user's history (the 15 highest-rated titles, up to 8 titles rated ≤ 2 and 7 recent ones, shuffled, each as "`rating/5 - title [genres]`") with the prompt "*Here are movies I have rated, from 1 (hated) to 5 (loved): . . . Write a first-person description of my movie taste in 60 to 90 words. Describe the genres, themes and styles I enjoy and the ones I tend to avoid. Mention at most three movie titles. Do not mention ratings or numbers. Output only the description.*"; a profile *mentions* an attribute when any word of a small per-genre lexicon occurs in it, and the same lexicon defines the reference users there. Rewriting such a profile (§4.5) prompts the same model with "*Here is my movie taste profile: "…" Rewrite it with exactly one change: {I no longer like g movies and now avoid them. | Remove every statement about g movies; do not say anything about g at all. | I also love g movies.} Keep everything else as close to the original wording as possible. Output only the rewritten profile.*" The zero-shot parser (Appendix G) is prompted with "*A user of a movie recommender wrote this update about their taste: "…" Classify the update. Answer with exactly two lines: kind: one of insert (the user now likes this kind of movie more), delete (the user withdraws a stated preference and wants it treated neutrally), negate (the user now wants to avoid this kind of movie); attribute: the kind of movie the update is about, copied from the update*", and the zero-shot ranker with "*Here is a user's movie taste profile: "…" Here is a movie: … Would this user enjoy this movie? Answer with yes or no.*", scoring by the log-odds of "yes".

E DIRECT PROFILE EDITS UNDER STANDARD OBJECTIVES

Table 6 supports §4.2: direct rewrites of the profile on MovieLens-1M (300 test users, 600 candidates, single seed) under rating regression, regression with a sampled negative, the sampled-softmax ranking loss, and CPC on direct edits. What Mamié & van den Bergh (2026) observed is a consequence of never showing the regression model a mismatched user–item pair; it is not what stands between the model and the edit. Two diagnostics are used here and in Appendix O: with $\Delta_i = s_\theta(p', x_i) - s_\theta(p, x_i)$ and σ_u the standard deviation of the user's scores over the candidates, the *efficacy* $\varsigma_k(\overline{\Delta}_{G_a} - \overline{\Delta}_{N_a})/\sigma_u$ ($\varsigma_k = +1$ for INSERT, -1 otherwise) is the selective part of a re-

Table 7: The 164 MovieLens-20M tags held out from training; reference shares from the RANK model (in parentheses: the share of users stating an aversion); $\Delta\%$: paired NDCG@10 change against RANK, mean $\pm$ SD over seeds. Seeds: RANK 3, CED ($\alpha = 1$) 3; single seed otherwise.

			INSERT		DELETE		NEGATE			
Method	NDCG@10	$\Delta\%$	share	ref.	share	ref.	share	ref.	success	unrelated
RANK	0.298 $\pm$0.005	0.0	0.037 $\pm$0.001	0.070	0.094 $\pm$0.002	0.061	0.094 $\pm$0.001	0 (0.042)	0.00 $\pm$0.02	0.975 $\pm$0.006
CPC, request appended	0.302	-0.3	0.239	0.070	0.009	0.061	0.009	0 (0.042)	0.91	0.978
CED ($\alpha = 1$)	0.305 $\pm$0.008	+2.3 $\pm$4.3	0.054 $\pm$0.003	0.070	0.061 $\pm$0.005	0.061	0.013 $\pm$0.001	0 (0.042)	0.85 $\pm$0.01	0.973 $\pm$0.007

Table 8: Other profiles and backbones on MovieLens-1M (appended requests, seen genres); $\Delta\%$ against RANK in the same setting, whose reference shares are used; "unseen phr.": negation success under unseen phrasings. CED on LLM-written profiles is trained on the extended phrasing set and evaluated under its six held-out sentences per kind. The zero-shot ranker row uses a 60-user, 300-candidate subset and its own reference shares (not comparable with the other rows; RANK on the same subset: NDCG@10 0.467, seed 0). Seeds: template RANK 3, template CED ($\alpha = 1$) 3; single seed otherwise.

Setting	Objective	NDCG@10	$\Delta\%$	success	INS. share (ref.)	DEL. share (ref.)	unseen phr.	unrelated
RoBERTa-base, template profiles	RANK	0.375 $\pm$0.019	0.0	0.08 $\pm$0.10	0.20 $\pm$0.01 (0.31)	0.31 $\pm$0.03 (0.22)	0.09 $\pm$0.08	0.983 $\pm$0.002
	CED ($\alpha = 1$)	0.371 $\pm$0.012	-1.1 $\pm$4.9	1.00 $\pm$0.00	0.26 $\pm$0.00 (0.31)	0.26 $\pm$0.01 (0.22)	0.92 $\pm$0.12	0.994 $\pm$0.000
RoBERTa-base, LLM-written profiles	RANK	0.332	0.0	0.03	0.15 (0.23)	0.41 (0.18)	0.04	0.986
	CED ($\alpha = 1$)	0.335	+0.8	1.00	0.18 (0.23)	0.28 (0.18)	0.87	0.995
	CED ($\alpha = 2$)	0.325	-2.2	1.00	0.27 (0.23)	0.19 (0.18)	0.93	0.991
Qwen2.5-0.5B (LoRA), template profiles	RANK	0.377	0.0	0.18	0.28 (0.33)	0.33 (0.22)	0.10	0.990
	CED ($\alpha = 1$)	0.384	+1.7	1.00	0.26 (0.33)	0.27 (0.22)	0.72	0.996
Qwen3-14B zero-shot ranker (SteerEval setting; 60 users, reference only)	–	0.250	–	0.94	0.36 (0.52)	0.23 (0.14)	0.92	0.884

sponse in units of that spread and the *uniform shift* $|\overline{\Delta}_{N_a}|/\sigma_u$ the non-selective part; the *interaction variance share* is the fraction of the score matrix's variance not explained by additive user and item effects. Under REG a negation shifts all scores by 0.51 spread units and interaction is 5% of the variance.

F RESULT TABLES: UNSEEN TAGS, OTHER PROFILES AND BACKBONES

Table 7 reports the MovieLens-20M results of §4.5, and Table 8 the results of §4.5: LLM-written profiles, a Qwen2.5-0.5B decoder backbone and the zero-shot Qwen3-14B ranker.

G RE-SCORING BASELINES: PARSERS AND CTRL-REC

Table 9 compares CED with parse-then-shift on the RANK checkpoint: the request is parsed into a kind and an attribute and the shift of Eq. (5) is added to the factual scores. Four parsers stand in for what a practitioner would build: a keyword parser written from the training phrasings alone, a keyword parser written with knowledge of every sentence (an upper bound no one can build in advance, which recovers the privileged teacher), a learned parser (a RoBERTa kind classifier trained on the same phrasings as CED, the attribute found by vocabulary match), and a zero-shot instruction-tuned LLM (Qwen3-14B) prompted to return the kind and the attribute. The privileged-teacher row uses the true kind and attribute. The zero-shot LLM parser misreads the kind or attribute of 338 of the 2,620 requests (13%), mostly withdrawals.

CTRL-Rec. CTRL-Rec (Carroll et al., 2025) steers a trained recommender with a natural-language request by adding a request-conditioned approval, learned from LLM-simulated judgments, to the signals the recommender already weighs. It cannot run unchanged here, since it is built for a feed recommender with its own signals and an LLM judge; we keep its mechanism and adapt its parts to our setting. The base scores are those of the seed-0 RANK checkpoint, the same scores the privileged teacher and the parsers use. The approval model embeds the request and the item text with a frozen text encoder (BGE-base (Xiao et al., 2024)) and trains an MLP on $[u, v, u \odot v]$ for the training attributes and training phrasings only, over 1,500 catalogue items outside the candidate pool. Its judgments are exact, as from a perfect judge: an insertion approves the items of its attribute, a negation disapproves them, and every other item, including every item under a withdrawal, is neutral. Under a request the score becomes $s_\theta(p, x_i) + w\,(2\hat{p}(e, i) - 1)$, with $w \in \{0.25, \ldots, 12\}$ chosen on 200 validation users by the rule used for CED: negation success of at least 0.90, then

Table 9: Parse-then-shift baselines against CED on MovieLens-1M, same users, candidates and requests. Cells give the top-20 share of the named genre after the request (NEGATE: share remaining, request success in brackets); "unseen phrasings" are request sentences from the disjoint test set; phrasing sets: base (3 training and 2 unseen sentences per kind) and extended (16 and 6; Appendix K). A keyword parser built from the training phrasings alone parses none of the unseen negations; the learned parser is a RoBERTa-base kind classifier fine-tuned on the training phrasings $\times$ training genres (attribute by vocabulary match), i.e. on the same text CED was trained on. All rows use the seed-0 checkpoints and parser rows the raw shift ($\alpha = 1$; the selected scale: Table 2).

Phrasing set	Method	INSERT seen / unseen		DELETE seen / unseen		NEGATE seen / unseen	
training phrasings							
base	RANK (no shift)	0.208	0.107	0.316	0.172	0.322 [0.07]	0.176 [0.11]
base	keyword parser, training phrasings only	0.263	0.146	0.263	0.134	0.000 [1.00]	0.000 [1.00]
base	keyword parser, all phrasings	0.263	0.146	0.263	0.134	0.000 [1.00]	0.000 [1.00]
base	learned parser (RoBERTa kind classifier, training phrasings)	0.264	0.146	0.263	0.134	0.000 [1.00]	0.000 [1.00]
base	privileged teacher (kind and attribute given)	0.263	0.146	0.263	0.134	0.000 [1.00]	0.000 [1.00]
base	CED	0.256	0.130	0.268	0.144	0.000 [1.00]	0.000 [1.00]
base	CED, selected scale	0.352	0.206	0.191	0.096	0.000 [1.00]	0.000 [1.00]
extended	RANK (no shift)	0.194	0.101	0.308	0.175	0.308 [0.12]	0.175 [0.11]
extended	keyword parser, training phrasings only	0.263	0.146	0.263	0.134	0.000 [1.00]	0.000 [1.00]
extended	keyword parser, all phrasings	0.263	0.146	0.263	0.134	0.000 [1.00]	0.000 [1.00]
extended	learned parser (RoBERTa kind classifier, training phrasings)	0.264	0.146	0.263	0.134	0.000 [1.00]	0.000 [1.00]
extended	LLM parser (Qwen3-14B, zero-shot)	0.263	0.146	0.159	0.067	0.000 [1.00]	0.002 [0.99]
extended	CED	0.246	0.144	0.261	0.135	0.000 [1.00]	0.000 [1.00]
extended	CED, selected scale	0.316	0.189	0.184	0.092	0.000 [1.00]	0.000 [1.00]
extended	CED, selected scale, 60 training phrasings per kind	0.324	0.189	0.186	0.101	0.000 [1.00]	0.000 [1.00]
unseen phrasings							
base	RANK (no shift)	0.209	0.108	0.329	0.194	0.320 [0.08]	0.166 [0.16]
base	keyword parser, training phrasings only	0.263	0.146	0.303	0.166	0.347 [0.00]	0.199 [0.00]
base	keyword parser, all phrasings	0.263	0.146	0.263	0.134	0.000 [1.00]	0.000 [1.00]
base	learned parser (RoBERTa kind classifier, training phrasings)	0.264	0.146	0.263	0.134	0.121 [0.65]	0.058 [0.71]
base	privileged teacher (kind and attribute given)	0.263	0.146	0.263	0.134	0.000 [1.00]	0.000 [1.00]
base	CED	0.207	0.097	0.256	0.132	0.003 [0.99]	0.005 [0.98]
base	CED, selected scale	0.261	0.140	0.166	0.088	0.063 [0.81]	0.039 [0.81]
extended	RANK (no shift)	0.181	0.096	0.307	0.178	0.307 [0.12]	0.172 [0.12]
extended	keyword parser, training phrasings only	0.182	0.098	0.285	0.149	0.347 [0.00]	0.199 [0.00]
extended	keyword parser, all phrasings	0.221	0.122	0.263	0.134	0.000 [1.00]	0.000 [1.00]
extended	learned parser (RoBERTa kind classifier, training phrasings)	0.264	0.146	0.215	0.115	0.050 [0.86]	0.019 [0.90]
extended	LLM parser (Qwen3-14B, zero-shot)	0.263	0.146	0.168	0.096	0.000 [1.00]	0.000 [1.00]
extended	CED	0.217	0.126	0.201	0.111	0.041 [0.88]	0.021 [0.89]
extended	CED, selected scale	0.299	0.184	0.136	0.063	0.016 [0.95]	0.006 [0.97]
extended	CED, selected scale, 60 training phrasings per kind	0.281	0.178	0.135	0.075	0.000 [1.00]	0.001 [1.00]

the smallest response deviation. On validation users negation reaches 0.90 only from $w = 6$, where an added genre already takes 0.67 of the top 20 (reference 0.34); the smallest deviation (0.16) is at $w = 1.5$, where negation success is 0.40 (0.41 on test users); and a withdrawn genre stays between 0.32 and 0.37 (reference 0.21) at every weight. One weight cannot give a negation its constraint and an insertion its amount, and an approval signal has no value for "neutral".

A 14B zero-shot ranker. Qwen3-14B scores each candidate by the log-odds of "yes" to "would this user like this movie?" given profile and request, the SteerEval setting (Zhou et al., 2026), on a 60-user, 300-candidate subset of the MovieLens-1M evaluation set against its own reference shares (Table 8; the row is not comparable with the others). It listens (negation success 0.94; 0.92 under unseen phrasings) but reads the profile almost as a filter (native lift 4.8 against 2.3 for RANK on the same subset), moves unrelated items more (0.88), and its NDCG@10 is 0.25 against 0.47 for RANK on the same users and candidates.

H WHERE THE EXTRA ITEMS ENTER THE LOSS

CED appends two unrated items of the edited attribute to every training group so that the distillation target always contains items the edit is about. If these two items instead replace two of the sampled negatives, they also enter the factual ranking loss $\mathcal{L}_{\mathrm{rank}}$. For DELETE and NEGATE the edited attribute is one the user *likes*, so unseen items of liked attributes would be treated as negatives. On MovieLens-1M (seed 0, measured on the 200 evaluation users) this costs 7.2% NDCG@10 relative to RANK, and lowers the model's factual lift for reference users of an attribute from 2.36 to 1.94; what matters is which items enter $\mathcal{L}_{\mathrm{rank}}$, not a hyperparameter. Keeping the two items out of $\mathcal{L}_{\mathrm{rank}}$

and using them only in the distillation term restores the lift (2.31) and the accuracy (-1.6% on seed 0) with the same request behaviour.

I HYPERPARAMETER SELECTION ON VALIDATION USERS

Table 10: Hyperparameter search on MovieLens-1M (200 validation users, seed 0); ΔNDCG@10 is paired against RANK on the same users (lr $2\cdot10^{-5}$, its best learning rate). "share (ref.)" as in Table 8; "deviation" is the sum of the four absolute deviations from the reference shares computed from each configuration's own factual recommendations (Appendix C); a configuration is feasible when ΔNDCG@10 $\geq -2\%$ and negation success ≥ 0.90 (the rule then picks the feasible configuration with the smallest deviation).

Method	α	λ	lr	NDCG@10	Δ%	NEG.	INS. seen	DEL. seen	INS. unseen	DEL. unseen	deviation	feasible
CED	1	0.3	3e-05	0.351	-0.1	1.00	0.25 (0.33)	0.29 (0.21)	0.13 (0.22)	0.13 (0.09)	0.317	✓
CED	1	1	3e-05	0.352	+0.2	1.00	0.26 (0.33)	0.28 (0.21)	0.14 (0.22)	0.14 (0.09)	0.290	✓
CED	1	3	3e-05	0.347	-1.1	1.00	0.28 (0.33)	0.28 (0.21)	0.14 (0.23)	0.14 (0.09)	0.260	✓
CED	1.5	0.3	3e-05	0.347	-1.2	1.00	0.29 (0.34)	0.25 (0.21)	0.15 (0.23)	0.11 (0.09)	0.181	✓
CED	1.5	1	3e-05	0.361	+2.7	1.00	0.31 (0.34)	0.25 (0.21)	0.16 (0.22)	0.11 (0.09)	0.146	✓
CED	1.5	3	3e-05	0.348	-0.8	1.00	0.31 (0.34)	0.24 (0.21)	0.16 (0.22)	0.11 (0.09)	0.139	✓
CED	2	0.3	2e-05	0.355	+1.1	1.00	0.35 (0.34)	0.21 (0.21)	0.20 (0.23)	0.09 (0.08)	0.056	✓
CED	2	0.3	3e-05	0.357	+1.7	1.00	0.35 (0.33)	0.21 (0.21)	0.20 (0.22)	0.09 (0.08)	0.040	✓
CED	2	0.3	5e-05	0.347	-1.1	1.00	0.32 (0.33)	0.21 (0.21)	0.17 (0.21)	0.09 (0.09)	0.054	✓
CED	2	1	3e-05	0.363	+3.4	1.00	0.36 (0.34)	0.21 (0.21)	0.18 (0.22)	0.10 (0.08)	0.068	✓
CED	2	3	3e-05	0.355	+1.2	1.00	0.35 (0.34)	0.20 (0.21)	0.18 (0.23)	0.10 (0.08)	0.084	✓
CPC	1	0.3	3e-05	0.334	-4.7	1.00	0.99 (0.35)	0.00 (0.21)	0.98 (0.21)	0.00 (0.08)	1.709	–
CPC	1	3	3e-05	0.336	-4.4	1.00	0.99 (0.34)	0.00 (0.21)	0.98 (0.21)	0.00 (0.08)	1.706	–

The selection rule: among configurations whose paired NDCG@10 on validation users is within -2% of the best RANK model on validation users and whose negation success is at least 0.90, take the one with the smallest deviation, breaking ties towards the smaller λ and the α closest to 1; the selected configuration is then trained with three seeds and evaluated on test users, and the untuned default ($\alpha = 1$, $\lambda = 1$) is reported alongside it. The odds ratio κ_a is a ratio of rates over positives while the metric is a share of the top 20, and Table 10 shows what the mismatch amounts to: with the raw shift ($\alpha = 1$) inserted seen genres land 16–26% below and withdrawn ones 32–41% above their references whatever λ is (unseen genres: about 40% below and 60% above); $\alpha = 1.5$ halves the gaps; $\alpha = 2$ closes them on seen genres (within about 5%) and at least halves them on unseen genres, at no visible cost in accuracy. All nine $\alpha \times \lambda$ configurations are feasible (the two further rows vary the learning rate of the selected one), and λ matters little at any scale; the selection rule picks $\alpha = 2$, $\lambda = 0.3$ (deviation 0.040 against 0.290 for the default).

J ABLATIONS

The natural repair of CPC is to ask for a fixed response: replacing the softplus in Eq. (3) by $(\varsigma_k(\Delta_{i^+} - \overline{\Delta}_U) - \log \kappa_a)^2$ for INSERT and DELETE (with the sign ς_k of Eq. (3); for NEGATE the margin is c) stops the over-reaction (INSERT lands at 0.28 against the 0.31 reference, closer than CED's 0.26; DELETE at 0.29 against 0.22, farther than CED's 0.26) at an accuracy change of $-1.9 \pm 4.7\%$ NDCG@10 over three seeds (-7.0, -1.0, $+2.2\%$), i.e. within seed noise, and raises the model's trust in stated likes (native lift 2.60 ± 0.06 against 2.32 ± 0.04 for RANK). A bounded margin thus buys the same calibration (Table 2); the distillation target additionally leaves the model's trust in stated likes at the baseline, which is why it is the default. CFA (counterfactual augmentation (Kaushik et al., 2020)) replaces the shift by positives sampled from the reference users: it reaches the INSERT reference on average (0.33 ± 0.08 against 0.31) but is seed-unstable (0.28–0.42 over seeds) and keeps 0.77 ± 0.10 negation success under unseen phrasings, against 0.92 ± 0.12 for CED (Table 4). CED is insensitive to λ (test users, seed 0): at $\lambda \in \{0.3, 1, 3\}$ the inserted genre lands at 0.24 / 0.26 / 0.26 (target 0.31), the withdrawn genre at 0.28 / 0.27 / 0.27 (target 0.22), negation success stays at 1.00, and NDCG@10 moves by $+0.3$ / -1.6 / -1.3%, all within seed noise.

Phrasing coverage. With the selected configuration ($\alpha = 2$, $\lambda = 0.3$) trained on the extended phrasing set (16 sentences per kind, seed 0) or on 60 sentences per kind (the extended set plus

43–45 Qwen3-14B paraphrases per kind, filtered so that no content-word bigram is shared with an unseen sentence; three seeds), the response under unseen phrasings reaches the reference on INSERT (0.30 with the extended set; 0.29 ± 0.01 with 60 sentences, against 0.31; 0.25 with the base set) with negation success 0.95 / 1.00 ± 0.00, while DELETE withdraws further than the reference (0.14 ± 0.00 against 0.22); in the training phrasings the 60-sentence model lands on the reference for INSERT (0.32 ± 0.01 against 0.31) and four points under it for DELETE (0.18 ± 0.01 against 0.22). Its paired NDCG@10 change is $-2.0 \pm 4.9\%$ over three seeds (-5.1, -4.5, $+3.6\%$; the three-sentence model: $-1.0 \pm 5.6\%$), and its native lift is 2.38 ± 0.02. The main tables keep the base set; Table 9 lists the per-kind shares.

The scale of the shift, at the level of the privileged teacher. Adding Eq. (5) to the seed-0 RANK scores with the kind and attribute given (the privileged teacher), at $\alpha = 1$, 1.5, 2, 2.5 and 3, gives response deviations of 0.090, 0.013, 0.065, 0.140 and 0.211 on seen genres (0.069, 0.020, 0.044, 0.103, 0.162 on unseen ones) against the RANK yardstick: the raw odds ratio under-shoots, the deviation is smallest near $\alpha = 1.5$ on the test users, and $\alpha = 2$ over-shoots insertion by three points while landing withdrawal within three. CED reproduces the privileged teacher at every scale it was trained at: 0.26 / 0.26 at $\alpha = 1$ and 0.34 / 0.19 at $\alpha = 2$ against the teacher's 0.263 / 0.263 and 0.344 / 0.190 (three seeds), and at $\alpha = 1.5$ (seed 0, $\lambda = 0.3$) 0.300 / 0.222 against the teacher's 0.302 / 0.226 (unseen genres 0.17 / 0.11 against 0.18 / 0.11; negation 1.00; NDCG@10 -2.9% on that seed), so the scale is a property of the target and not of the learner, as Proposition 3 says. Does the population statistic matter, or only the size? An attribute-independent constant equal to the mean shift at $\alpha = 2$ (0.895 for every genre) matches the pooled deviation (0.063 against 0.065 on seen genres) but is farther from the reference on 9 of the 12 genres for both INSERT and DELETE (mean per-genre gap 0.050 / 0.035 against 0.036 / 0.025 for $\log \kappa_a$): the odds ratio sizes the shift per attribute, and the pooled number hides it. Halving the negation constant to $c = 4$ changes nothing either (negation success 1.00, NDCG@10 -2.6%). CPC is also insensitive to its weight, but it over-reacts at every weight: on validation users $\lambda \in \{0.3, 3\}$ gives the same full success, the same over-reaction (INSERT 0.99 against a reference share of 0.35) and -4.7 / -4.4% NDCG@10, so no CPC configuration meets the utility bound of the selection rule (Appendix I). How many attributes must be edited in training for the behaviour to transfer? Restricting the MovieLens-20M training edits to 12 or 60 of the 474 seen tags (unseen set unchanged) gives unseen negation success of 0.76 and 0.86 against 0.84 with all 474 (seed 0), withdrawn unseen tags at 0.076 / 0.063 / 0.056 against a neutral level of 0.06, and NDCG@10 of 0.289 / 0.309 / 0.300: a few dozen edited attributes suffice, a dozen is too few.

K REQUEST PHRASINGS

Every request is appended to the unchanged profile as "*Update from me: …*". The *base* phrasing set, used throughout unless stated otherwise, has 3 training and 2 unseen sentences per request kind, shown in bold; the *extended* set (§4.5) has 16 and 6, and every base sentence is also in it, on the same side. Several NEGATE and DELETE sentences are deliberately close in phrasing so that the kind of a request cannot be read off a keyword.

INSERT. *Training:* **I also love ⟨attr⟩.**; **Lately I have become a big fan of ⟨attr⟩.**; **I really enjoy ⟨attr⟩ too.**; Please recommend more ⟨attr⟩.; I want to see more ⟨attr⟩.; I have started to like ⟨attr⟩ a lot.; Add ⟨attr⟩ to the things I enjoy.; ⟨Attr⟩ are something I am into now.; I am in the mood for ⟨attr⟩ these days.; I have discovered that I like ⟨attr⟩.; Show me ⟨attr⟩ as well.; I am keen on ⟨attr⟩ now.; Include ⟨attr⟩ in my recommendations.; My new interest is ⟨attr⟩.; I have developed a taste for ⟨attr⟩.; I am enjoying ⟨attr⟩ more and more. *Held out:* **I would like to see more ⟨attr⟩; I have grown fond of them.**; **⟨Attr⟩ have become a favorite of mine.**; Recently ⟨attr⟩ have really won me over.; Feel free to suggest ⟨attr⟩; I am a fan now.; I cannot get enough of ⟨attr⟩ lately.; Put some ⟨attr⟩ in my feed, I like them.

DELETE. *Training:* **Please disregard what my profile says about ⟨attr⟩.**; **I am not particularly interested in ⟨attr⟩ anymore; treat them like any other kind.**; **Forget my preference for ⟨attr⟩.**; I no longer have a special preference for ⟨attr⟩.; Ignore my interest in ⟨attr⟩.; Do not give ⟨attr⟩ any special weight for me.; My liking for ⟨attr⟩ has faded; I feel neutral about them.; I feel indifferent about ⟨attr⟩ now.; Stop assuming I prefer ⟨attr⟩.; Drop ⟨attr⟩ from my stated preferences.; Treat ⟨attr⟩

as neither liked nor disliked.; I am neutral on ⟨attr⟩ these days.; Remove my preference for ⟨attr⟩; I do not mind them either way.; ⟨Attr⟩ are no longer a particular favorite of mine, though I do not mind them.; Please do not prioritize ⟨attr⟩ for me anymore.; My profile should not highlight ⟨attr⟩. *Held out:* **Do not assume I have any special taste for ⟨attr⟩.**; **My profile overstates my interest in ⟨attr⟩; ignore that part.**; I can take or leave ⟨attr⟩ now.; ⟨Attr⟩ are not a big thing for me anymore, but I do not avoid them.; No need to favor ⟨attr⟩ on my account.; Scratch the bit about ⟨attr⟩; I have no strong feelings.

NEGATE. *Training:* **I no longer enjoy ⟨attr⟩ and now avoid them.**; **These days I stay away from ⟨attr⟩.**; **I have lost interest in ⟨attr⟩ and would rather not see them.**; Please stop recommending ⟨attr⟩.; I do not want ⟨attr⟩ anymore.; No more ⟨attr⟩ for me.; I have had enough of ⟨attr⟩; leave them out.; I dislike ⟨attr⟩ now.; Remove ⟨attr⟩ from what you suggest to me.; I am tired of ⟨attr⟩ and want to avoid them.; ⟨Attr⟩ no longer appeal to me at all.; Skip ⟨attr⟩ when choosing things for me.; I would prefer never to be shown ⟨attr⟩.; My taste has changed: I cannot stand ⟨attr⟩ now.; Exclude ⟨attr⟩ from my recommendations.; I am done with ⟨attr⟩. *Held out:* **Please keep ⟨attr⟩ out of my recommendations; I do not like them anymore.**; **⟨Attr⟩ are not for me anymore.**; Going forward, I would rather you did not suggest ⟨attr⟩.; I have grown to hate ⟨attr⟩.; Count me out when it comes to ⟨attr⟩.; Steer clear of ⟨attr⟩ in my feed.

Synonym attribute phrases (probe in Appendix O). One per genre, each a paraphrase of the genre phrase used in training requests, many without the genre word: Horror → "scary movies"; Sci-Fi → "science fiction films"; Romance → "love stories"; Crime → "gangster and heist films"; War → "war films"; Musical → "song-and-dance films"; Comedy → "funny films"; Action → "action-packed films"; Thriller → "suspense films"; Drama → "dramatic films"; Animation → "animated films"; Children's → "kids' films"; Adventure → "adventure films"; Mystery → "whodunits"; Fantasy → "fantasy films"; Western → "cowboy films"; Documentary → "documentaries"; Film-Noir → "noir films".

L TAG GENOME ATTRIBUTE SPLIT

From the 1,128 genome tags we keep the 638 that have at least 60 movies with relevance ≥ 0.6 among the 6718 movies with at least 200 ratings and that cover at most 15% of them. Tags whose relevance profiles over these movies correlate above 0.6 are joined into clusters (single linkage), giving 133 clusters; the largest join near-synonyms such as {adaptation, literature, books, book was better, book}. Clusters of at most six tags are assigned to the unseen side at random, which holds out 164 of the 638 tags (26%). An unseen tag is never a request target in training and has no near-synonym among the seen tags; it still occurs in profiles and item texts. The unseen tags are:

> 18th century; 1930s; 1970s; 1980s; 70mm; 80s; adoption; adultery; africa; alone in the world; awesome soundtrack; based on a tv show; betrayal; better than expected; biographical; biography; biopic; brainwashing; british; british comedy; brothers; buddy movie; california; camp; campy; cancer; cars; censorship; cold war; computer game; controversial; costume drama; courtroom; creativity; dark fantasy; deadpan; death; desert; destiny; directorial debut; divorce; doctors; documentary; dog; dreams; dysfunctional family; eccentricity; england; ensemble cast; epic; family bonds; family drama; farce; fascism; father daughter relationship; father son relationship; foul language; france; french; gay; gay character; glbt; god; goth; gothic; graphic novel; hip hop; hitchcock; holiday; homeless; homosexuality; hospital; hotel; humanity; immortality; indians; infidelity; innocence lost; internet; irish; irish accent; islam; island; japan; japanese; jungle; kidnapping; latin america; lawyer; lawyers; life & death; life philosophy; light; london; long; los angeles; lyrical; mad scientist; male nudity; marriage; medieval; memory; multiple storylines; new jersey; new york; new york city; nostalgic; nudity (topless - brief); nudity (topless - notable); nudity (topless); original plot; parallel universe; parenthood; paris; period piece; president; prison; prison escape; product placement; psychiatry; pulp; queer; rags to riches; rape; rebellion; redemption; religion; remake; road movie; road trip; robot; robots; san francisco; screwball; screwball comedy; sexual abuse; shallow; single father; sisters; small town; stereotypes; stop motion; suicide; survival; swashbuckler; sword fight; sword fighting; time travel; tokyo; tom clancy; transformation; trilogy; undercover cop; vampire; vampires; video game; vigilante; vigilantism; weapons; western; whimsical; women; workplace; writers.

Table 11: An independent unseen phrasing set: 12 hand-written sentences per request kind (Appendix M), written after training and sharing no content-word bigram with any training or test sentence, applied to MovieLens-1M test users (200 users, 500 candidates). Cells: top-20 share of the named genre after the request (reference users' share in parentheses) and negation success. RANK: seed 0; CED: three seeds.

		seen genres			unseen genres		
Model	trained on	INSERT	DELETE	NEGATE	INSERT	DELETE	NEGATE
RANK (ignores the request)	–	0.20 (0.31)	0.33 (0.22)	0.06	0.10 (0.17)	0.19 (0.09)	0.13
CED ($\alpha = 2$)	60 sentences	0.27 $\pm$0.02 (0.31)	0.15 $\pm$0.00 (0.22)	1.00 $\pm$0.00	0.14 $\pm$0.01 (0.17)	0.08 $\pm$0.01 (0.09)	1.00 $\pm$0.00

M AN INDEPENDENT UNSEEN PHRASING SET

The independent set was written by hand after all models were trained, in a different register from the template-like sentences of the other sets (imperatives, questions, colloquial phrasings), and filtered so that no stemmed content-word bigram is shared with any sentence of the other sets. Table 11 evaluates the 60-sentence CED models under it: all three request kinds move in the requested direction and negation stays complete. The sentences are: INSERT: Could you work more ⟨attr⟩ into my recommendations?; Lately ⟨attr⟩ are exactly my thing; Give ⟨attr⟩ a bigger place in my feed; Consider me a convert to ⟨attr⟩; I keep coming back to ⟨attr⟩ these days, so lean into that; Make room for ⟨attr⟩; they have grown on me; ⟨Attr⟩ deserve more weight in my profile than they get; Bring ⟨attr⟩ to the front of my recommendations; I am into them at the moment; Count ⟨attr⟩ among the kinds I actively look for now; Bump up ⟨attr⟩ for me; My taste has shifted towards ⟨attr⟩; Treat ⟨attr⟩ as one of my favourite kinds from now on. DELETE: Don't read anything special into my liking ⟨attr⟩; it's not a big deal; Downgrade ⟨attr⟩ to an ordinary kind for me, neither favoured nor avoided; That ⟨attr⟩ preference in my profile is out of date; drop it; Handle ⟨attr⟩ the same way as everything else; Do not let ⟨attr⟩ get a thumb on the scale for me; I never meant ⟨attr⟩ to be a highlight of my profile; Set my ⟨attr⟩ preference back to neutral; Whatever my profile claims about ⟨attr⟩, treat it as a plain, ordinary kind; Take ⟨attr⟩ off my list of favourites; ordinary is fine; Remove the emphasis on ⟨attr⟩; I am indifferent to them now; ⟨Attr⟩ should no longer count as something I favour; Please stop reading ⟨attr⟩ as a thing I especially want. NEGATE: Please make sure ⟨attr⟩ never show up for me; ⟨Attr⟩ put me off; I would rather they stayed out of sight; I'd rather not be shown ⟨attr⟩ at all; Cut ⟨attr⟩ from my recommendations entirely; I have gone off ⟨attr⟩ completely; No ⟨attr⟩, please, they are not for me; Filter out ⟨attr⟩ from now on; ⟨Attr⟩ are a hard pass for me these days; Keep ⟨attr⟩ well away from my feed; I now actively dislike ⟨attr⟩; Stop suggesting ⟨attr⟩; I won't watch them; Leave ⟨attr⟩ aside for good; my interest in them is gone.

N COMPUTE, SEEDS AND RUN TIMES

Experiments ran on one NVIDIA RTX 4090 (24 GB) and, for the hyperparameter search and the larger backbones, one RTX A6000 (48 GB). With RoBERTa-base, training and evaluation take about 20 minutes for RANK and 35 minutes for CED on MovieLens-1M, and about 50 and 85 minutes on MovieLens-20M tags (on the 4090, with gradient accumulation over two micro-batches for CED on MovieLens-20M). Evaluating one model scores every candidate under every edited profile, roughly $200 \times 500 \times 12$ profile–item pairs on MovieLens-1M. The user split and the 200 evaluation users are fixed; seeds control the sampled negatives, the order in which a profile lists its likes and the model initialisation, so reference shares are recomputed per seed.

O BEHAVIOURAL PROBES

Five behavioural probes on MovieLens-1M (100 test users, single seed) ask how a trained model reads an appended request. *Attribute sensitivity* swaps the attribute named in the request for another and measures the resulting score change on the items of the original attribute, in units of the user's score spread. The baseline moves by 0.16 for a negation and 0.11 for an insertion: it barely registers *which* attribute the sentence is about. CED moves by 2.25 for a negation; a CPC model by 4.3 for a negation and 5.5 for an insertion. *Marker ablation* removes the "*Update from me:*" prefix. CED complies identically with and without it (1.00 vs. 1.00); the model does not key on the

marker but on the sentence, whose form does not occur in system-written profiles. *Position ablation* prepends the request instead of appending it. CED keeps 0.92–0.98 success; the baseline goes to -0.17, i.e. a negation placed first *raises* the share of the negated genre, the bag-of-words reading in which mentioning a genre is evidence of liking it. CPC's inflated INSERT response, 0.98 when appended, drops to 0.33 when prepended: part of what an unbounded contrastive objective learns is where in the input the sentence sits. *Precedence* rewrites the favourites so that all five belong to the genre being negated, the strongest cue a profile can carry against the request; CED still complies at 1.00. *Synonyms* replace the genre phrase of the request by a paraphrase ("*scary movies*" for Horror, "*whodunits*" for Mystery; Appendix K); CED keeps 0.90 negation success and about 70% of its insertion response, the baseline none. The unseen-attribute results are therefore not string matching: the model reads what the request is about.