EM-Style Schema Refinement: Operations and Probability Calculations

EM is a classic algorithm that is TYPICALLY used to jointly (1) update model parameters θ or (2) update latent variable assignments z to data x (look up EM for Gaussian mixtures). In this proposal, we're proposing to running an EM-style loop where the latent variables are schema assignments and the "parameters" we update are not weights—but the schema itself.

On a high-level, we will do the doing the following steps:

1. **E-step**: assign labels to data based on the current (or initial) schema. Calculate metrics around how well these labels fit the data.

2. **M-step**: update the schema —through split, merge, add, remove operations — then recompute metrics. Accept the new schema as the current schema if metrics improve.

We iterate between the two steps until scores and assignments stabilize. We call this structural variational EM.

We can also imagine the following extensions to this EM approach:

- **Supervised Variant**: Instead of the E-step measuring *how well the labels fit the datapoints* (unsupervised variant), it can instead measure how well how schema labels fits *other* labels (see Concept Bottleneck Models [1]). This means we will tailor a schema towards a particular downstream variable.

- **Causal Variant**: supporting JB's directions, we can also extend the E-step to measure *how well the schema captures causality* by measuring for e.g. interventional log-likelihood, invariance penalties Inv, and yield identifiable, larger (C)ATEs. This means we will tailor a schema towards particular causal variable. (I know this is vague, I haven't fully thought it through, and am also not as much of an expert on causality).

[1] `https://arxiv.org/pdf/2007.04612`

I will know describe the algorithm in a little more detail.

1 Setup

Given observed texts be $X = \{x_i\}_{i=1}^{N}$, an initial schema Z_0 and a reliable-enough mechanism $Z^0 = \{z_j\}_{j=1}^{K}$ to apply this to all texts (i.e. our initial methodology)[2].

2 E-Step (Labeling / Scoring)

2.1 Label Assignment and Probability Calculation

We will calculate several different probabilities to help us measure how well a label fits a text.

$$\log p(z_i \mid x) = \sum_{t \in \text{text tokens of } z_j} \log p(z_i \mid z_j, x)\,, \tag{1}$$

$$\log p(x_i) = \sum_{t \in \text{text tokens of } x_i} \log p(x_{i,t})\,, \tag{2}$$

$$p(x_i \mid z_j) = \frac{p(z_j \mid x_i)\, p(z_j)}{\sum_{k=1}^{K} p(z_k \mid x_i)\, p(z_k)} \tag{3}$$

Where $p(\cdot)$ are probabilities generated by an LLM. There are a million ways to calculate the calibrated probability $p(z_i|x)$ (note that it has to be calibrated, because some labels z are just more likely to be generated, anyway). I've experimented with this one[3] We can use $p(z_j|x_i)$ to assign labels to texts. Then, we can use these probabilities to calculate statistics:

2.2 Document-Level Scores

Given probabilities for x, z, $x|z$, $z|x$, we can then compute scores across a document, given a label scheme.

[2]Texts may be single- or multi-label

[3]`https://arxiv.org/pdf/2102.09690`, my code here: `https://github.com/alex2awesome/schema-generation/blob/main/src/utils_probability_calibrator.py`

Average log-likelihood per category.

$$L(z_j) \;=\; \frac{1}{|X_{z_j}|} \sum_{x_i \in X_{z_j}} \log p(x_i \mid z_j). \tag{4}$$

Average posterior confidence.

$$C(z_j) \;=\; \frac{1}{N} \sum_{i=1}^{N} p(z_j \mid x_i) \tag{5}$$

Embedding centroid and intra-category variance. Let e_{x_i} be a text embedding.

$$e_{z_j} = \frac{1}{|X_{z_j}|} \sum_{x_i \in X_{z_j}} e_{x_i}, \qquad V(z_j) = \frac{1}{|X_{z_j}|} \sum_{x_i \in X_{z_j}} \|e_{x_i} - e_{z_j}\|^2. \tag{6}$$

Pairwise category similarity.

$$\mathrm{sim}(z_a, z_b) \;=\; \frac{e_{z_a} \cdot e_{z_b}}{\|e_{z_a}\| \, \|e_{z_b}\|}. \tag{7}$$

Baseline

$$L_{\text{baseline}} \;=\; \frac{1}{N} \sum_{i=1}^{N} \log p(x_i). \tag{8}$$

Poorly explained texts.

$$p_{\max}(x_i \mid Z) \;=\; \max_{1 \le j \le K} p(x_i \mid z_j). \tag{9}$$

2.3 Corpus-Level Scores

We can also calculate the following corpus level scores. These scores give us a sense of the schema at timestep t, V_t, overall. Let $\mathcal{D}_{\text{train}}$ be the working set and $\mathcal{D}_{\text{test}}$ held-out.

Total (conditional) log-likelihood.

$$\log L_{\text{cond}}(\mathcal{D}) \;=\; \sum_{x_i \in \mathcal{D}} \log p\big(x_i \mid \hat{z}_i\big). \tag{10}$$

AIC & BIC. Let k be schema complexity (e.g., $k = K$ or a richer count) and $n = |\mathcal{D}_{\text{test}}|$:

$$\text{AIC} = 2k - 2\log L_{\text{cond}}(\mathcal{D}_{\text{test}}), \tag{11}$$

$$\text{BIC} = k\log n - 2\log L_{\text{cond}}(\mathcal{D}_{\text{test}}). \tag{12}$$

Perplexity. Let $T = \sum_{x_i \in \mathcal{D}} |x_i|$ be the total token count:

$$\text{PPL}(\mathcal{D}) \;=\; \exp\left(-\frac{1}{T}\sum_{x_i \in \mathcal{D}} \log p\big(x_i \mid \hat{z}_i\big)\right). \tag{13}$$

ELBO (variational framing). With $q_{ij} = p(z_j \mid x_i)$ as a variational posterior and prior $p(z_j)$:

$$\mathcal{L} \;=\; \sum_{i=1}^{N}\sum_{j=1}^{K} q_{ij}\,\log p(x_i, z_j) \;+\; H(Q), \quad \text{where } p(x_i, z_j) = p(x_i \mid z_j)p(z_j). \tag{14}$$

3 M-Step (Schema Update Proposals)

Given scores calculated during the E-step, we then decide how to update our schema, yeilding our current schema Z_t.

3.1 Structural Moves

Split z_j. When $L(z_j)$ is in the bottom quartile and/or $C(z_j)$ in the bottom quartile and/or $V(z_j)$ in the top quartile: cluster X_{z_j} (e.g., k=2) into $X_{z_{j_1}}, X_{z_{j_2}}$; form z_{j_1}, z_{j_2}.

Merge z_a, z_b. When $\text{sim}(z_a, z_b) > \tau_{\text{sim}}$ and $|L(z_a) - L(z_b)| < \epsilon_L$, $|C(z_a) - C(z_b)| < \epsilon_C$. Pool members into $z_{a \cup b}$, recompute quantities, and score.

Remove z_j. When $|X_{z_j}|/|X| < \tau_{\text{min-size}}$ or $L(z_j) \leq L_{\text{baseline}}(1 + \delta)$. Drop z_j, reassign its texts by $p(z \mid x)$ over remaining labels, recompute, and score.

Add z_{new}. Cluster worst-explained texts (e.g., bottom 5–10% by $p_{\max}(x_i \mid Z)$); introduce z_{new}, recompute, and score.

3.2 Scoring & Acceptance

Recalculate scores across the new schema V_t. Accept the proposed new schema Z_t if BIC (or AIC) decreases on $\mathcal{D}_{\text{test}}$ *or* if $\mathcal{L}$ increases. After accepting a schema change, re-run the E-step

4 Stopping Criteria

Stop when any holds:

1. **Score convergence:** $\dfrac{\left|\text{Score}^{(t+1)} - \text{Score}^{(t)}\right|}{\left|\text{Score}^{(t)}\right|} < \epsilon$ (e.g., $\epsilon = 0.01$).
2. **Schema stability:** fewer than, say, 5% of items change labels and no structural move is accepted.
3. **Generalization plateaus:** held-out perplexity no longer improves.
4. **Max iterations:** e.g., 5–10 passes.