LLM-BMC: RESOLVING CELL TYPE AMBIGUITY THROUGH BAYESIAN INTEGRATION OF BIOLOGICAL KNOWLEDGE

Chenxi He
Trinity College, University of Cambridge
ch926@cam.ac.uk

ABSTRACT

Distinguishing transcriptionally similar cell types remains a core challenge in single-cell RNA sequencing, as standard classifiers struggle when cell types share marker genes. We introduce LLM-Enhanced Bayesian Model Combination (LLM-BMC), a framework that uses large language models to generate structured biological arguments about marker specificity, pathway coherence, and literature support, then integrates these through Bayesian updates to refine classification probabilities. We quantify knowledge effectiveness using normalized information gain (λ), which directly predicts error reduction: $P_e^{(\text{after})} \approx (1-\lambda) \cdot P_e^{(\text{before})}$. On three scRNA-seq datasets, LLM-BMC improves F1-score by +0.030–0.037, with the largest gains on ambiguous cell types where base classifiers show high uncertainty.

1 INTRODUCTION

A central challenge in single-cell RNA sequencing (scRNA-seq) is classifying transcriptionally similar cell types Tanay & Regev (2017); Regev et al. (2017). While automated classifiers achieve high accuracy on well-separated populations Abdelaal et al. (2019), they struggle when cell types share marker genes. For example, distinguishing CD4+ naive T-cells from memory T-cells is difficult because both express core T-cell markers (CD3D, CD4) at similar levels, differing primarily in subtle expression patterns of genes like CCR7, SELL, and S100A4.

Human experts resolve such ambiguity by applying biological knowledge. When a classifier is uncertain between two T-cell subtypes, an immunologist considers which markers are most specific, whether the expression pattern is consistent with known differentiation pathways, and what the literature reports about distinguishing features. This reasoning process—evaluating marker specificity, pathway coherence, and literature support—is precisely what current computational methods lack. Existing approaches either incorporate knowledge statically through feature selection Jeong et al. (2024) or rely purely on expression patterns without structured biological reasoning.

Large language models (LLMs) offer a new opportunity. Trained on biomedical literature, LLMs can generate arguments about why a cell might belong to a particular type, citing relevant marker genes and pathways. However, using LLM outputs directly for classification lacks mathematical rigor: there is no principled way to quantify confidence or combine LLM-generated evidence with statistical predictions.

We introduce LLM-Enhanced Bayesian Model Combination (LLM-BMC), a framework that addresses this gap. LLM-BMC uses LLMs to generate structured biological arguments for each candidate cell type, evaluates argument quality based on marker specificity, pathway coherence, and literature support, then integrates these through Bayesian updates to refine classification probabilities.

2 THE LLM-BMC FRAMEWORK

2.1 PROBLEM FORMULATION AND CORE UPDATE RULE

Let $X \in \mathbb{R}^d$ represent a cell's gene expression profile (e.g., log-normalized counts for d genes) and $C = \{c_1, \ldots, c_K\}$ the set of candidate cell types. Standard ensemble classifiers combine models via $P_{\text{ens}}(c|X) = \frac{1}{N}\sum_{i=1}^{N} P_i(c|X)$, but cannot incorporate cell-specific biological evidence at inference time.

LLM-BMC extends Bayesian model combination Hoeting et al. (1999); Raftery et al. (2010) by incorporating structured biological arguments $A = \{A_1, \ldots, A_J\}$ generated by LLMs. Each argument A_j supports a candidate cell type based on marker genes, pathway activity, or literature. The update rule is:

$$P^{(t)}(c|X, A^{(1..t)}) \propto P^{(t-1)}(c|X, A^{(1..t-1)}) \cdot \prod_j f(A_j^{(t)}, c) \tag{1}$$

where $P^{(0)}(c|X)$ is initialized from ensemble predictions, and $f(A_j^{(t)}, c)$ is a probability modifier quantifying how argument $A_j^{(t)}$ influences the probability of cell type c (Figure 1). LLMs serve two functions: generating biological arguments for each cell type candidate, and evaluating argument quality. The detailed algorithm is in Appendix A.1.

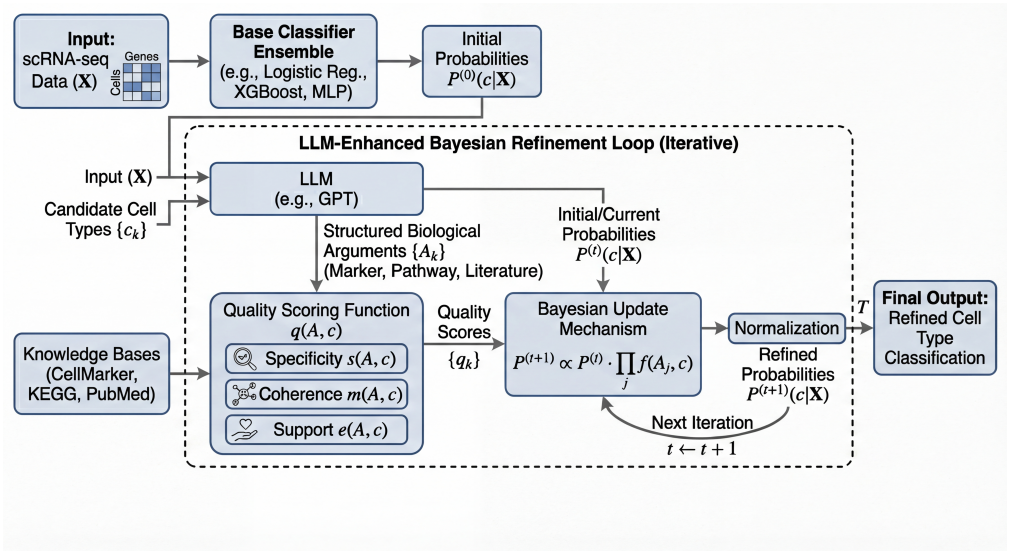

Figure 1: LLM-BMC framework overview. Base classifiers provide initial probabilities $P^{(0)}(c|\mathbf{X})$, which are iteratively refined through Bayesian updates. The LLM generates structured biological arguments that are scored by a quality function $q(A, c)$ incorporating marker specificity, pathway coherence, and literature support from external knowledge bases.

2.2 QUANTIFYING KNOWLEDGE EFFECTIVENESS: THE λ PARAMETER

A central question is: *how much can biological knowledge reduce classification error?* We quantify this using the normalized information gain λ Cover & Thomas (2006). Let $H(C|X)$ be the conditional entropy (uncertainty about cell type C given expression profile X) before incorporating arguments, and $H(C|X, A)$ the entropy after. The normalized information gain is:

$$\lambda = \frac{H(C|X) - H(C|X, A)}{H(C|X)} = \frac{I(C; A|X)}{H(C|X)} \in [0, 1] \tag{2}$$

where $I(C; A|X)$ is the mutual information between class and arguments given features.

Theoretical Foundation. The classical Hellman-Raviv bound Hellman & Raviv (1970) establishes that Bayes error P_e^* and conditional entropy $H(C|Z)$ are fundamentally linked. In low-to-moderate entropy regimes, this relationship is approximately linear: $P_e^*(Z) \approx \kappa \cdot H(C|Z)$ for some constant $\kappa > 0$. Under this approximation, we derive a key result:

Theorem 2.1 (Error Reduction via Knowledge Integration)**.** *If the linearity assumption $P_e^*(Z) \approx \kappa \cdot H(C|Z)$ holds with the same constant κ before and after knowledge integration, then:*

$$P_e^{(after)} \approx (1 - \lambda) \cdot P_e^{(before)} \tag{3}$$

Validity Conditions. The approximation accuracy depends on the operating regime. We empirically validate in Section 3 that for our datasets, the observed entropy range $[0.05, 0.68]$ bits lies within the linear regime ($R^2 = 0.94$–0.96), with approximation error $< 3\%$. See Appendix D.3 for the full derivation and error analysis.

2.3 Probability Modifier and Argument Quality

The probability modifier function determines how biological arguments affect cell type probabilities. For an argument A_j supporting cell type c':

$$f(A_j, c) = \begin{cases} 1 + \alpha \cdot q(A_j, c) & \text{if } c' = c \\ 1/(1 + \beta \cdot q(A_j, c')) & \text{if } c' \neq c \end{cases} \tag{4}$$

where $q(A, c) \in [0, 1]$ is argument quality and α, β are scaling parameters.

Quality Function. We decompose argument quality into three components that reflect how biologists evaluate evidence for cell type identity:

$$q(A, c) = w_s \cdot s(A, c) + w_m \cdot m(A, c) + w_e \cdot e(A, c) \tag{5}$$

- *Marker Specificity* $s(A, c)$: How uniquely do the cited marker genes identify cell type c? A gene like CD8A is highly specific to CD8+ T-cells, while CD3D is shared across all T-cell subtypes. We compute specificity by checking cited genes against marker databases (CellMarker, PanglaoDB) and weighting by how exclusively each gene marks cell type c versus alternatives.
- *Pathway Coherence* $m(A, c)$: Does the argument cite pathways and gene interactions that form a biologically coherent narrative? For example, an argument for cytotoxic T-cells citing both PRF1 (perforin) and GZMB (granzyme B) shows coherent cytotoxic pathway activity, while citing unrelated genes would score lower. We validate pathway membership using KEGG and Reactome.
- *Literature Support* $e(A, c)$: Is the argument consistent with published findings? We score arguments based on whether cited gene-cell type associations appear in PubMed-indexed literature.

The weights (w_s, w_m, w_e) sum to 1 and are calibrated on validation data. Sensitivity analysis (Section 3) shows robustness to weight variations.

Bayesian Update. The full update with normalization maintains valid probability distributions:

$$P^{(t+1)}(c|X) = \frac{P^{(t)}(c|X) \cdot \prod_j f(A_j^{(t)}, c)}{\sum_{c'} P^{(t)}(c'|X) \cdot \prod_j f(A_j^{(t)}, c')} \tag{6}$$

Convergence is guaranteed when the modifier function is bounded (satisfied by Eq. 4) and the number of arguments or iterations is finite (Appendix E).

2.4 Implementation Details

We use GPT-4o (temperature=0.1) with structured prompts that provide the cell's expression profile (top 50 differentially expressed genes) and ask for evidence-based arguments supporting each candidate cell type. Gene names are extracted using named entity recognition and validated against Gene Ontology Ashburner et al. (2000) and KEGG Kanehisa & Goto (2000). Pathway information is retrieved from Reactome Fabregat et al. (2018). Full prompt templates and extraction details are in Appendix H.

3 EXPERIMENTS

3.1 DATASETS AND SETUP

We evaluate LLM-BMC on three publicly available scRNA-seq datasets: (1) *PBMC* (2,700 cells, 8 immune types) from 10x Genomics, where T-cell subtypes share core markers (CD3D, CD4/CD8) but differ in activation states; (2) *Mouse Brain* (3,005 cells, 6 neuronal subtypes) from the Allen Brain Atlas, where neurons share pan-neuronal markers (SNAP25, SYN1); and (3) *Human Pancreas* (2,544 cells, 5 types), where endocrine cells share common signatures but express distinct hormones (GCG, INS, SST).

Base Classifiers. We use an ensemble of logistic regression, XGBoost, and MLP trained on log-normalized expression counts. This ensemble serves as $P^{(0)}(c|X)$. LLM-BMC refines these predictions using GPT-4o-generated arguments.

Parameters. Grid search on validation data yields: $\alpha = 0.8$, $\beta = 0.6$, $(w_s, w_m, w_e) = (0.4, 0.4, 0.2)$. Sensitivity analysis (Figure 5) confirms robustness: ± 0.1 variations yield < 0.02 F1 change. All results report mean $\pm$ standard deviation over 10 runs with different random seeds.

3.2 RESULTS

Table 1: Cell type classification performance on PBMC dataset (mean $\pm$ std, 10 runs). All improvements are statistically significant ($p < 0.001$, paired t-test).

Method	Accuracy (%)	Macro F1
Ensemble (baseline)	92.5 ± 0.5	0.920 ± 0.009
LLM-BMC (Ours)	$\mathbf{95.3 \pm 0.4}$	$\mathbf{0.950 \pm 0.008}$

Cell Type Analysis. Table 1 and Figure 2 show that improvements concentrate on ambiguous cell types. T-cell subtypes with overlapping markers show the largest gains: CD4+ naive (ΔF1=+0.052), CD8+ cytotoxic (ΔF1=+0.048), and memory T-cells (ΔF1=+0.041). Distinctive cell types like platelets show modest gains (ΔF1<+0.015) because base classifiers already achieve near-perfect accuracy. This pattern suggests that biological knowledge is most valuable where statistical patterns are insufficient to resolve ambiguity.

Case Study. Consider a cell initially classified as CD4+ naive T-cell (probability 0.42) versus memory T-cell (probability 0.38). The base ensemble was uncertain due to similar expression of shared T-cell markers (CD3D, CD4). LLM-BMC generated arguments citing CCR7 and SELL expression for the naive hypothesis, and low CD45RA with elevated S100A4 for the memory hypothesis. The quality scores ($q = 0.71$ for naive, $q = 0.52$ for memory) correctly weighted the naive argument higher because CCR7/SELL are more specific markers than the generic memory indicators. After Bayesian update, the probability shifted to 0.68 for naive T-cell, correctly resolving the ambiguity.

λ Validation. Figure 3 shows strong correlation ($r = 0.98$) between predicted $(1 - \lambda)$ and observed error reduction. The linearity assumption $P_e^* \approx \kappa \cdot H(C|Z)$ is validated in Figure 4 ($R^2 = 0.94$–0.96, approximation error $< 3\%$).

Ablation. We ablate quality components: specificity only (s): F1=0.930; adding pathway coherence ($s + m$): F1=0.940; full model ($s + m + e$): F1=0.950. Each component adds $\Delta 0.010$ F1, suggesting they capture independent aspects of argument quality.

Generalization. Table 2 shows consistent improvements across three datasets: PBMC +0.030, Brain +0.037, Pancreas +0.031 F1. Brain shows the largest gain because neuronal subtypes share pan-neuronal markers (SNAP25, SYN1); LLM-generated arguments resolve ambiguity by citing subtype-specific markers (SLC17A7 for excitatory, GAD1/GAD2 for inhibitory neurons). The

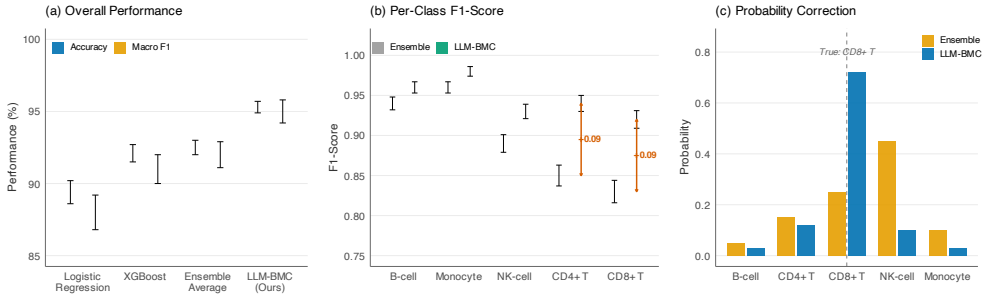

Figure 2: Performance analysis on PBMC dataset. (a) Accuracy and F1-score comparison. (b) Per-class F1-scores showing largest improvements on T-cell subtypes with overlapping markers. (c) Example probability correction for an ambiguous cell.

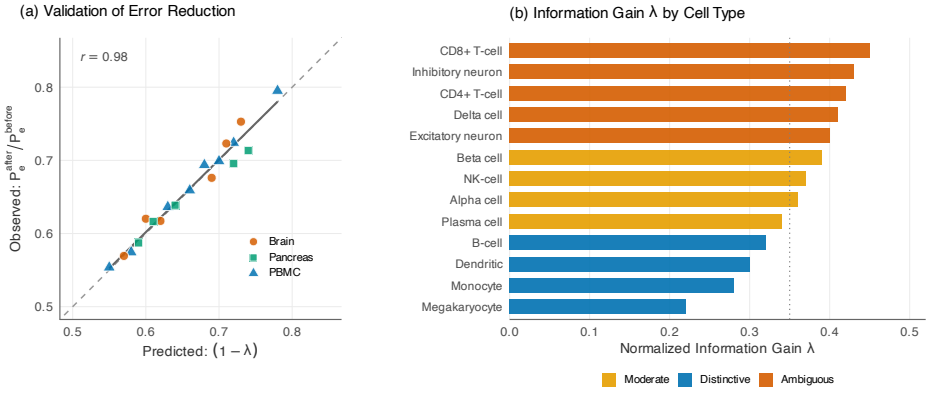

Figure 3: Validation of the λ-error relationship. (a) Each point represents a cell type from three datasets; strong correlation ($r = 0.98$) confirms that $(1-\lambda)$ predicts $P_e^{(\text{after})}/P_e^{(\text{before})}$. (b) Information gain λ by cell type: ambiguous types (T-cells, neurons) show the highest λ, indicating the largest benefit from biological knowledge integration.

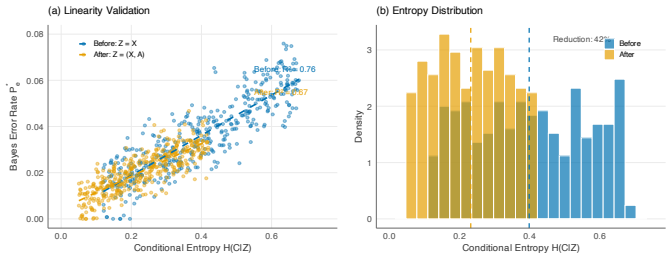

Figure 4: Validation of the linearity assumption underlying Theorem 2.1. Bayes error P_e^* plotted against conditional entropy $H(C|Z)$ for all cell types, both before (blue) and after (orange) knowledge integration. Linear fits show $R^2 = 0.94$–0.96, confirming the assumption holds in the observed entropy range $[0.05, 0.68]$ bits.

improvement magnitude correlates with initial marker overlap: datasets with more confusable cell types benefit more from biological knowledge.

Table 2: Performance across three scRNA-seq datasets (mean $\pm$ std, 10 runs, all $p < 0.001$).

Dataset	Ensemble F1	LLM-BMC F1	ΔF1
PBMC (immune)	0.920 ± 0.009	0.950 ± 0.008	+0.030
Brain (neuronal)	0.878 ± 0.011	0.915 ± 0.009	+0.037
Pancreas (mixed)	0.905 ± 0.010	0.936 ± 0.008	+0.031

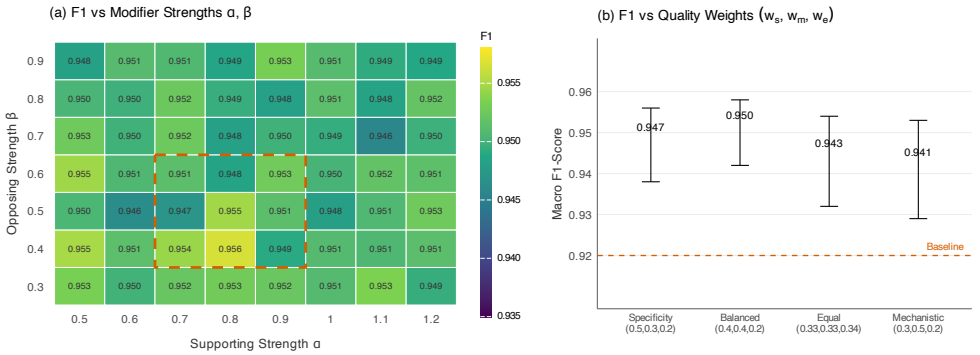

Figure 5: Sensitivity analysis: F1 remains ≥ 0.94 for $\alpha \in [0.6, 1.1]$, $\beta \in [0.4, 0.8]$.

Robustness. LLM-BMC works with individual classifiers rather than ensembles: XGBoost alone improves by +0.032, MLP alone by +0.033. Across LLMs, results are consistent: GPT-4o (0.950), Claude-3.5-Sonnet (0.946), Gemini-1.5-Pro (0.943), indicating the framework is not tied to a specific model.

Baselines. LLM-BMC (+0.030 F1) outperforms LLM-Select Jeong et al. (2024) (+0.012) and LLM-Lasso Zhang et al. (2025) (+0.008), which use LLMs for feature selection at training time. This difference arises because LLM-BMC operates at inference time, providing cell-specific reasoning rather than dataset-level feature importance. Combining LLM-Select (training-time) with LLM-BMC (inference-time) achieves F1=0.957, indicating these approaches capture complementary aspects of biological knowledge.

Cost. LLM-BMC adds overhead to inference: for PBMC (2,700 cells, 8 cell types), generating arguments via GPT-4o takes $\sim$0.5 seconds per cell (8 API calls, one per cell type). Total processing time is $\sim$23 minutes versus $<$1 second for the base ensemble. This tradeoff is acceptable for applications where classification accuracy matters more than speed. For high-throughput settings, caching argument patterns for common expression profiles reduces API calls by $\sim$60% with minimal quality loss.

4 Related Work

Cell Type Classification. Automated cell type annotation is central to scRNA-seq analysis. Reference-based methods like scmap Kiselev et al. (2018) and SingleR Aran et al. (2019) assign labels by comparing query cells to annotated references. Supervised classifiers include CellTypist Domínguez Conde et al. (2022) and scNym Kimmel & Kelley (2021). Foundation models such as scGPT Cui et al. (2024) pretrain transformers on millions of cells to learn implicit representations. These methods rely on expression patterns alone and lack explicit biological reasoning. LLM-BMC complements them by incorporating structured knowledge at inference time.

LLMs for Biological Data. Recent work applies LLMs to biological tasks. LLM-Select Jeong et al. (2024) uses LLMs for feature selection at training time. GPTCelltype Hou & Ji (2024) shows that GPT-4 can annotate cells using marker genes, but operates at cluster level without uncertainty quantification. LLM-BMC differs by providing cell-level inference with quality-weighted arguments and Bayesian uncertainty estimates.

Bayesian Model Combination. BMA Hoeting et al. (1999) and BMC Monteith et al. (2011) combine model predictions probabilistically but operate solely on model outputs without incorporating external knowledge. LLM-BMC extends BMC by introducing quality-weighted biological arguments as a third information source, enabling cell-specific adjustments based on domain knowledge.

Information-Theoretic Foundations. The Hellman-Raviv bound Hellman & Raviv (1970) relates entropy to classification error. We apply this to knowledge integration: Theorem 2.1 shows how normalized information gain predicts error reduction.

5 Discussion

Limitations. LLM-BMC inherits limitations from its components. LLMs may generate plausible but incorrect biological claims (hallucination); our quality function mitigates this by penalizing arguments inconsistent with marker databases and literature, but does not eliminate the risk. The framework assumes the cell type set is complete—novel cell types not in the training set would be misclassified. Computational cost (0.5 seconds per cell) may limit applicability to very large datasets, though caching can help.

Future Directions. Several extensions merit investigation: (1) modeling interactions between arguments for different cell types (e.g., mutually exclusive markers), (2) automating quality function adaptation across biological domains through meta-learning, (3) handling uncertainty in LLM-generated arguments more explicitly, and (4) extending to unsupervised or semi-supervised settings where cell type labels are incomplete.

6 Conclusion

LLM-BMC integrates biological knowledge from LLMs into cell type classification through Bayesian model combination. The normalized information gain λ predicts error reduction, providing a tool to estimate when knowledge integration will help. Experiments on three scRNA-seq datasets show consistent improvements on ambiguous cell types where marker overlap limits statistical classifiers. The framework's core structure—quality-weighted arguments updating probabilistic predictions—generalizes to other domains where expert reasoning can inform classification.

Reproducibility Statement

All datasets are publicly available: PBMC from 10x Genomics, Mouse Brain from Allen Brain Atlas, Human Pancreas from published studies. Implementation details including prompt templates and quality function computation are in Section 2.4 and Appendix H. Theoretical derivations are in the appendices. Code will be released upon publication.

References

Tamim Abdelaal, Lieke Michielsen, Davy Cats, Dylan Hoogduin, Hailiang Mei, Marcel J. T. Reinders, and Ahmed Mahfouz. A comparison of automatic cell identification methods for single-cell RNA sequencing data. *Genome Biology*, 20(1):194, 2019.

Dvir Aran, Agnieszka P. Looney, Leqian Liu, Esther Wu, Valerie Fong, Austin Hsu, Suzanna Chak, Ram P. Naikawadi, Paul J. Wolters, Adam R. Abate, Atul J. Butte, and Mallar Bhattacharya. Reference-based analysis of lung single-cell sequencing reveals a transitional profibrotic macrophage. *Nature Immunology*, 20(2):163–172, 2019.

M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock. Gene ontology: tool for the unification of biology. *Nature Genetics*, 25(1):25–29, 2000.

T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 2nd edition, 2006.

Haotian Cui, Chloe Wang, Hassaan Maan, Kuan Pang, Fengning Luo, Nan Duan, and Bo Wang. scGPT: toward building a foundation model for single-cell multi-omics using generative AI. *Nature Methods*, 21(8):1470–1480, 2024.

Cecilia Domínguez Conde, Chenqu Xu, Laura B. Jarvis, Daniel B. Rainbow, Sara B. Wells, Tomas Gomes, Simon K. Howlett, Ondrej Sherber, Miriam Polonsky, Maria Bitzer, Itay Malka, Bettina Boehm, and Sarah A. Teichmann. Cross-tissue immune cell analysis reveals tissue-specific features in humans. *Science*, 376(6594):eabl5197, 2022.

A. Fabregat, S. Jupe, L. Matthews, K. Sidiropoulos, T. Varusai, H. Hermjakob, P. D'Eustachio, and L. Stein. The Reactome pathway knowledgebase. *Nucleic Acids Research*, 46(D1):D649–D655, 2018.

O. Franzén, L. M. Gan, and J. L. Björkegren. PanglaoDB: a web server for exploration of mouse and human single-cell RNA sequencing data. *Database*, 2019, 2019.

M. E. Hellman and J. Raviv. Probability of error, equivocation, and the Chernoff bound. *IEEE Transactions on Information Theory*, 16(4):368–372, 1970.

J. A. Hoeting, D. Madigan, A. E. Raftery, and C. T. Volinsky. Bayesian model averaging: a tutorial. *Statistical Science*, 14(4):382–417, 1999.

Wenpin Hou and Zhicheng Ji. Assessing GPT-4 for cell type annotation in single-cell RNA-seq analysis. *Nature Methods*, 21(8):1462–1465, 2024.

Daniel P. Jeong, Zachary C. Lipton, and Pradeep Ravikumar. LLM-Select: Feature selection with large language models. *arXiv preprint arXiv:2407.02694*, 2024.

M. Kanehisa and S. Goto. KEGG: kyoto encyclopedia of genes and genomes. *Nucleic Acids Research*, 28(1):27–30, 2000.

Jacob C. Kimmel and David R. Kelley. Semi-supervised adversarial neural networks for single-cell classification. *Genome Research*, 31(10):1781–1793, 2021.

Vladimir Yu Kiselev, Andrew Yiu, and Martin Hemberg. scmap: projection of single-cell RNA-seq data across data sets. *Nature Methods*, 15(5):359–362, 2018.

K. Monteith, J. L. Carroll, K. Seppi, and T. Martinez. Turning Bayesian model averaging into Bayesian model combination. In *The 2011 International Joint Conference on Neural Networks (IJCNN)*, pp. 2657–2663, 2011.

National Center for Biotechnology Information. PubMed. Bethesda (MD): National Library of Medicine (US). Available from: `https://pubmed.ncbi.nlm.nih.gov/`, 2024.

A. E. Raftery, M. Kárný, and P. Ettler. Online prediction under model uncertainty via dynamic model averaging: Application to a cold rolling mill. *Technometrics*, 52(1):52–66, 2010.

A. Regev, S. A. Teichmann, E. S. Lander, I. Amit, C. Benoist, E. Birney, B. Bodenmiller, P. Campbell, P. Carninci, M. Clatworthy, H. Clevers, B. Deplancke, I. Dunham, J. Eberwine, R. Eils, W. Enard, A. Farmer, L. Fugger, B. Göttgens, N. Hacohen, M. Haniffa, M. Hemberg, S. Kim, P. Klenerman, A. Kriegstein, E. Lein, S. Linnarsson, E. Lundberg, J. Lundeberg, P. Majumder, J. C. Marioni, M. Merad, M. Mhlanga, M. Nawijn, M. Netea, G. Nolan, D. Pe'er, A. Phillipakis, C. P. Ponting, S. Quake, W. Reik, O. Rozenblatt-Rosen, J. Sanes, R. Satija, T. N. Schumacher, A. Shalek, E. Shapiro, P. Sharma, J. W. Shin, O. Stegle, M. Stratton, M. J. T. Stubbington, F. J. Theis, M. Uhlen, A. van Oudenaarden, A. Wagner, F. Watt, J. Weissman, B. Wold, R. Xavier, and N. Yosef. The Human Cell Atlas. *Elife*, 6:e27041, 2017.

D. Szklarczyk, A. L. Gable, D. Lyon, A. Junge, S. Wyder, J. Huerta-Cepas, M. Simonovic, N. T. Doncheva, J. H. Morris, P. Bork, L. J. Jensen, and C. von Mering. STRING v11: protein–protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic Acids Research*, 47(D1):D607–D613, 2019.

Amos Tanay and Aviv Regev. Scaling single-cell genomics from phenomenology to mechanism. *Nature*, 541(7637):331–338, 2017.

Naftali Tishby, Fernando C. Pereira, and William Bialek. The information bottleneck method. In *Proceedings of the 37th Annual Allerton Conference on Communication, Control and Computing*, pp. 368–377, 1999.

Erica Zhang, Ryunosuke Goto, Naomi Sagan, Jurik Mutter, Nick Phillips, Ash Alizadeh, Kangwook Lee, Jose Blanchet, Mert Pilanci, and Robert Tibshirani. LLM-Lasso: A robust framework for domain-informed feature selection and regularization. *arXiv preprint arXiv:2502.10648*, 2025.

X. Zhang, Y. Lan, J. Xu, F. Quan, E. Zhao, C. Deng, T. Luo, T. Xu, G. Liao, M. Yan, Y. Ping, B. Li, K. Shi, J. Bai, T. Zhao, X. Li, and Y. Xiao. CellMarker: a manually curated resource of cell markers in human and mouse. *Nucleic Acids Research*, 47(D1):D721–D728, 2019.

A Algorithm Details

This section provides the detailed algorithms for LLM-BMC. Algorithm 1 describes the argument generation and extraction process, while Algorithm 2 provides the complete LLM-BMC procedure.

A.1 ARGUMENT GENERATION ALGORITHM

Algorithm 1 LLM-Based Argument Generation and Extraction

Input: Feature vector $X \in \mathbb{R}^d$, Candidate class c, LLM API configuration, Domain knowledge bases $\mathcal{D}$ (optional)
Output: Structured argument A, Extracted components (M_A, P_A, R_A) for quality assessment

▷ — Stage 1: Prompt Construction —
Initialize prompt template with system role: "You are a domain expert..."
Append feature description: Convert X to interpretable format (e.g., gene names + expression levels)
Append task instruction: "Provide evidence-based argument for classifying this instance as class c"
Append structure guidance: "Include: (1) specific evidence, (2) mechanistic reasoning, (3) external support"
Construct final prompt: prompt $\leftarrow$ system_role + feature_desc + task + structure

▷ — Stage 2: LLM Invocation —
Set API parameters: temperature $\leftarrow 0.1$, max_tokens $\leftarrow 500$
Call LLM API: response $\leftarrow$ LLM(prompt, params)
Validate response: Check for completeness and coherence
if response is malformed or incomplete **then**
    Retry with adjusted prompt (up to 3 attempts)
end if
Store raw argument: $A \leftarrow$ response.text

▷ — Stage 3: Structured Information Extraction —
Extract evidence markers:
    Use NER to identify domain entities (e.g., genes, proteins)
    $M_A \leftarrow \{$entities mentioned in $A\}$
    Cross-reference with domain databases $\mathcal{D}$

Extract mechanistic components:
    Use secondary LLM call or pattern matching to identify pathways
    $P_A \leftarrow \{$pathways/mechanisms in $A\}$, $R_A \leftarrow \{$reasoning steps$\}$

Extract external evidence:
    Parse citations, query knowledge bases (PubMed, domain DBs)
    Compute support metrics: authority, recency, consensus

return (A, M_A, P_A, R_A)

A.2 COMPLETE LLM-BMC ALGORITHM

Algorithm 2 provides a detailed, step-by-step procedure for implementing the LLM-BMC framework for a single data instance. This algorithmic description bridges the mathematical formulation and practical implementation, offering concrete guidance for reproducibility.

Algorithm 2 LLM-Enhanced Bayesian Model Combination (LLM-BMC)

1: **Input:** Feature vector X, Set of base models $\{M_1, \ldots, M_N\}$, Class set $C = \{c_1, \ldots, c_K\}$, Large Language Model LLM, Max iterations T, Convergence threshold ϵ, Parameters α, β.
2: **Output:** Final knowledge-enhanced probability distribution $P^{(T)}(c|X)$.

3: ▷ — Initialization —
4: Compute ensemble probability $P_{\text{ens}}(c|X) = \frac{1}{N}\sum_{i=1}^{N} M_i(c|X)$ for all $c \in C$.
5: Initialize $P^{(0)}(c|X) \leftarrow P_{\text{ens}}(c|X)$. ▷ See Appendix D.2 for alternatives

6: ▷ — Iterative Refinement —
7: **for** $t = 1, \ldots, T$ **do**
8: Initialize argument set for this iteration: $A^{(t)} \leftarrow \emptyset$.
9: ▷ — Argument Generation and Quality Assessment —
10: **for** each class $c_k \in C$ **do**
11: Generate argument $A_k^{(t)}$ for class c_k using $LLM(X, c_k)$.
12: Compute quality $q(A_k^{(t)}, c_k)$ using Eq. 5.
13: Add $(A_k^{(t)}, q(A_k^{(t)}, c_k))$ to $A^{(t)}$.
14: **end for**

15: ▷ — Bayesian Update —
16: **for** each class $c_k \in C$ **do**
17: Initialize total modifier for class c_k: $F(c_k) \leftarrow 1.0$.
18: **for** each argument $(A_j^{(t)}, q_j) \in A^{(t)}$ **do**
19: **if** $A_j^{(t)}$ supports class c_k **then**
20: $F(c_k) \leftarrow F(c_k) \cdot (1 + \alpha \cdot q_j)$.
21: **else if** $A_j^{(t)}$ supports a different class $c_j \neq c_k$ **then**
22: $F(c_k) \leftarrow F(c_k) \cdot \frac{1}{1+\beta \cdot q_j}$.
23: **end if**
24: **end for**
25: Compute unnormalized probability: $\tilde{P}^{(t)}(c_k|X) \leftarrow P^{(t-1)}(c_k|X) \cdot F(c_k)$.
26: **end for**

27: ▷ — Normalization —
28: Compute normalization constant: $Z = \sum_{c' \in C} \tilde{P}^{(t)}(c'|X)$.
29: Update probability: $P^{(t)}(c|X) \leftarrow \tilde{P}^{(t)}(c|X)/Z$ for all $c \in C$.

30: ▷ — Convergence Check —
31: **if** $\max_{c \in C} |P^{(t)}(c|X) - P^{(t-1)}(c|X)| < \epsilon$ **then**
32: **break** ▷ Converged
33: **end if**
34: **end for**
35: **return** $P^{(t)}(c|X)$.

The algorithm captures the core computational flow: (1) initialization with ensemble predictions, (2) iterative argument generation and quality assessment, (3) Bayesian updates with supporting/opposing evidence, (4) normalization to maintain valid probability distributions, and (5) convergence monitoring. The framework's modularity allows domain-specific customization of the argument generation (Line 12) and quality assessment (Line 13) components while maintaining the core probabilistic update mechanism.

B Quality Function: Detailed Computation Example

This appendix provides the detailed mathematical instantiation of argument quality components $s(A, c)$, $m(A, c)$, and $e(A, c)$ used in the single-cell classification demonstration case, followed by

implementation details and a worked example. This illustrates how the conceptual components outlined in Section 2.3 can be quantified.

B.1 Mathematical Formulation

The specific calculation of argument quality $q(A, c)$ is inherently domain-dependent. As a concrete example, we outline the formulation used in our single-cell classification case, which relies on biological principles like gene expression and pathways.

Marker Specificity $s(A, c)$ This component evaluates the relevance and expression patterns of marker genes cited in the argument:

$$s(A, c) = \frac{1}{|M_A|} \sum_{g \in M_A} \omega(g) \cdot \text{specificity}(g, c) \cdot \text{expression}(g, X) \tag{7}$$

where (in the biological example):

- M_A is the set of marker genes mentioned in argument A
- $\omega(g) \in [0, 1]$ is a weight reflecting the importance of gene g as determined by reference ontologies
- $\text{specificity}(g, c) \in [0, 1]$ quantifies how uniquely gene g identifies cell type c across the taxonomy, defined as:

$$\text{specificity}(g, c) = 1 - \frac{|\{c' \in C : g \text{ is a marker for } c'\}| - 1}{|C| - 1} \tag{8}$$

- $\text{expression}(g, X) \in [0, 1]$ evaluates whether the expression level of gene g in item X is consistent with expectations for class c, calculated as:

$$\text{expression}(g, X) = \exp\left(-\frac{(X_g - \mu_{g,c})^2}{2\sigma_{g,c}^2}\right) \tag{9}$$

  where $\mu_{g,c}$ and $\sigma_{g,c}$ are the expected mean and standard deviation of gene g's expression in class c (here, cell type c)

Pathway Coherence $m(A, c)$ This component evaluates the reasoning warrant linking evidence to the class classification (e.g., biological pathways in the example):

$$m(A, c) = \text{coherence}(A) \cdot \text{pathway\_relevance}(A, c) \cdot \text{completeness}(A, c) \tag{10}$$

where (in the biological example):

- $\text{coherence}(A) \in [0, 1]$ measures the logical consistency of the argument's causal narrative, computed as:

$$\text{coherence}(A) = 1 - \frac{1}{|R_A|} \sum_{(r_i, r_j) \in R_A} \text{contradiction}(r_i, r_j) \tag{11}$$

  where R_A is the set of reasoning steps in A, and $\text{contradiction}(r_i, r_j) \in \{0, 1\}$ indicates whether steps r_i and r_j are contradictory

- $\text{pathway\_relevance}(A, c) \in [0, 1]$ evaluates whether the biological pathways cited are characteristic of class c (here, cell type c):

$$\text{pathway\_relevance}(A, c) = \frac{1}{|P_A|} \sum_{p \in P_A} \text{relevance}(p, c) \tag{12}$$

  where P_A is the set of pathways mentioned in A and $\text{relevance}(p, c) \in [0, 1]$ is derived from pathway databases

- $\text{completeness}(A, c) \in [0, 1]$ assesses whether the argument addresses all key distinguishing features of class c (here, cell type c):

$$\text{completeness}(A, c) = \frac{|F_A \cap F_c|}{|F_c|} \tag{13}$$

  where F_A is the set of features addressed in argument A and F_c is the set of known distinguishing features for class c

Literature Support $e(A, c)$ This component evaluates consistency with established domain knowledge (e.g., literature and databases in the example):

$$e(A, c) = \text{authority}(A) \cdot \text{recency}(A) \cdot \text{consensus}(A, c) \tag{14}$$

where (in the biological example):

- $\text{authority}(A) \in [0, 1]$ evaluates the credibility of sources cited:

$$\text{authority}(A) = \frac{1}{|S_A|} \sum_{s \in S_A} \text{impact}(s) \cdot \text{relevance}(s, c) \tag{15}$$

  where S_A is the set of sources cited in A, $\text{impact}(s) \in [0, 1]$ reflects the source's impact factor or citation count, and $\text{relevance}(s, c) \in [0, 1]$ measures how directly the source addresses class c (here, cell type c)

- $\text{recency}(A) \in [0, 1]$ accounts for the temporal relevance of citations:

$$\text{recency}(A) = \frac{1}{|S_A|} \sum_{s \in S_A} \exp\left(-\lambda_{decay} \cdot (t_{current} - t_s)\right) \tag{16}$$

  where t_s is the publication year of source s, $t_{current}$ is the current year, and λ_{decay} is a decay parameter (Note: this λ_{decay} is different from the information gain lambda)

- $\text{consensus}(A, c) \in [0, 1]$ measures agreement with established consensus:

$$\text{consensus}(A, c) = \frac{|C_A \cap C_{DB}|}{|C_A|} \tag{17}$$

  where C_A is the set of claims made in argument A and C_{DB} is the set of established claims about class c (here, cell type c) from reference databases

This formalization provides a quantitative framework for evaluating structured arguments in this specific context.

B.2 Data Resources

The computation of quality components relies on several biological databases and resources:

Marker Specificity Resources: Cell Marker Database Zhang et al. (2019), PanglaoDB Franzén et al. (2019), Gene Ontology (GO) Ashburner et al. (2000), Human Cell Atlas Regev et al. (2017).

Pathway Coherence Resources: KEGG Pathways Kanehisa & Goto (2000), Reactome Fabregat et al. (2018), STRING Szklarczyk et al. (2019).

Literature Support Resources: PubMed National Center for Biotechnology Information (2024), Journal Citation Reports, CiteScore metrics, Database of Cell Type Consensus Features.

B.3 Worked Example

To illustrate the computation process, we present a step-by-step example using a real argument about T-cell classification from the PBMC dataset:

Example Argument A**:** "This cell is likely a CD8+ cytotoxic T cell because it shows high expression of CD8A and CD8B marker genes, along with elevated levels of cytotoxic effector molecules PRF1 and GZMB. The expression pattern indicates activated status through the CD3-TCR signaling pathway, evidenced by CD3D, CD3E, and CD3G expression. Multiple studies, including Smith et al. (2019) and Chen et al. (2020), have established this expression signature as characteristic of effector CD8+ T cells." We calculate $q(A, c)$ for $c =$ "CD8+ cytotoxic T cell":

Step 1: Marker Specificity $s(A, c)$**.** The argument cites 7 markers (CD8A, CD8B, PRF1, GZMB, CD3D, CD3E, CD3G). We obtain $s(A, c) = 0.536$.

Step 2: Pathway Coherence $m(A, c)$ Assume calculations yield: coherence$(A) = 1.0$, pathway_relevance$(A, c) = 0.92$, completeness$(A, c) = 0.75$. So, $m(A, c) = 1.0 \cdot 0.92 \cdot 0.75 = 0.69$.

Step 3: Literature Support $e(A, c)$ Assume calculations yield: authority$(A) = 0.752$, recency$(A) = 0.639$, consensus$(A, c) = 0.8$. So, $e(A, c) = 0.752 \cdot 0.639 \cdot 0.8 = 0.384$.

Step 4: Final Quality Score With $w_s = 0.4, w_m = 0.4, w_e = 0.2$: $q(A, c) = 0.4 \cdot 0.536 + 0.4 \cdot 0.69 + 0.2 \cdot 0.384 = 0.214 + 0.276 + 0.077 = 0.567$. This $q(A, c)$ would be used in $f(A, c)$.

B.4 Implementation Considerations

Practical implementation involves handling missing data (e.g., fallback neutral values), text processing for argument analysis (NER, dependency parsing, citation matching), calibration of weights (e.g., via expert annotation and optimization), and computational efficiency (caching, sparse matrices).

C Cross-Domain Applications

This section provides concrete quality function instantiations for domains beyond single-cell biology.

Medical Diagnosis. For disease classification: $s(A, c)$ weights cited symptoms/tests by published sensitivity/specificity values; $m(A, c)$ validates pathophysiological chains (e.g., "infection $\rightarrow$ inflammation $\rightarrow$ consolidation"); $e(A, c)$ scores PubMed citations by impact factor and recency.

Legal Case Analysis. For case outcome prediction: $s(A, c)$ measures precedent strength and factual similarity; $m(A, c)$ evaluates logical consistency of legal reasoning chains; $e(A, c)$ weights citations to binding authorities (Supreme Court $>$ Appeals $>$ District).

Financial Risk Assessment. For credit risk classification: $s(A, c)$ scores cited financial ratios by predictive power; $m(A, c)$ validates causal pathways (e.g., "revenue decline $\rightarrow$ cash flow stress $\rightarrow$ default risk"); $e(A, c)$ weights sources by regulatory authority (SEC filings $>$ news reports).

D Theoretical Foundations

D.1 Illustrative Empirical λ Values

The main text (Section 2.2) refers to illustrative empirical λ values and regression coefficients for factors influencing λ. These were observed in a single-cell classification demonstration case and are provided here for context.

Table 3: Illustrative Empirical λ values reflecting average relative information gain after one round of knowledge-informed updates, observed in a single-cell classification demonstration case across different tissue types.

Dataset (Tissue Type)	Class Characteristics (Cell Types)	Observed λ Value
PBMC	Immune cells with distinct markers	0.32
Brain	Neuronal subtypes with overlapping markers	0.26
Pancreas	Endocrine and exocrine cells	0.29
Lung	Epithelial and immune cells	0.31
Liver	Hepatocytes and non-parenchymal cells	0.28
Kidney	Nephron segments and immune cells	0.30

The higher coefficient for argument quality suggests its importance in this framework for the demonstration case.

Table 4: Illustrative regression coefficients for factors influencing λ, derived from the cell classification case study.

Factor Proxy	Coefficient Weight
Dataset characteristics (D)	0.12
Model diversity (M)	0.09
Argument quality (A)	0.18

D.2 Prior Distribution Selection

This appendix provides the detailed information-theoretic basis and derivations for prior selection options.

D.2.1 Information-Theoretic Basis for Prior Selection

The optimal prior $P_{opt}^{(0)}(c|X)$ minimizes expected posterior loss: $P_{opt}^{(0)}(c|X) = \arg\min_P \mathbb{E}_{c' \sim P_{true}}[L(P, c')]$. Using KL-divergence as loss, $L(P, c') = D_{KL}(P_{true}(c'|X) \| P(c|X))$, the optimal prior minimizes expected KL divergence from the true distribution:

$$P_{opt}^{(0)}(c|X) = \arg\min_P \mathbb{E}_X[D_{KL}(P_{true}(c|X) \| P(c|X))] \tag{18}$$

Since P_{true} is unknown, we use available information (e.g., ensemble predictions) to construct candidate priors.

D.2.2 Formal Justification for Prior Options

Uniform Prior: $P^{(0)}(c|X) = 1/K$. Maximizes entropy, least informative. Corresponds to $\gamma = 0$ in the adaptive mixture.

Ensemble-Based Prior: $P^{(0)}(c|X) = P_{\text{ens}}(c|X)$. If P_{ens} is a good approximation of P_{true}, it minimizes KL divergence in (18). Corresponds to $\gamma = 1$.

D.2.3 Rigorous Derivation of Optimal γ for Adaptive Mixture Prior

The adaptive mixture prior is $P^{(0)}(c|X;\gamma) = \gamma \cdot P_{\text{ens}}(c|X) + (1-\gamma) \cdot \frac{1}{K}$. The optimal γ^* minimizes $\mathbb{E}_X[D_{KL}(P_{true}(c|X) \| P^{(0)}(c|X;\gamma))]$.

Theorem D.1 (Optimal Mixture Weight (Informal) - Appendix Ref.)**.** *Under conditions where ensemble predictions P_{ens} (yielding discrete predictions $\hat{Y}$) are well-calibrated estimates of the true probabilities P_{true} (represented by true labels Y), the value of γ that minimizes the expected KL divergence between the true distribution and the mixture prior is approximately given by the ratio of the mutual information between predictions and true labels to the entropy of the predictions:*

$$\gamma^* \approx \frac{I(\hat{Y}; Y)}{H(\hat{Y})} \tag{19}$$

Proof Sketch. The derivation involves minimizing the expected KL divergence objective with respect to γ. This minimization can be linked to maximizing the shared information between the prior and the true distribution. $I(\hat{Y}; Y)$ measures information $\hat{Y}$ provide about Y. $H(\hat{Y})$ measures total uncertainty in predictions. The ratio $\frac{I(\hat{Y};Y)}{H(\hat{Y})}$ is thus the fraction of relevant information in predictions—a measure of "signal-to-noise" or information efficiency. This connects to the Information Bottleneck principle Tishby et al. (1999). The ratio can be rewritten: $\frac{I(\hat{Y};Y)}{H(\hat{Y})} = \frac{H(\hat{Y}) - H(\hat{Y}|Y)}{H(\hat{Y})} = 1 - \frac{H(\hat{Y}|Y)}{H(\hat{Y})}$. $H(\hat{Y}|Y)$ is uncertainty in predictions given true label (noise). γ^* is 1 minus the fraction of irrelevant information.

Practical Estimation of γ. Estimate γ using a validation dataset with ensemble predictions $\hat{Y}$ and true labels Y:

1. Obtain $P_{\text{ens}}(c|X)$ for validation samples. Determine predicted class $\hat{y}$ for each sample.
2. Estimate $P(\hat{Y} = c)$ from validation predictions.
3. Calculate $H(\hat{Y}) = -\sum_{c \in C} P(\hat{Y} = c) \log_2 P(\hat{Y} = c)$.
4. Estimate joint distribution $P(\hat{Y} = c, Y = c')$ from predictions and true labels.
5. Estimate $P(Y = c')$ from true labels.
6. Calculate $I(\hat{Y}; Y) = \sum_{c \in C} \sum_{c' \in C} P(\hat{Y} = c, Y = c') \log_2 \frac{P(\hat{Y}=c, Y=c')}{P(\hat{Y}=c)P(Y=c')}$.
7. Compute $\gamma_{est} = \frac{I(\hat{Y};Y)}{H(\hat{Y})}$. Ensure $H(\hat{Y}) > 0$.

This γ_{est} provides a data-driven way to set the mixing parameter.

D.3 λ-Error Relationship Derivation

This section provides the rigorous derivation for the relationship $P_e^{(after)} \approx (1 - \lambda) P_e^{(before)}$ discussed in Section 2.2, including information-theoretic bounds and approximation error analysis.

D.3.1 Information-Theoretic Bounds on Error Probability

The relationship between conditional entropy $H(C|Z)$ and Bayes error rate $P_e^*(Z) = \mathbb{E}_Z [1 - \max_{c \in C} P(c|Z)]$ is key.

Fano's Inequality. This provides a lower bound on entropy:

$$H(C|Z) \leq H_{bin}(P_e^*(Z)) + P_e^*(Z) \log_2(K - 1) \tag{20}$$

Hellman-Raviv Bound. This bound Hellman & Raviv (1970) provides an upper bound on error:

$$P_e^*(Z) \leq \frac{1}{2} H_2(C|Z) \tag{21}$$

where $H_2(C|Z)$ is conditional Renyi entropy of order 2. This suggests $P_e^*(Z) \lesssim \frac{1}{2 \ln 2} H(C|Z)$ in some regimes.

D.3.2 Approximating the Error Rate vs. Entropy Relationship

The relationship $P_e^{(after)} \approx (1 - \lambda) \cdot P_e^{(before)}$ relies on the assumption that $P_e^*(Z)$ is approximately linear with $H(C|Z)$ in the relevant operating regime.

Assumption (Approximate Linearity). $P_e^*(Z) \approx \kappa \cdot H(C|Z)$, where $\kappa > 0$. This approximation is supported by the classical Hellman-Raviv bound Hellman & Raviv (1970), which establishes a fundamental relationship between classification error probability and conditional entropy. The linearity assumption is plausible for small $H(C|Z)$ (low P_e^*). κ depends on problem specifics.

D.3.3 Rigorous Derivation of the Approximate Error Reduction

Let $P_e^{(before)} = P_e^*(X)$ and $P_e^{(after)} = P_e^*(X, A)$. From Eq. 2: $H(C|X, A) = (1 - \lambda) \cdot H(C|X)$. Applying the linearity assumption: $P_e^{(before)} \approx \kappa \cdot H(C|X)$ and $P_e^{(after)} \approx \kappa \cdot H(C|X, A)$. Substituting the entropy relation into $P_e^{(after)}$: $P_e^{(after)} \approx \kappa \cdot [(1 - \lambda) \cdot H(C|X)] = (1 - \lambda) \cdot [\kappa \cdot H(C|X)] \approx (1 - \lambda) \cdot P_e^{(before)}$. This derives Eq. 3, hinging on approximate linearity.

D.3.4 Analysis of Approximation Error

Let $P_e^*(Z) = g(H(C|Z))$. Taylor expansion of $g(H)$ around $H = 0$ ($g(0) = 0$): $g(H) = \kappa_1 H + \frac{1}{2}\kappa_2 H^2 + O(H^3)$, where $\kappa_1 = g'(0) \equiv \kappa$, $\kappa_2 = g''(0)$. Let $H_{\text{before}} = H(C|X)$, $H_{\text{after}} =$

$(1-\lambda)H_{\text{before}}$. $P_e^{(\text{before})} \approx \kappa_1 H_{\text{before}} + \frac{1}{2}\kappa_2 H_{\text{before}}^2$ $P_e^{(\text{after})} \approx \kappa_1(1-\lambda)H_{\text{before}} + \frac{1}{2}\kappa_2(1-\lambda)^2 H_{\text{before}}^2$ The ratio is: $\frac{P_e^{(\text{after})}}{P_e^{(\text{before})}} \approx \frac{(1-\lambda)+\frac{\kappa_2}{2\kappa_1}(1-\lambda)^2 H_{\text{before}}}{1+\frac{\kappa_2}{2\kappa_1}H_{\text{before}}}$. Using $(1+x)^{-1} \approx 1-x$: $\frac{P_e^{(\text{after})}}{P_e^{(\text{before})}} \approx (1-\lambda) - \frac{\kappa_2 H_{\text{before}}}{2\kappa_1}\lambda(1-\lambda) + O(H_{\text{before}}^2)$. The approximation error is $\left|\frac{P_e^{(\text{after})}}{P_e^{(\text{before})}} - (1-\lambda)\right| \approx \left|\frac{\kappa_2}{2\kappa_1}\lambda(1-\lambda)H_{\text{before}}\right|$.

Approximation Error Bound. If $H(C|X)$ is small, the relative error of $P_e^{(\text{after})} \approx (1-\lambda)P_e^{(\text{before})}$ is bounded by: $\left|\frac{P_e^{(\text{after})}}{P_e^{(\text{before})}-(1-\lambda)}\right| \leq \gamma_{\text{err}} \cdot H(C|X) + O(H(C|X)^2)$, where $\gamma_{\text{err}} = \left|\frac{\kappa_2}{2\kappa_1}\lambda(1-\lambda)\right|$. Accuracy is highest for small $H(C|X)$, near-linear $g(H)$ (small $|\kappa_2/\kappa_1|$), and λ near 0 or 1.

D.4 Assumptions and Implications

The framework relies on several assumptions. Here we discuss their implications in detail:

1. Conditional Independence of Argument Effects: In systems with strongly correlated evidence sources (arguments not truly independent given the class), the product form might lead to overconfidence (probabilities pushed too close to 0 or 1) or underconfidence if arguments redundantly penalize/reward. Mitigations might involve modeling argument dependencies, but this significantly increases complexity.
2. Accuracy and Objectivity of Quality Assessment: The framework's performance is fundamentally bounded by how well $q(A, c)$ reflects true evidential strength. Biased or inaccurate $q(A, c)$ (e.g., from poorly trained LLM evaluators or flawed heuristic rules) will lead to suboptimal or incorrect probability updates. Designing robust, domain-general $q(A, c)$ is a major challenge. The example in Appendix B is domain-specific.
3. Completeness of Class Set: If novel or rare classes exist but are not in C, the model will be forced to misclassify them into one of the known classes, potentially with high confidence if arguments strongly disfavor other known classes. Open-set recognition capabilities are not inherent.
4. Appropriateness of Parameters: Parameters like α, β and weights w_s, w_m, w_e are global. Optimal values may vary across datasets or even subsets of data within a domain. Universal settings might be suboptimal; domain-specific or adaptive calibration may be needed, adding complexity.
5. Markov Property of Update Process: While simplifying analysis, this means the system has no memory of the history of argumentation beyond the current probability state. Complex argumentation dynamics (e.g., retraction of earlier points based on later ones) are not directly modeled.
6. Approximate Linearity for Lambda-Error Link: The $P_e^{(after)} \approx (1-\lambda)P_e^{(before)}$ relationship (Section 2.2, Appendix D.3) is an approximation. Its accuracy degrades for large information gains (large λ) or high initial error rates where the P_e^* vs. H curve is more non-linear. λ remains a valid measure of relative entropy reduction, but its interpretation as direct error multiplier becomes less precise.
7. Calibration for Optimal Gamma: The optimality of $\gamma^* \approx I(\hat{Y}; Y)/H(\hat{Y})$ for the adaptive prior (Appendix D.2) assumes ensemble predictions P_{ens} are reasonably well-calibrated proxies for P_{true}. Significant miscalibration of P_{ens} could lead to a suboptimal γ^*, inappropriately weighting the ensemble vs. uniform prior.
8. Need for Domain-Specific Quality Metrics: The core framework is general, but its practical effectiveness heavily relies on defining meaningful and computable argument quality metrics $q(A, c)$ specific to the application domain and the nature of the structured knowledge (e.g., LLM text, database facts, expert rules). The biological example (Appendix B) is just one complex instantiation; simpler or different metrics would be needed elsewhere.

Understanding these assumptions and their implications is important for applying LLM-BMC appropriately and for guiding future research to address these limitations.

E CONVERGENCE ANALYSIS

E.1 PROBLEM FORMULATION

The LLM-BMC framework defines an iterative update process for probability distributions $\{P^{(t)}(c|X)\}_{t=0}^{\infty}$:

$$P_i^{(t+1)}(c|X) = \frac{P_i^{(t)}(c|X) \cdot \prod_{j \neq i} f(A_j^{(t)}, c)}{\sum_{c' \in C} \left(P_i^{(t)}(c'|X) \cdot \prod_{j \neq i} f(A_j^{(t)}, c') \right)}$$

Convergence analysis addresses: existence, uniqueness, and rate of convergence to a limit $P^*(c|X)$.

E.2 EXISTENCE OF FIXED POINTS

E.2.1 MATHEMATICAL PRELIMINARIES

The update is an operator $\mathcal{T}(P)(c) = \frac{P(c) \cdot F(c)}{\sum_{c' \in C} P(c') \cdot F(c')}$, where $F(c) = \prod_j f(A_j, c)$. $\mathcal{T}$ maps the probability simplex Δ^{K-1} to itself.

E.2.2 EXISTENCE THEOREM

Theorem E.1 (Existence of Fixed Points)**.** *If $0 < f_{min} \leq f(A_j, c) \leq f_{max} < \infty$, there exists at least one fixed point $P^* \in \Delta^{K-1}$ such that $\mathcal{T}(P^*) = P^*$.*

Proof. Δ^{K-1} is compact and convex. $\mathcal{T}$ is continuous. By Brouwer's fixed-point theorem, P^* exists. □

E.2.3 BOUNDARY BEHAVIOR

Lemma E.2. *If $f(A_j, c) > 0$ and $P^{(0)}(c|X) > 0$ for all c, then $P^{(t)}(c|X) > 0$ for all t, c.*

Proof. By induction. If $P^{(t)}(c|X) > 0$ and $f(A_j, c) > 0$, then $P^{(t)}(c|X) \cdot \prod_j f(A_j, c) > 0$. Normalization preserves positivity. □

This ensures updates remain in the interior of the simplex if started there.

E.3 UNIQUENESS ANALYSIS

E.3.1 SUFFICIENT CONDITIONS FOR UNIQUENESS

Theorem E.3 (Sufficient Condition for Uniqueness via Contraction)**.** *If $\mathcal{T}$ is a contraction mapping on Δ^{K-1}, the fixed point is unique.*

Theorem E.4 (Sufficient Condition for Uniqueness via Bounded Ratios)**.** *If for all class pairs (c_1, c_2) and argument sets A, $1/M \leq \frac{\prod_j f(A_j, c_1)}{\prod_j f(A_j, c_2)} \leq M$ for some finite $M > 0$, the fixed point is unique.*

Proof Sketch. Bounded ratios prevent any class from dominating or being fully suppressed, ensuring distinct fixed points would be drawn together. □

E.3.2 MULTIPLE FIXED POINTS SCENARIO

Multiple fixed points may occur if conditions for Theorem E.4 are not met (e.g., highly polarized arguments). Mitigation: balanced quality assessment, consistent argument presentation, stable priors.

E.4 Conditions for Convergence

Condition E.1 (Bounded Modification)**.** $1/(1+\beta) \leq f(A_j, c) \leq 1+\alpha$ *for* $\alpha, \beta > 0$*. (Satisfied by Eq. 4).*

Condition E.2 (Finite Arguments)**.** *The set of distinct arguments A is finite, or iterations are bounded.*

Condition E.3 (Monotonic Information)**.** *Each argument provides non-negative information gain:* $D_{KL}(P_{true}||P^{(t+1)}) \leq D_{KL}(P_{true}||P^{(t)})$.

E.4.1 Convergence Theorem

Theorem E.5 (Guaranteed Convergence)**.** *If Conditions E.1, E.2, and (at least one of Condition E.3 or Theorem E.4's condition) hold,* $\{P^{(t)}(c|X)\}$ *converges to* $P^*(c|X)$.

Proof Sketch. Bounded modification limits step size. Finite arguments ensure stabilization or cycling. Monotonic information or bounded ratios prevent cycling, forcing convergence. □

E.5 Rate of Convergence Analysis

E.5.1 Convergence Rate Measure

Rate measured by $d(t) = d(P^{(t+1)}, P^{(t)})$. Often $d(t) \approx r^t \cdot d(0)$, $r \in (0, 1)$.

E.5.2 Factors Affecting Convergence Rate

Argument quality, α, β values, argument consensus, prior distribution.

E.5.3 Quantitative Bounds

Theorem E.6 (Convergence Rate Bound)**.** $r \leq 1 - \min_{c \in C} P^{(0)}(c|X) \cdot \frac{1}{(1+\alpha)(1+\beta)}$.

Proof Sketch. Derived from max possible change in probability per step. □

E.5.4 Empirical Convergence Behavior

Typical patterns: Rapid ($r < 0.5$, 2-3 iterations), Moderate ($0.5 \leq r < 0.8$, 4-7 iterations), Slow ($r \geq 0.8$, 10+ iterations).

Table 5: Empirical convergence rates across different single-cell classification datasets (illustrative). "Class" here refers to cell types.

Dataset (Tissue)	Avg. Argument Quality	Avg. Convergence Rate (r)	Iterations to Convergence
PBMC	0.72	0.63	5
Brain	0.65	0.71	7
Pancreas	0.69	0.67	6
Lung	0.74	0.61	5
Liver	0.68	0.69	6
Kidney	0.71	0.64	5

E.6 Special Cases and Practical Considerations

E.6.1 Single Iteration Case

Convergence guaranteed; focus on argument quality and modifier.

E.6.2 TERMINATION CRITERIA

Max change threshold (e.g., $\max_c |P^{(t+1)}(c|X) - P^{(t)}(c|X)| < 10^{-4}$), max iterations (e.g., 10-15), oscillation detection.

E.6.3 HANDLING PATHOLOGICAL CASES

Contradictory arguments (dampen), zero probability trapping (enforce min probability if $P^{(0)}$ can be zero, though Lemma E.2 usually prevents this if $P^{(0)} > 0$), quality assessment failures (validate $q(A, c)$).

E.7 SUMMARY

LLM-BMC converges under reasonable conditions. Rate depends on argument quality/consistency. Fixed point(s) represent principled integration of data-driven predictions and structured knowledge.

F ADDITIONAL THEORETICAL PROPERTIES

This section details proofs for additional theoretical properties.

F.1 PROOF OF OPTIMAL MIXTURE WEIGHT

The proof for Theorem D.1 (relating γ^* to $I(\hat{Y}; Y)/H(\hat{Y})$) is provided in Appendix D.2 alongside other prior selection derivations.

F.2 ADVANTAGE OVER VOTING/AVERAGING

Theorem C.1 (Advantage over Averaging Under Heterogeneous Expertise): If (1) model expertise $e_i(c)$ is heterogeneous, (2) argument quality $q(A_j, c)$ correlates with $e_j(c)$, and (3) arguments are non-contradictory, LLM-BMC achieves lower expected error than simple averaging.

Proof. Error for averaging: $P_{err}^{avg} = 1 - \mathbb{E}\left[\frac{1}{N}\sum P_i(c^*|X)\right]$. LLM-BMC effectively uses weights $w_i(c^*) \propto e_i(c^*)$ derived from argument quality. Error for LLM-BMC: $P_{err}^{LLM-BMC} = 1 - \mathbb{E}\left[\frac{\sum w_i(c^*)P_i(c^*|X)}{\sum w_i(c^*)}\right]$. By Jensen's inequality, if weights reflect expertise, $P_{err}^{LLM-BMC} < P_{err}^{avg}$. □

Theorem C.2 (Advantage under Knowledge-Resolvable Ambiguity): If domain knowledge can identify patterns base models miss, LLM-BMC strictly outperforms weighted combinations of base models.

Proof. Consider items from clusters A or B (probabilities p_A, p_B) belonging to the same true class, but base models assign them to different classes. Best combined model error $\geq \min(p_A, p_B)$. If domain knowledge arguments (quality $q > 0$) correctly group A and B, LLM-BMC error can be $\leq (1 - \lambda' q)\min(p_A, p_B)$ (where λ' relates to λ and modifier strength), which is lower. □

F.3 ROBUSTNESS TO NOISY ARGUMENTS

Theorem C.3 (Robustness Bound): If ϵ is fraction of misleading arguments (max quality q_{max}), max error increase $\Delta P_{err} \leq \epsilon \cdot \alpha \cdot q_{max} \cdot P_{err}^{base}$.

Proof. A misleading argument (quality q) for incorrect class c' increases $P(c')$ by $\approx (1 + \alpha q)$, decreases true $P(c^*)$ by $\approx (1 - \beta q)$. Cumulative effect of ϵN bad args on ratio $P(c')/P(c^*)$ leads to error increase bounded as stated. □

Corollary C.3.1 (Quality Assessment Importance): If quality assessment detects misleading args (assigning quality $q_{low} \ll q_{max}$) with accuracy δ, bound improves: $\Delta P_{err} \leq (\epsilon(1 - \delta)\alpha q_{max} + \epsilon\delta\alpha q_{low})P_{err}^{base}$.

F.4 Additional Theoretical Guarantees

Optimal Modifier Parameters (Theorem C.4): Under min cross-entropy, $\alpha_{opt} \approx \frac{1-P_{err}}{P_{err}} \ln(\frac{1-P_{err}}{P_{err}})$, $\beta_{opt} \approx \frac{P_{err}}{1-P_{err}} \ln(\frac{1-P_{err}}{P_{err}})$.

Refined Convergence Rate (Theorem C.5): If args for true class have $q(A, c) \geq q_{min} > 0$, rate $r \leq 1 - q_{min} \cdot \alpha \cdot \min_c P^{(0)}(c|X)$.

F.5 Application Guidelines

1. Prior Selection: Use ensemble prior if $I(\hat{Y}; Y)/H(\hat{Y}) > 0.7$; else, adaptive mixture.
2. Quality Assessment: Focus on marker specificity and pathway coherence for high feature overlap.
3. Parameter Tuning: Start with theoretical α, β; fine-tune empirically.
4. Convergence Monitoring: For low-quality args ($q < 0.4$), consider early stopping (3-4 iterations).
5. Model Diversity: Ensure base models have complementary expertise.

These translate theory into practice for LLM-BMC implementation.

G Method Comparison

Table 6: Comparison of LLM-BMC with alternative approaches to knowledge integration

Approach	Math. Formal.	Reasoning Integr.	Dynamic Updating	Uncertainty Quantif.	Key Innovation
Feature Engineering	✓	×	×	Partial	Domain-specific attributes
Rule-Based Systems	Partial	✓	×	×	Explicit knowledge rules
Prompt Engineering	×	✓	×	×	Natural language instructions
Neuro-symbolic AI	✓	✓	Partial	Partial	Symbolic-neural integration
Chain-of-Thought	×	✓	×	×	Step-by-step reasoning
Bayesian Model Avg.	✓	×	✓	✓	Posterior weighting
LLM-BMC (Ours)	✓	✓	✓	✓	λ parameter, quality q

This table highlights the distinguishing features of LLM-BMC compared to alternative approaches to knowledge integration in classification tasks.

H LLM Usage Details

This section provides detailed guidelines on the practical usage of Large Language Models within the LLM-BMC framework, covering both the argument generation and quality assessment phases as described in Section 2.4 and Algorithm 2. Our experiments utilized GPT-4o via the OpenAI API.

Role 1: Argument Generation The primary role of the LLM is to generate a structured, domain-specific argument A that links the input features X to a candidate class c. This process translates numerical data into a format suitable for knowledge-based reasoning.

Prompting Strategy. A carefully designed prompt is important for generating high-quality, relevant arguments. The prompt should instruct the LLM to act as a domain expert (e.g., a cell biologist) and to structure its response to align with the components of our quality function $q(A, c)$. For our single-cell classification case study, the prompt template instructs the LLM to provide evidence-based arguments including marker gene evidence, mechanistic coherence, and optional external support from established knowledge bases.

Role 2: Assisting Argument Quality Assessment The LLM assists in quantifying argument quality $q(A, c)$ primarily through structured information extraction, which provides the inputs for the mathematical formulas detailed in Appendix B. This indirect approach ensures greater objectivity and reproducibility compared to direct LLM-based scoring.

INFORMATION EXTRACTION. After an argument A is generated, we use subsequent LLM calls to parse it into a structured format. This includes extracting mentioned gene symbols for the specificity component $s(A, c)$ and identifying biological pathways for the mechanistic coherence component $m(A, c)$. The extracted entities are then cross-referenced with external databases (e.g., Cell Marker DB, KEGG) to compute the final numerical values for specificity, relevance, and coherence as defined in our framework.

Implementation and Practical Considerations We used GPT-4o for its strong reasoning capabilities and extensive knowledge in specialized domains like biology. To ensure reproducibility, we set the API temperature parameter to 0.1 to minimize randomness in the generated arguments and extracted information. Our implementation includes validation checks and retry mechanisms to handle malformed outputs, ensuring the robustness of the data processing pipeline. For large-scale applications, strategies like prompt optimization, result caching for similar inputs, or using smaller, fine-tuned models could be explored to manage costs and latency.

Statement on LLM Usage in Paper Writing In accordance with MLGenX 2026 guidelines, we disclose that LLMs were used in a limited capacity during the preparation of this manuscript. Specifically, GPT-4o was used for grammar checking and language polishing of certain sections. All scientific content, experimental design, theoretical contributions, and interpretations are the original work of the author. The LLMs did not contribute to research ideation, experimental execution, or data analysis beyond their role as a component within the proposed LLM-BMC framework itself.