

Unifying Conformal Language Tasks with In-Context Ensembles

Xiao Shi Huang¹ Chen-Yuan Lin¹ Bruce Kuwahara¹
Kin Kwan Leung² Jesse C. Cresswell²

¹Signal 1 AI ²Layer 6 AI

{gary, thomas, bruce}@signal1.ai {kk, jesse}@layer6.ai

Abstract

Many NLP tasks, such as summarization and extractive question answering, reduce to retrieving relevant content from documents under two constraints: *coverage*, retaining enough pertinent information to achieve some goal, and *conciseness*, removing as much irrelevant information as possible. Conformal prediction methods have been used to guarantee coverage, and must be optimized for conciseness through design of a score function. State-of-the-art scoring functions use hand-engineered LLM prompts asking the model to rate the importance of content, but manual prompt engineering is labor-intensive and task-specific. We introduce the *Conformal Relevance* framework which uses in-context learning example curation and ensembling to create a score function which maintains coverage while improving conciseness with minimal manual input. We demonstrate this framework’s application on seven NLP tasks, and also theoretically study the impact of diversity for ensembled conformal scores, giving a complementarity condition that characterizes when ensembling improves worst-case sentence scores, and a saturation bound on ensemble improvement.

1 Introduction

Many NLP tasks, including summarization (Mukherjee et al., 2022), extractive question-answering (QA) (Yang et al., 2018), legal review (Koreeda and Manning, 2021), and clinical evidence selection (DeYoung et al., 2020), reduce to retrieving relevant content from documents. Systems for these tasks must satisfy two demands: *coverage* (relevant content is retained) and *conciseness* (irrelevant content is excluded). Conformal prediction (Vovk et al., 2005) has become popular as a general purpose framework for providing distribution-free, finite sample coverage guarantees. It operates by calibrating an arbitrary scoring function over a labeled dataset. While coverage is

guaranteed, conciseness strongly depends on the scoring function’s predictive power.

Several examples of this framework have been studied in detail, differing in their relevance criterion. For question-answering, Mohri and Hashimoto (2024) define relevant content as non-hallucinated claims in the answer, while hallucinations should be filtered out. Kuwahara et al. (2025) look at extractive summarization, defining relevant content as important sentences, and filter out unimportant sentences for conciseness. For these NLP tasks and others, the scoring function is typically a large language model (LLM) configured through task-specific prompt engineering, which is labor-intensive, brittle, and not scalable (Lu et al., 2022; Min et al., 2022).

We aim to subsume all such content-selection tasks under a general purpose relevance scoring function that replaces manual prompt-writing with in-context learning (ICL) (Brown et al., 2020; Dong et al., 2023) example curation. Rather than *describing* a relevance criterion in natural language as done in past work, we *demonstrate* it through curated examples and let the LLM infer the criterion from the demonstrations. We call this framework *Conformal Relevance*: recall-oriented conformal calibration paired with a relevance scoring function, instantiated through an ICL-driven LLM score.

To address the diversity of relevance criteria in a unified way, we develop multiple ICL example selection strategies that induce LLM scoring functions with systematically different strengths and failure modes. Ensembling these scores improves performance while remaining entirely task-agnostic. Concretely, we average K mechanistically distinct ICL-based scoring functions—chosen to span different signal types so that their failures are unlikely to coincide—and treat the result as a single conformal scoring function (Ochoa Rivera et al., 2025; Waldron, 2026). We mathematically formalize a *complementarity condition* that char-

acterizes when ensembling improves worst-case conformal scores. The marginal gain from each additional sub-scoring function gives diminishing returns, bounded by $O(1/K)$.

We apply a fixed ICL scoring configuration on seven NLP datasets spanning five domains and show performance improvements over manually crafted prompts that define the relevance criterion per task. Our Conformal Relevance method removes substantially more irrelevant content at the same coverage on all seven tasks, with up to a $\sim 50\%$ reduction in retained length. Through several ablations and controls, we determine that the gain stems from retrieval-induced diversity across the ensemble’s in-context demonstrations. The required label budget is modest: 150–440 labels per task, comprising a shared ICL pool plus a 100-sample calibration set. Our contributions are:

1. A universal ICL ensemble scoring function for disparate content selection tasks with conformal coverage guarantees.
2. Four mechanistically diverse sub-scoring functions spanning distinct signal types.
3. A mathematical formalization of score ensembling for recall-oriented conformal prediction.

2 Background & Related Work

Content selection. Content selection is a generic task in NLP where relevant information must be extracted from a document. It occurs under many instantiations depending on the definition of relevance. The prototype is extractive summarization where relevance means a content span of the document is important (Mukherjee et al., 2022). Extractive QA requires a model to select content from a corpus that is relevant to a question (retrieval) before using it to generate an answer (Yang et al., 2018), for example as done in Retrieval-Augmented Generation (RAG) (Lewis et al., 2020). PII detection (Pilán et al., 2022; Shen et al., 2026; Ponomarenko et al., 2026) (similar to named entity recognition (Wang et al., 2025)) selects content that constitutes private information so that it can be appropriately masked. These examples are far from comprehensive as content selection appears in a vast array of forms. Our aim is to unify content selection tasks throughout language modeling under a single framework for providing conformal guarantees on the capture of relevant content.

Conformal prediction. Given inputs $x \in \mathcal{X}$ and ground-truth values $y^* \in \mathcal{Y}$ drawn jointly from a distribution $(x, y^*) \sim \mathbb{P}$, split conformal predic-

tion (Vovk et al., 2005; Shafer and Vovk, 2008) constructs a prediction set $C_{\hat{q}}(x_{\text{test}}) \subseteq \mathcal{Y}$ for a new datapoint x_{test} that come with a finite-sample coverage guarantee

$$\mathbb{P}[y_{\text{test}}^* \in C_{\hat{q}}(x_{\text{test}})] \geq 1 - \alpha, \quad (1)$$

where the error rate α is user-defined. Conformal prediction requires an arbitrary score function $S(x, y)$, which is often derived from a black-box machine learning model. By computing scores on a labeled calibration dataset, the conformal threshold $\hat{q}$ is set as the $\lceil (1 - \alpha)(n + 1) \rceil / n$ quantile of the n scores. Then, prediction sets are generated as

$$C_{\hat{q}}(x_{\text{test}}) = \{y \in \mathcal{Y} \mid S(x_{\text{test}}, y) < \hat{q}\}. \quad (2)$$

Smaller sets are preferred, *ceteris paribus*. Conformal prediction is extremely useful because it makes no assumptions about the nature of scores, only that x_{test} is exchangeable with the calibration data, a mild assumption that holds for IID settings (Angelopoulos and Bates, 2023).

Conformal methods for content selection.

Conformal factuality (Mohri and Hashimoto, 2024) studies a QA content selection setting where claims in an existing answer are filtered out until no hallucinations remain with high confidence. This is a *precision*-style coverage guarantee where selected content must all fit the relevance criteria (here being non-hallucinated). Conformal factuality was further studied in RAG settings (Feng et al., 2025; Chakraborty et al., 2026). Extending from the QA setting, conformal importance (Kuwahara et al., 2025) gives a *recall*-style coverage guarantee for extractive summarization where the prediction set must select all relevant content with high confidence. Since recall-style coverage is more widely applicable to NLP tasks, we adopt it and recount conformal importance in more detail below.

Given a document x with content spans $c_1, \dots, c_m$ and a ground-truth important subset $y^* \subseteq \{c_1, \dots, c_m\}$, a relevance function $R(c; x) \in [0, 1]$ assigns an importance level to each candidate content span c_i . The conformal score S_β of data point (x, y^*) is set as the lowest relevance level where a fraction β of ground-truth important content spans are retained, rounded up to the nearest whole number. If we sort the scores R for each $c \in y^*$ so that $R_{(1)} \leq \dots \leq R_{(|y^*|)}$, then

$$S_\beta(x, y^*) = R_{(r)}, \quad r = |y^*| - \lceil \beta |y^*| \rceil + 1. \quad (3)$$

For the case $\beta = 1$ of perfect recall, we write

$$S(x, y^*) = \min_{c \in y^*} R(c; x). \quad (4)$$

A threshold $\hat{q}$ is calibrated on the conformal scores of a calibration dataset. For a new document x_{test} , any content span c_i with relevance $R(c_i; x_{\text{test}}) < \hat{q}$ is filtered out. The remaining content spans form the prediction set $y_{\text{test}} \subseteq x$. This method provides a coverage guarantee that at least a fraction β of important content per datapoint is retained with high probability (Kuwahara et al., 2025),

$$\mathbb{P}\left[\frac{|y_{\text{test}} \cap y_{\text{test}}^*|}{|y_{\text{test}}^*|} \geq \beta\right] \geq 1 - \alpha. \quad (5)$$

Coverage holds for *any* relevance function R under exchangeability of the documents x , but the quality of R —its power in classifying important content spans—determines *conciseness* for the retained set.

Similar recall-style guarantees have been developed for conformal agent error attribution (Feng et al., 2026). In this task the relevance function predicts how likely a step in the agent’s trace is to be a decisive error, and coverage ensures that decisive errors are contained in the prediction set.

Conformal factuality, conformal importance, and conformal agent error attribution all assume a fixed S deriving from R , which is typically an LLM with a manually crafted prompt designed specifically for one definition of relevance. Generative tasks in this class have been studied generally by Loaiza-Ganem et al. (2026). In this work, we construct a scoring function that generalizes across content selection tasks without per-domain engineering.

Conformal ensembles. When multiple scoring functions are available, they can be ensembled to produce stronger coverage guarantees, or more concise sets. Conformal score aggregation (Ochoa Rivera et al., 2025) establishes that score-level aggregation is strictly more efficient than set-level, yielding tighter prediction regions in classification and regression. Gasparin and Ramdas (2024b) prove a $1 - 2\alpha$ coverage floor for majority-vote ensembling of prediction sets. Other score or e -value combination methods (Luo and Zhou, 2025; Alami et al., 2026) address classification and regression, not language tasks. In the language space, and building on conformal factuality, Cherian et al. (2024) propose a boosted linear combination of four heterogeneous scoring functions with level-adaptive conformal risk control (Angelopoulos et al., 2024). Our work characterizes when ensembling provides a conciseness advantage in recall-oriented conformal language tasks. We discuss additional related work on conformal aggregation in Appendix A.

ICL example selection. A substantial literature studies which in-context examples to present to an LLM, spanning similarity-based retrieval (Liu et al., 2022a) and learned retrievers (Rubin et al., 2022). Ordering and input–label mapping substantially affect downstream performance (Lu et al., 2022; Min et al., 2022), and diverse, balanced demonstrations improve compositional generalization (Levy et al., 2023). Determinantal point process (DPP) based selection over in-context examples (Ye et al., 2023; Sui et al., 2024) is closest in spirit to our work, but instead of generation we consider relevance scoring within conformal frameworks.

3 Theoretical Framework

Considering recall-style coverage guarantees for content selection tasks (Eq. 5), we replace a single relevance function R with a K -strategy ICL ensemble. The remainder of this section asks three questions in turn: *does ensembling preserve the $1 - \alpha$ coverage guarantee?*; *under which conditions does ensembling with $K = 2$ improve conciseness?*; and *when does adding an additional R_{K+1} raise the ensembled conformal score?*

In this section we consider $\beta = 1$ for clarity before mentioning results for general β . See Appendix C for extended details.

3.1 Problem Setup

From Eq. 4, the conformal score $S(x, y^*)$ for a given relevance function $R(c; x)$ is defined as the *floor*—the lowest relevance of any positive $c \in y^*$ —which we write $S(R)$. Raising the floor will in general also raise the calibrated threshold $\hat{q}$, which shrinks prediction sets, assuming relevance scores for $c \notin y^*$ are unchanged. As long as coverage is unaffected, raising the floor under these conditions is desirable. We aim to show that ensembling different relevance scores can raise the floor while maintaining coverage. For K distinct relevance functions $R_1, \dots, R_K$, the mean ensemble $\bar{R}^{(K)}(c; x) \equiv \frac{1}{K} \sum_{j=1}^K R_j(c; x)$ has ensemble floor $S^{(K)} \equiv \min_{c \in y^*} \bar{R}^{(K)}(c; x)$.

3.2 Validity via ensembling

We ask: *does ensembling preserve the $1 - \alpha$ coverage guarantee?* Set-level ensembling degrades coverage to $1 - 2\alpha$ (Gasparin and Ramdas, 2024b). Instead, we adopt score-level ensembling by taking the mean of several R_j , ensuring that they output relevance on a $[0, 1]$ scale. Since $\bar{R}^{(K)}$ gives a

Table 1: Notation

x, c, y^*	document, content span, relevant set	S_β	conformal score (Eq. 3 for $\beta = 1$)
$R_j, \bar{R}^{(K)}$	relevance fn, mean ensemble	$\hat{q}$	calibrated threshold
K, k	# relevance fns, ICL examples per fn	$\Pi, \mathcal{C}$	ICL example pool, calibration set
α, β	miscoverage, recall target	n	calibration set size
$S, S^{(K)}$	individual / ensemble floor at $\beta = 1$	Comp	complementarity (Def. 1)
$S_{\max}, S_{\min}$	max/min individual floor at $K=2$	$\delta^K(c)$	floor margin (Prop. 1)

fixed conformal score function via Equation (4), standard split-conformal theory yields $1 - \alpha$ coverage without degradation as long as exchangeability holds. We assume access to a pool of labeled examples Π from which ICL examples are drawn for the R_j , disjoint from calibration and test sets. For per-sample ICL retrieval, conditioning on $\Pi = \pi$ restores exchangeability, and the law of total expectation lifts the conditional guarantee to an unconditional one (App. B). Hence, our mean ensembling will preserve $1 - \alpha$ coverage.

3.3 Complementarity at $K = 2$

We ask: under which conditions does ensembling with $K = 2$ improve conciseness? Whether the ensemble floor $S^{(2)}$ improves over the individual floors depends on their values, and on how aligned the scorers' failures are; we capture these with $S_{\max}, S_{\min}$ and a new quantity Comp, for *complementarity*, then state the decomposition that combines them.

Concretely, let $S_{\max} \equiv \max(S(R_1), S(R_2))$ and $S_{\min} \equiv \min(S(R_1), S(R_2))$ denote the better and worse individual floors; their difference $S_{\max} - S_{\min} \geq 0$ is the *floor gap*.

Definition 1 (Complementarity). *For two relevance functions R_1, R_2 on input x ,*

$$\text{Comp}(R_1, R_2) \equiv \min_{c \in y^*} [R_1(c) + R_2(c)] - [S(R_1) + S(R_2)] \geq 0. \quad (6)$$

Interpretation. Comp measures the extent to which the two relevance functions disagree on the lowest scoring positives $c \in y^*$. Comp = 0 when R_1 and R_2 share a lowest scoring positive; Comp > 0 (complementarity) when their minimum is achieved on different positives, except possibly when R_j has degeneracy across the c_i .

Lemma 1. (*Ensemble floor advantage*) *At $K = 2$ and $\beta = 1$,*

$$S^{(2)} = \frac{1}{2}(S_{\max} + S_{\min} + \text{Comp}), \quad (7)$$

and $S^{(2)} > S_{\max}$ iff $\text{Comp} > S_{\max} - S_{\min}$.

See proof in App. C.2.

Interpretation. The lemma tells us when a mean ensemble of two scorers has an advantage over both individual components by raising the floor: Comp must exceed the floor gap. Designing good ensembles thus means searching for scorers whose lowest scoring positives do not align. See App. C.2 for a content span-level example showing that the complementarity condition $\text{Comp} > S_{\max} - S_{\min}$ can be achieved in practice.

3.4 Diminishing returns

We ask: when does adding an additional R_{K+1} raise the ensemble floor $S^{(K)}$? We study $\delta^K(c) \equiv \bar{R}^{(K)}(c; x) - S^{(K)}$, the margin between c 's average relevance score and the ensemble floor, which is non-negative for every $c \in y^*$.

Proposition 1. ($O(\frac{1}{K})$ diminishing returns)

We have $S^{(K+1)} > S^{(K)}$ iff $R_{K+1}(c) > S^{(K)} - K \delta^K(c)$ for every $c \in y^*$. When the condition holds,

$$0 < S^{(K+1)} - S^{(K)} \leq \frac{1 - S^{(K)}}{K + 1} = O\left(\frac{1}{K}\right). \quad (8)$$

See proof in App. C.3.

Interpretation. Adding additional scoring functions to the ensemble can help when the new function R_{K+1} assigns higher relevance to the current lowest scoring spans, i.e. those with $\delta^K(c) = 0$. This again emphasizes the need for diversity among the relevance functions. Even when the condition for improvement holds, each additional scorer contributes proportionally less, and the marginal gain at step $K + 1$ is at most $(1 - S^{(K)})/(K + 1)$. For small K , this can still be a meaningful improvement, but large K is not necessary in practice.

3.5 Extension to $\beta < 1$

The theory developed above works in the setting of $\beta = 1$, perfect recall of relevant content spans $c \in y^*$. Requiring perfect recall may be overly conservative, and result in larger than desirable prediction sets for a given coverage (Equation (5)). Here we briefly summarize the extension of our theory to $\beta < 1$, with full results in Appendix C.4.

Algorithm 1 Conformal calibration and prediction for content selection with a K -scorer ensemble.

```

1: Inputs: ICL pool  $\Pi$ , calibration set  $\mathcal{C}$ ,  $K$  ICL selection strategies  $\sigma_j$ , ICL count  $k$ , miscoverage  $\alpha$ , recall target  $\beta$ 
   Offline calibration on  $\mathcal{C}$ 
2: for each  $x \in \mathcal{C}$  do
3:   for all  $j \in \{1, \dots, K\}$  do
4:      $\text{ICL}_j \leftarrow \sigma_j(\Pi, x)$   $\triangleright$  draw  $k$  examples
5:      $R_j(\cdot; x) \leftarrow \text{LLM}(\cdot; x, \text{ICL}_j)$   $\triangleright$  define scorers
6:   end for
7:    $\bar{R}(\cdot; x) \leftarrow \frac{1}{K} \sum_{j=1}^K R_j(\cdot; x)$   $\triangleright$  mean ensemble
8:   compute  $S_\beta(x)$  using  $\bar{R}(\cdot; x)$  (Equation (3))
9: end for
10:  $\hat{q} \leftarrow [\alpha(n+1)]$ -th order statistic of
     $\{S_\beta(x) : x \in \mathcal{C}\}$  (recall-style conformal quantile)
   Online prediction on test document  $x$ 
11: Repeat steps 3–7 on a test document  $x$  to obtain  $\bar{R}(\cdot; x)$ 
12: Output:  $\{c \in x : \bar{R}(c; x) \geq \hat{q}\}$ 

```

To mirror the definition in Equation (3), we replace the minimum in $\text{Comp}(R_1, R_2)$ by the r th order statistic (Equation (20)), which maintains the interpretation of measuring complementarity—how much two scoring functions disagree on the r th lowest scoring positive span. Then the score $S_\beta^{(2)}$, the r th lowest score from the mean ensemble $\bar{R}$, improves over $S_{\beta, \max}$ exactly when $\text{Comp}_\beta > S_{\beta, \max} - S_{\beta, \min}$ (Lemma 2). In other words, the floor is raised when at least $\lceil \beta |y^*| \rceil$ of the positive spans satisfy $\bar{R}(c) > S_{\beta, \max}$.

For general K , adding an additional scoring function to an existing ensemble will give $S_\beta^{(K+1)} > S_\beta^{(K)}$ when at least $\lceil \beta |y^*| \rceil$ of the positive spans satisfy $R_{K+1}(c) > S_\beta^{(K)} - K\delta_\beta^K(c)$, where now the margin is $\delta_\beta^K(c) = \bar{R}^{(K)}(c; x) - S_\beta^{(K)}$. In this case, the improvement is bounded above by $(1 - S_\beta^{(K)})/(K + 1)$ which again shows $O(\frac{1}{K})$ diminishing returns (Proposition 2).

4 Method - Conformal Relevance

Guided by the theoretical framework of Section 3, we compose an ensemble of mechanistically diverse ICL strategies into a mean scoring function $\bar{R}$, with the end-to-end procedure summarized in Algorithm 1. This section defines the data split (Section 4.1), the ICL-based relevance functions (Section 4.2), and the ICL-selection strategies (Section 4.3). The design and hyperparameters for our reference implementation are held fixed across all main results for every dataset and task, with key ablations in Section 5.3.

4.1 Data Split

The datasets we experiment on are shown in Table 2. Each dataset is partitioned into three sub-

sets: a **pool** Π supplies ICL examples, a **calibration set** $\mathcal{C}$ is used to set the conformal threshold $\hat{q}$, and a held-out **test set** is used for all reported results. We use $|\Pi|=50$ for single-intent datasets, or 20 examples per intent for multi-intent ones, $|\mathcal{C}| = n = 100$, and assign the remainder to test. Calibration and test are drawn uniformly at random from the available data so that the exchangeability assumption holds; Π is stratified by intent for the multi-intent datasets (PUMA, SubSumE, ContractNLI, Evidence Inference), so each of the k ICL examples per strategy are drawn from the same-intent subpool of Π as the test sample. The labeling budget ranges from 150 to 440 at maximum (ContractNLI with 17 intents).

4.2 ICL-Based Relevance Scoring

A relevance scoring function $R(c; x) \in [0, 1]$ assigns a score to every defined content span c of document x (e.g. each sentence). We instantiate R using LLMs, but not with a manually crafted prompt comprising the definition of what should be considered relevant for a given task (e.g. describing what constitutes important information to summarize), but only with k ICL examples sampled from Π , each a labeled document with its content spans and binary relevance labels. Avoiding manual prompt engineering means the same scoring function can be applied across content selection tasks, from extractive QA to summarization to PII detection. Details on the construction of the ICL prompts, such as prompt format, windowed compression, optional task hint, and the output schema are elaborated in detail in Appendix E.

4.3 ICL Example Selection Strategies

Building on the potential benefits of scoring ensembles described in Section 3, our aim is to design diverse relevance scoring functions R_j that can be combined via mean ensembling as $\bar{R}^{(K)}$, which then goes into the conformal score $S_\beta^{(K)}$ for content selection (Equation (3)). To realize the benefits in Lemma 1, we need relevance functions with mechanistically uncorrelated errors (low scores assigned to relevant spans $c \in y^*$). However, there are diminishing returns when adding many scorers to the ensemble (Proposition 1), and added costs. We therefore strike a balance by constructing $K = 4$ ICL-selection strategies σ_j spanning distinct retrieval signals. Each strategy determines how ICL examples are selected from the pool given an input x , which then go into a fixed prompt format.

Table 2: Dataset statistics. Full processing details in Appendix D.

Dataset	Domain	Task	Avg # Spans	Pos. Rate	Intents	# ICL Pool	# Cal	# Test
ECTSum (Mukherjee et al., 2022)	Financial	Summarization	46	9.8%	—	50	100	2,275
SubSumE (Yadav et al., 2021)	Encyclopedic	Query-focused summ.	461	2.5%	10	200	100	1,900
PUMA (Naik et al., 2024)	Medical	Perspective-based QA	14	36.0%	5	100	100	6,082
PhysioNet (Douglass et al., 2007)	Medical	PII detection	23	13.9%	—	50	100	637
HotpotQA (Yang et al., 2018)	General	Multi-hop QA	41	6.3%	—	50	100	10,000
Evidence Inf. (DeYoung et al., 2020)	Medical	Evidence extraction	155	2.0%	8	160	100	9,036
ContractNLI (Koreeda and Manning, 2021)	Legal	NDA clause scoring	84	2.7%	17	340	100	5,733

Hence, each strategy σ_j gives rise to a distinct scoring function R_j .

- `anchor_dpp` embeds the query document x , finds the document from the ICL pool with highest cosine similarity, and sets that as the anchor. Then $k - 1$ other documents are selected from the pool via a DPP conditioned on the anchor.
 - `pattern_dpp` embeds each content span in a pool document, and computes centroid embeddings of the positive ($c \in y^*$) and negative ($c \notin y^*$) spans. The difference of positive and negative centroids forms a relevance direction vector. We then select k examples via a DPP over the relevance directions.
 - `bm25` is lexical top- k retrieval with BM25 (Robertson and Zaragoza, 2009), using x as the query into the index over ICL pool documents;
 - `random` selects k documents from Π uniformly at random, and serves as an ensemble regularizer.
- Extended descriptions of the ICL selection strategies are in Appendix F.

Because every R_j outputs on the same $[0, 1]$ scale, the natural ensembling method is the element-wise mean $\bar{R}(c; x) = \frac{1}{K} \sum_j R_j(c; x)$. Our proposed conformal scoring function for arbitrary content selection tasks is **Ens4**, the mean ensemble of the four ICL-selection strategies mentioned above, using $k=2$ ICL examples per scorer (Algorithm 1). Validity of the coverage guarantee follows from our argument in Section 3.2.

5 Experiments

5.1 Experimental Setup

Code is available at github.com/layer6ai-labs/conformal-relevance. We evaluate on seven sentence-level relevance datasets spanning five domains (financial, encyclopedic, medical, general QA, and legal), four task types (summarization, question answering, entity/span detection, and clause scoring), and document lengths from 14 to 461 sentences (Table 2). Four datasets (HotpotQA, ECTSum, ContractNLI, Evidence Inference) were reformulated to sentence-level binary relevance with per-dataset construction details in App. D.1.

Metrics. We report two scoring-function quality measures. *Mean Average Precision* (MAP), the mean of per-sample AP across the test set, ranks the scoring function across the full score distribution and is the primary metric in Sections 5.2 and 5.3. *Conciseness*, $\mathbb{E}_{x \sim \text{test}, \text{MC splits}}[1 - |C_{\hat{q}}(x)|/|x|]$, is the expected fraction of sentences removed by the calibrated prediction set $C_{\hat{q}}(x)$ at fixed (α, β) , with the expectation taken over the test set. To account for variance in the conformal algorithm due to dataset splitting, we repeat the experiments with 400 random calibration/test splits and average metrics over these runs. Higher is better for both MAP and conciseness. We further examine the S_β order statistic underlying the calibrated threshold $\hat{q}$ on a per-sample level in Section 5.3. Empirical coverage at target level $1 - \alpha$ is reported in Section 5.4 as a validity check on the theoretical guarantees which must hold for any fixed scoring function under exchangeability.

Baselines. **ICL0**: a manually crafted per-dataset prompt that thoroughly describes what should be considered relevant for that task, with no ICL examples (full prompts in App. G). **Best Single**: a post-hoc oracle, the single highest MAP strategy configuration of the four ICL selection strategies, selected per-dataset with tuned hyperparameter k . The *Best Single (Config)* column in Table 3 reports the winning (σ_j, k) pair, and *Avg Single* reports the mean of the four strategies MAPs at $k=2$.

Default configuration. Unless stated otherwise, all **Ens4** experiments use $K=4$ scoring functions in the mean ensemble, created from the four ICL selection strategies (`anchor_dpp`, `pattern_dpp`, `bm25`, `random`), $k=2$ ICL examples per strategy, Gemini-2.5-Flash-Lite (Google Developers, 2025) as the scoring LLM, with fixed seeds. Single-strategy baselines use the same model and seeds. We sweep over the hyperparameter k for values $\{1, 2, 3, 5, 8\}$ with each strategy, both to find what value drives the Best Single oracle, and for the k -ablation of Section 5.3.

Table 3: Main MAP results on the seven datasets. **Bold** marks the best performing setting. Δ columns report the improvement of Ens4 over baselines with 95% bootstrap CI in brackets.

Dataset	ICL0	Avg Single ($k=2$)	Best Single (Config)	Ens4	Δ vs ICL0 [95% CI]	Δ vs Best Single
ECTSum	0.350	0.392	0.469 (anchor_dpp, $k=3$)	0.516	+0.166 _[+0.150,+0.181] (+47%)	+0.047 _[+0.032,+0.063] (+10%)
Evidence Inf.	0.206	0.201	0.213 (anchor_dpp, $k=1$)	0.304	+0.099 _[+0.093,+0.105] (+48%)	+0.091 _[+0.082,+0.100] (+43%)
HotpotQA	0.749	0.740	0.749 (anchor_dpp, $k=2$)	0.839	+0.090 _[+0.084,+0.096] (+12%)	+0.090 _[+0.084,+0.097] (+12%)
PhysioNet	0.764	0.766	0.795 (random, $k=3$)	0.879	+0.115 _[+0.083,+0.147] (+15%)	+0.080 _[+0.049,+0.112] (+10%)
PUMA	0.648	0.752	0.777 (anchor_dpp, $k=2$)	0.814	+0.165 _[+0.156,+0.175] (+25%)	+0.036 _[+0.027,+0.045] (+5%)
SubSumE	0.289	0.354	0.373 (bm25, $k=3$)	0.464	+0.170 _[+0.153,+0.187] (+59%)	+0.089 _[+0.072,+0.107] (+24%)
ContractNLI	0.718	0.711	0.733 (pattern_dpp, $k=3$)	0.828	+0.109 _[+0.099,+0.120] (+15%)	+0.095 _[+0.085,+0.105] (+13%)

5.2 Main Results

Table 3 displays our main results for MAP across all seven datasets, comparing Ens4 to the two baseline strategies and average performance. We emphasize that a single Ens4 configuration is used for all tests—only the datasets change. Ens4 improves MAP on all seven datasets, beating both ICL0 and the per-dataset best single strategy. To make the comparisons more explicit, we show the amount of improvement (Δ) of Ens4 over ICL0 and Best Single, with 95% confidence intervals computed over 10,000 bootstrap samples of the 400 measurements comprising each average value in the table. Improvements over ICL0 range from +0.090 to +0.170 MAP (+12% to +59%), with all confidence intervals excluding zero. This shows that ICL examples can capture the meaning of relevance across a variety of tasks better than a manually crafted description. Additional visualizations of the data are shown in App. H.

Several other patterns stand out from the data. **Complementarity.** For HotpotQA no single ICL strategy beats ICL0, yet Ens4 improves +0.090 MAP. Additionally, even though the $k = 2$ single strategies are weaker on average (Avg Single) than the Best Single configuration with k tuned, Ens4 using the $k = 2$ version of the strategies still outperforms. These observations demonstrate that ensembling over strategies of similar strength can lead to improvements due to complementarity.

Winner heterogeneity. No single sub-strategy dominates across datasets (Best Single column). Each of the four selection strategies is the best for at least one dataset, including the random strategy. This demonstrates that it can be difficult to choose a single ICL selection strategy that works broadly, whereas our Ens4 ensemble is consistently strong. **Statistical significance and robustness.** The bootstrap CIs in Table 3 lower-bound the improvement over ICL0 strictly above zero on every dataset. Because the seven benchmarks share a scoring LLM, prompt template, and seeds, we treat the 7/7 direc-

tional pattern as a consistency check rather than as seven independent trials.

Ens4 comes with higher computational burden than ICL0, using four calls to an LLM scorer instead of one, with negligible ICL-retrieval overhead. We analyze structural and empirical computational costs in App. H.

5.3 Ablations and Validation

We ablate key hyperparameters (k , K), validate the theory, isolate retrieval-induced diversity, and report framework robustness.

ICL examples per strategy (k). Existing research on general NLP tasks has reported inconsistent findings on whether additional ICL examples are helpful or harmful (Chen et al., 2023; Zou et al., 2025). The optimal number of ICL examples appears to be task-dependent, so we test here for conformal content selection tasks. In Table 4 (left) we vary $k \in \{1, 2, 3, 5, 8\}$ with $K=4$ fixed. $k=1$ underperforms $k=2$ on 6/7 datasets, likely due to insufficient guidance for pattern recognition. Higher values ($k \geq 5$) degrade performance on most datasets, which could be from DPP-kernel saturation and prompt-length effects starting to dilute instructions to the model. Instead, small values of $k \in \{2, 3\}$ show the most consistent and strong performance. We use $k=2$ in the default Ens4 since it is the more efficient of the two similar options.

Ensemble size (K) and composition. With $k = 2$ fixed, we run Ens K across all non-singleton combinations of the four ICL selection strategies (11 total configurations). Results are shown in Table 4 (right). Our default Ens4 performs best on every dataset, showing that ensembling more scoring functions, even potentially weak ones like random, remains beneficial due to complementarity gains. However, marginal gains are strictly decreasing across all 7 datasets going from $K = 3 \rightarrow 4$ compared to $2 \rightarrow 3$, consistent with Proposition 1’s $O(\frac{1}{K})$ bound and diminishing returns. See App. I.1 for further breakdown.

Table 4: Metric reported is MAP. Left: Ablation of ICL examples per strategy (k) in Ens4. Right: Ablation of ensemble size (K) in Ens K . For $K = 2$ and $K = 3$ we show the single best combination of all 4 ICL strategies over every combination.

Dataset	<i>k</i> ablation ($K=4$ fixed)					Oracle-best ensemble size ($k=2$ fixed)		
	$k=1$	$k=2$	$k=3$	$k=5$	$k=8$	Best Ens2	Best Ens3	Ens4
ECTSum	0.466	0.516	0.507	0.518	0.517	0.492	0.509	0.516
Evidence Inf.	0.307	0.304	0.303	0.300	0.293	0.261	0.288	0.304
HotpotQA	0.832	0.839	0.836	0.839	0.835	0.812	0.832	0.839
PhysioNet	0.864	0.879	0.874	0.869	0.861	0.856	0.874	0.879
PUMA	0.812	0.814	0.813	0.810	0.806	0.800	0.808	0.814
SubSumE	0.439	0.464	0.458	0.460	0.460	0.427	0.452	0.464
ContractNLI	0.821	0.828	0.833	0.832	0.820	0.787	0.816	0.828

Theory validation. App. J demonstrates empirical checks on the conditions and assumptions made throughout Section 3. Notably, we check how often the advantage condition from Lemma 1 is satisfied, that $\text{Comp} > S_{\max} - S_{\min}$, and how often this co-occurs with a measured advantage (see Table 17). When the condition is met, advantage is observed over 99% of the time.

ICL complementarity. We test the importance of complementarity compared to other ICL factors: quantity, supervision, and increased compute.

Complementarity vs. ICL quantity. Ens4 at $k=2$ deploys $K \cdot k = 8$ ICL examples for each query x , spread out over 4 scoring functions. This can be directly compared to Best Single at $k=8$, using the entire ICL budget in a single retrieval signal (App. I.2, Table 8). Still, Ens4 outperforms by +0.070 to +0.119 MAP on every dataset.

Complementarity vs. labeled data. To test whether Ens4’s gains can be explained by access to additional labeled examples, we train a supervised classifier using the available ICL pool and calibration data, which replaces the LLM-based ICL mechanism for scoring. Ens4 outperforms the classifier on 6/7 datasets (App. I.3, Table 9).

Complementarity vs. stochasticity and compute. We replace retrieval diversity with stochastic diversity by ensembling scoring functions with different temperature values $T \in \{0.3, 0.5, 0.8, 1.0\}$ for a single ICL selection strategy. These approaches have matched LLM compute budgets, isolating the type of diversity in the scoring functions. Results in App. I.4, Table 10 show that Ens4 outperforms the stochastic-diversity baseline in 12/16 settings, with overlapping CIs in the other 4. Individual temperature runs show relatively small MAP differences across T (Table 11). We additionally ensemble four temperature-varied ICL0 calls. This baseline directly matching Ens4’s LLM call count, but with-

out the ICL examples or retrieval complementarity. Ens4 significantly outperforms this baseline on 6/7 datasets, with a statistical tie on ContractNLI (Table 12).

Together these controls suggest retrieval-induced complementarity (Lemma 1) as the operative mechanism for improved MAP of the ensembled conformal score function.

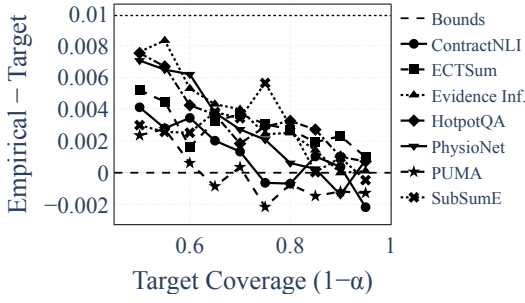
Robustness. We vary other design choices of our method to verify robustness.

LLM: We replace Gemini-2.5-Flash-Lite with Llama3-8B and Qwen3-8B on 4 representative datasets. Ens4 outperforms ICL0 on 7/8 (model, dataset) combinations and Best Single on 8/8. We further evaluate GPT-5.6-terra on seeded 500-sample test subsets of all seven datasets, where Ens4 outperforms both baselines on 7/7 datasets. Results are shown in App. I.5, Table 13.

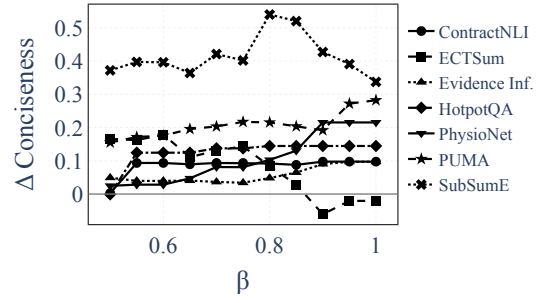
Task hint: The default Ens4 configuration includes in the scoring model prompt a one-line specification of the task. We test whether ICL examples are sufficient to describe the task on their own by removing the hint altogether, making the configuration and prompts completely dataset-agnostic. Results are shown in App. I.6, Table 14. 6/7 datasets stay within 0.04 MAP when the hint is removed. Only PhysioNet, which tests non-semantic personal health information, is materially affected, collapsing from 0.879 $\rightarrow$ 0.549 MAP.

5.4 Conformal Coverage and Conciseness

The preceding results focused on MAP, which is a measure of the accuracy of the conformal score function in isolation. We now evaluate the entire conformal pipeline which produces sets of content spans with coverage guarantees. First, we sweep over $\alpha \in \{0.05, \dots, 0.50\}$, with fixed $\beta=0.8$ to calibrate $\hat{q}$ on the calibration set, and use it to generate prediction sets on the test data. This process



(a) Empirical minus target coverage vs. $1-\alpha$. All datapoints remain within 1 pp. of the target. Dashed: bounds from Theorem 1 of Kuwahara et al. (2025).



(b) Conciseness gain $\Delta=\text{Ens4}-\text{ICL0}$ vs. β at $\alpha=0.20$.

Figure 1: Conformal coverage validity (left) and Ens4-ICL0 conciseness gain (right) across all 7 datasets ($n=100$).

is repeated 400 times with random splits of calibration and test data (while the ICL pool remains fixed), with results averaged. quality measure.

Coverage. Figure 1a shows the average empirical coverage of prediction sets produced with the Ens4 conformal score as $1-\alpha$ is varied. The dashed upper and lower bounds reflect the coverage guarantee of the algorithm we employ (Kuwahara et al., 2025). Empirical coverage remains within 1 pp of the target $1-\alpha$ coverage for all datasets (Figure 1a). The coverage random variable has variance that depends on calibration set size, but even though we use a modest 100 labeled samples, we find empirical coverage is consistently close to the target value.

Conciseness. Figure 1b measures conciseness - the fraction of sentences filtered out of the prediction set. Given a fixed coverage level, smaller prediction sets are more useful, and reflect a more powerful and confident scoring function. Compared to ICL0, Ens4 removes more content at $1-\alpha=0.8$ for all datasets at almost all tested values of β . This shows that the MAP gains of Ens4 do translate into more useful prediction sets. We perform a larger sweep over α values in App. K.

Alternative conformal ensemble rules. We additionally compare mean ensembling with alternative conformal ensemble rules, using the same four Ens4 constituent scoring functions in each case. Mean ensembling stays closest to the target coverage while achieving the highest conciseness. Majority voting under-covers at $\alpha=0.2$, whereas more conservative rules over-cover and produce less concise prediction sets (App. K.1, Table 19).

6 Conclusion

Many NLP tasks share a common core objective — extract relevant content from a document. The definition of relevance differs from one task to the next. In this work we introduced the *Conformal Relevance* framework for providing coverage guarantees over extractive content selection tasks with minimal manual input, instead relying on in-context learning to define relevance implicitly for any given task. Compared to past work, we replace task-specific scoring functions (an LLM with a hand-crafted prompt) by a mean ensemble of four ICL example selection strategies, and find that a single configuration outperforms bespoke prompts on seven datasets spanning five domains. Consequently, for a range of coverage and recall levels, Conformal Relevance filters out substantially more irrelevant content. We presented theoretical arguments to capture when a mean ensemble should improve, and the limitations for how much improvement can be gained by adding additional scorers. Empirically, we studied the importance of complementarity amongst the ICL selection strategies, and ablated this against quantity and stochasticity controls.

We pose three extensions for future work: replace the fixed mean aggregator with an adaptive scheme similar to (Angelopoulos et al., 2025), which would require an additional labeled data subset; extend binary relevance to graded labels (tiered or continuous) via conformal risk control (Angelopoulos et al., 2024); or to expand scope from purely extractive content selection to general generative tasks (Loaiza-Ganem et al., 2026).

Limitations

AI Use Disclosure: We used coding and research agents to support the theoretical discussions and experimental verification of this work. We used LLM tools to assist with writing.

The Conformal Relevance framework assumes access to ~ 150 – 440 labeled examples (the ICL pool plus 100 sample calibration set) which may need to be manually curated in practice; a static relevance criterion that does not drift between calibration and test to ensure exchangeability; and a one-line task hint which is manually crafted to explain what relevance means for a dataset, although this is optional and our method was largely unchanged on 6/7 datasets without it.

Our main results were reported on Gemini-2.5-Flash-Lite, with ablations on two 8B open-source LLMs. This is a limited set of models that have been surpassed in strength since this work was performed.

Our Ens4 ensemble of scoring functions requires $K = 4$ LLM calls per document, compared to one for the simpler ICL0 baseline. This is an example of test-time compute, where spending additional computation with LLMs can improve performance, but of course increases costs.

All experiments fix temperature $T=0$, seed, model version, and prompt templates, however there is residual variation from the hosted LLM’s server-side execution which is outside our control and can shift individual sample scores between runs. Replication should target aggregate MAP rather than per-sample scores.

The $1-\alpha$ guarantee for Conformal Relevance is marginal over the test distribution. Conditional coverage, for example over intents, document-length strata, or protected attributes, is not implied. Methods such as Mondrian conformal prediction (Vovk et al., 2003) can be used when conditional coverage is required.

The ICL selection strategies select examples from the labeled pool, but on datasets where positives are very sparse, the effective number of positive ICL candidates may be small, which may prevent the LLM from seeing sufficiently informative context. Larger or more carefully constructed pools may narrow this gap, at proportionally higher labeling cost.

We do not foresee any particular societal risks from use of our methods.

References

- Nabil Alami, Jad Zakharia, and Souhaib Ben Taieb. 2026. [Symmetric aggregation of conformity scores for efficient uncertainty sets](#). *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(24):19607–19614.
- Anastasios N. Angelopoulos and Stephen Bates. 2023. A gentle introduction to conformal prediction and distribution-free uncertainty quantification. *Foundations and Trends in Machine Learning*, 16(4):494–591.
- Anastasios N. Angelopoulos, Stephen Bates, Emmanuel J. Candès, Michael I. Jordan, and Lihua Lei. 2025. [Learn then test: Calibrating predictive algorithms to achieve risk control](#). *The Annals of Applied Statistics*, 19(2):1641–1662.
- Anastasios N. Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. 2024. Conformal risk control. In *International Conference on Learning Representations*.
- Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901.
- Debashish Chakraborty, Eugene Yang, Daniel Khoshabi, Dawn Lawrie, and Kevin Duh. 2026. [Principled Context Engineering for RAG: Statistical Guarantees via Conformal Prediction](#). In *Advances in Information Retrieval: 48th European Conference on Information Retrieval, ECIR 2026, Delft, The Netherlands, March 29 – April 2, 2026, Proceedings, Part II*, page 537–546. Springer-Verlag.
- Jiuhai Chen, Lichang Chen, Chen Zhu, and Tianyi Zhou. 2023. [How many demonstrations do you need for in-context learning?](#) In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11149–11159. Association for Computational Linguistics.
- John J. Cherian, Isaac Gibbs, and Emmanuel J. Candès. 2024. Large language model validity via enhanced conformal prediction methods. In *Advances in Neural Information Processing Systems*, volume 37, pages 114812–114842.
- Jesse C Cresswell. 2025. Trustworthy AI must account for interactions. *arXiv:2504.07170*.
- Jesse C. Cresswell, Bhargava Kumar, Yi Sui, and Mouloud Belbahri. 2025. Conformal Prediction Sets Can Cause Disparate Impact. In *The Thirteenth International Conference on Learning Representations*.

- Jesse C. Cresswell, Yi Sui, Bhargava Kumar, and Noël Vouitsis. 2024. Conformal Prediction Sets Improve Human Decision Making. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pages 9439–9457.
- Jay DeYoung, Eric Lehman, Ben Nye, Iain J. Marshall, and Byron C. Wallace. 2020. *Evidence inference 2.0: More data, better models*. Preprint, arXiv:2005.04177.
- Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Tianyu Liu, Baobao Chang, Zhifang Sui, and 1 others. 2023. A survey on in-context learning. *arXiv:2301.00234*.
- Margaret Douglass, Bill Long, George Moody, Peter Szolovits, Li-wei Lehman, Roger Mark, and Gari D. Clifford. 2007. *Deidentified Medical Text*. *PhysioNet*. Version 1.0.
- Naihe Feng, Yi Sui, Shiyi Hou, Jesse C. Cresswell, and Ga Wu. 2025. *Response quality assessment for retrieval-augmented generation via conditional conformal factuality*. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, page 2832–2836.
- Naihe Feng, Yi Sui, Shiyi Hou, Ga Wu, and Jesse C Cresswell. 2026. Conformal agent error attribution. *arXiv:2605.06788*.
- Ziang Gao, Pengqi Liu, Archer Yi Yang, Mouloud Belbahri, Jesse C Cresswell, and Masoud Asgharian. 2026. On the burden of achieving fairness in conformal prediction. *arXiv:2605.14260*.
- Matteo Gasparin and Aaditya Ramdas. 2024a. Conformal online model aggregation. *arXiv:2403.15527*.
- Matteo Gasparin and Aaditya Ramdas. 2024b. Merging uncertainty sets via majority vote. *arXiv:2401.09379*.
- Google Developers. 2025. Developers can now start building with Gemini 2.5 Flash. <https://blog.google/products/gemini/gemini-2-5-flash-preview/>. Accessed May 15, 2025.
- Ozgur Guldogan, Neeraj Sarna, Yuanyuan Li, and Michael Berger. 2026. Counterfactually fair conformal prediction. In *Proceedings of the 29th International Conference on Artificial Intelligence and Statistics*, volume 300 of *Proceedings of Machine Learning Research*. PMLR.
- Yuta Koreeda and Christopher D Manning. 2021. ContractNLI: A dataset for document-level natural language inference for contracts. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1907–1919.
- Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. 2017. Counterfactual fairness. In *Advances in Neural Information Processing Systems*, volume 30.
- Bruce Kuwahara, Chen-Yuan Lin, Xiao Shi Huang, Kin Kwan Leung, Jullian Arta Yapeter, Ilya Stanevich, Felipe Perez, and Jesse C. Cresswell. 2025. Document summarization with conformal importance guarantees. In *Advances in Neural Information Processing Systems*.
- Itay Levy, Ben Bogin, and Jonathan Berant. 2023. Diverse demonstrations improve in-context compositional generalization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1401–1422.
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474.
- Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022a. What makes good in-context examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114. Association for Computational Linguistics.
- Meichen Liu, Lei Ding, Dengdeng Yu, Wulong Liu, Linglong Kong, and Bei Jiang. 2022b. Conformalized fairness via quantile regression. In *Advances in Neural Information Processing Systems*, volume 35, pages 11561–11572. Curran Associates, Inc.
- Pengqi Liu, Zijun Yu, Mouloud Belbahri, Arthur Charpentier, Masoud Asgharian, and Jesse C. Cresswell. 2026. Beyond Procedure: Substantive Fairness in Conformal Prediction. In *Proceedings of the 43rd International Conference on Machine Learning*.
- Gabriel Loaiza-Ganem, Kevin Zhang, Wei Cui, Marc T Law, and Kin Kwan Leung. 2026. Conf-Gen: Conformal Uncertainty Quantification for Generative Models. In *Proceedings of the 43rd International Conference on Machine Learning*.
- Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098. Association for Computational Linguistics.
- Rui Luo and Zhixin Zhou. 2025. Conformity score averaging for classification. In *Proceedings of the*

- 42nd International Conference on Machine Learning, volume 267 of *Proceedings of Machine Learning Research*, pages 41586–41603. PMLR.
- Sewon Min, Xinxu Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. [Rethinking the role of demonstrations: What makes in-context learning work?](#) In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11048–11064. Association for Computational Linguistics.
- Christopher Mohri and Tatsunori Hashimoto. 2024. Language models with conformal factuality guarantees. In *Proceedings of the 41st International Conference on Machine Learning*.
- Rajdeep Mukherjee, Abhinav Bohra, Akash Banerjee, Soumya Sharma, Manjunath Hegde, Afreen Shaikh, Shivani Shrivastava, Koustuv Dasgupta, Niloy Ganguly, Saptarshi Ghosh, and Pawan Goyal. 2022. [ECT-Sum: A New Benchmark Dataset For Bullet Point Summarization of Long Earnings Call Transcripts](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10893–10906.
- Gauri Naik, Sharad Chandakacherla, Shweta Yadav, and Md Shad Akhtar. 2024. No perspective, no perception!! perspective-aware healthcare answer summarization. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 15919–15932.
- Ishna Neamatullah, Margaret M Douglass, Li-Wei H Lehman, Andrew Reisner, Mauricio Villarroel, William J Long, Peter Szolovits, George B Moody, Roger G Mark, and Gari D Clifford. 2008. Automated de-identification of free-text medical records. *BMC medical informatics and decision making*, 8(1):32.
- Eduardo Ochoa Rivera, Yash Patel, and Ambuj Tewari. 2025. [Conformal prediction for ensembles: Improving efficiency via score-based aggregation](#). In *Advances in Neural Information Processing Systems*, volume 38, pages 172791–172820.
- Ildikó Pilán, Pierre Lison, Lilja Øvrelid, Anthi Papadopoulou, David Sánchez, and Montserrat Batet. 2022. [The text anonymization benchmark \(TAB\): A dedicated corpus and evaluation framework for text anonymization](#). *Computational Linguistics*, 48(4).
- Mariia Ponomarenko, Sepideh Abedini, Masoumeh Shafieinejad, D. B. Emerson, Shubhankar Mohapatra, and Xi He. 2026. [CAPID: Context-Aware PII Detection for Question-Answering Systems](#). In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pages 320–331. Association for Computational Linguistics.
- Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of EMNLP-IJCNLP*.
- Stephen Robertson and Hugo Zaragoza. 2009. [The Probabilistic Relevance Framework: BM25 and Beyond](#). *Foundations and Trends in Information Retrieval*, 4(1-2):1–174.
- Yaniv Romano, Rina Foygel Barber, Chiara Sabatti, and Emmanuel Candès. 2020. With Malice Toward None: Assessing Uncertainty via Equalized Coverage. *Harvard Data Science Review*, 2(2).
- Ohad Rubin, Jonathan Herzig, and Jonathan Berant. 2022. Learning to retrieve prompts for in-context learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2655–2671. Association for Computational Linguistics.
- Glenn Shafer and Vladimir Vovk. 2008. A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9(3).
- Hao Shen, Zhouhong Gu, Haokai Hong, Weili Han, and Hongfeng Chai. 2026. [PII-Bench: Evaluating Query-Aware Privacy Protection Systems](#). In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4991–5026. Association for Computational Linguistics.
- Aldo Solari and Vera Djordjilović. 2022. [Multi split conformal prediction](#). *Statistics & Probability Letters*, 184:109395.
- Amber Stubbs, Christopher Kotfila, and Özlem Uzuner. 2015. Automated systems for the de-identification of longitudinal clinical narratives: Overview of 2014 i2b2/UTHealth shared task track 1. *Journal of Biomedical Informatics*, 58:S11–S19.
- Yi Sui, Tongzi Wu, Jesse Cresswell, Ga Wu, George Stein, Xiao Shi (Gary) Huang, Xiaochen Zhang, and Maksims Volkovs. 2024. Self-supervised Representation Learning from Random Data Projectors. In *International Conference on Learning Representations*, volume 2024, pages 6792–6814.
- Aditya T. Vadlamani, Anutam Srinivasan, Pranav Maneriker, Ali Payani, and Srinivasan Parthasarathy. 2025. A generic framework for conformal fairness. In *The Thirteenth International Conference on Learning Representations*.
- Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. 2005. *Algorithmic Learning in a Random World*. Springer.
- Vladimir Vovk, David Lindsay, Ilia Nouretdinov, and Alex Gammerman. 2003. Mondrian confidence machine. Technical Report CLRC-TR-03-04, Royal Holloway, University of London.
- Maja Waldron. 2026. CAOS: Conformal Aggregation of One-Shot Predictors. In *Proceedings of the 43rd International Conference on Machine Learning*.

- Fangxin Wang, Lu Cheng, Ruocheng Guo, Kay Liu, and Philip S Yu. 2024. Equal opportunity of coverage in fair regression. *Advances in Neural Information Processing Systems*, 36.
- Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, Guoyin Wang, and Chen Guo. 2025. [GPT-NER: Named Entity Recognition via Large Language Models](#). In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 4257–4275. Association for Computational Linguistics.
- Congbin Xu, Yue Yu, Haojie Ren, Zhaojun Wang, and Changliang Zou. 2025. Aggregating Conformal Prediction Sets via α -Allocation. *arXiv:2511.12065*.
- Nishant Yadav, Matteo Brucato, Anna Fariha, Oscar Youngquist, Julian Killingback, Alexandra Meliou, and Peter Haas. 2021. Subsume: A dataset for subjective summary extraction from wikipedia documents. In *Proceedings of the Third Workshop on New Frontiers in Summarization*, pages 131–141.
- Yachong. Yang and Arun Kumar Kuchibhotla. 2025. [Selection and Aggregation of Conformal Prediction Sets](#). *Journal of the American Statistical Association*, 120(549):435–447.
- Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.
- Jiacheng Ye, Zhiyong Wu, Jiangtao Feng, Tao Yu, and Lingpeng Kong. 2023. Compositional exemplars for in-context learning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202.
- Yanfei Zhou and Matteo Sesia. 2024. [Conformal classification with equalized coverage for adaptively selected groups](#). *Preprint*, arXiv:2405.15106.
- Kaijian Zou, Muhammad Khalifa, and Lu Wang. 2025. [On many-shot in-context learning for long-context evaluation](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 25605–25639. Association for Computational Linguistics.

Appendix Contents

Related Work (referenced by Section 2)

[A Additional Related Work](#)

Theory (referenced by Section 3)

[B Detailed Exchangeability Argument](#)

[C Ensemble Theory — Full Proofs and Additional Results](#)

Method (referenced by Section 4)

[D Reproducibility Details](#)

[E Prompting and ICL Specification](#)

[F ICL Selection Strategy Details](#)

[G ICL0 Prompts](#)

Experiments (referenced by Section 5)

[H Main Empirical Results](#)

[I Robustness Evaluations](#)

[J Empirical Validation of Ensemble Theory](#)

[K Conformal Conciseness Across \$\alpha\$](#)

Notation Used in the Appendix

Symbol	Meaning	First defined
<i>Span and document level</i>		
x, c	Document, content span	§2
y^*	Relevant (positive) sentence set	§2
$R_j(c; x)$	j -th relevance function, $\in [0, 1]$	§2
<i>Conformal calibration</i>		
α, β	Coverage miscoverage budget; recall target	§2
r	Order-statistic index: $r = y^* - \lceil \beta y^* \rceil + 1$	§2
$\hat{q}$	Calibrated conformal threshold	§2
Π	Pool of ICL examples	§3
<i>Ensemble level</i>		
K, k	Number of scoring functions R_j ; ICL examples per strategy	§2
$\bar{R}^{(K)}$	Mean ensemble of scoring functions; $\bar{R}^{(K)} = \frac{1}{K} \sum_j R_j$	§3
g	Score aggregation function (we fix $g = \text{mean}$)	§3
$S_\beta, S_\beta^{(K)}$	Conformal score using scoring function $R_j, \bar{R}^{(K)}$	§3
$\delta^K(c), \delta_\beta^K(c)$	Per-sentence margin above floor: $\bar{R}^{(K)}(c) - S^{(K)}$, etc.	§3
$S_{\max}, S_{\min}$	max, min of S across the $K=2$ pair	§3
Comp	Complementarity measure	§3
A_j	Set of minimizers for R_j : $\{c \in y^* : R_j(c) = S(R_j)\}$	App C.2

A Additional Related Work

There are two major approaches to ensembling applied to conformal prediction: set-level and score-level. In set-level prediction, multiple scoring functions each run the conformal algorithm to produce prediction sets that individually come with conformal coverage guarantees. Then the sets are merged in some way, maintaining an overall guarantee on coverage, which may not be as tight as the individual guarantees, but with the objective of reducing final set size. Methods in this direction include multi-split conformal prediction (Solari and Djordjilović, 2022), majority vote merging (Gasparin and Ramdas, 2024b), conformal online model aggregation (Gasparin and Ramdas, 2024a), and confidence-level allocation (Xu et al., 2025).

Our work is more closely related to past works on score-level aggregation. Ochoa Rivera et al. (2025) discuss aggregating multiple scoring functions into a single conformal score, and introduce a particular method for conformal score aggregation (CSA) on classification problems. They establish the central structural observation that aggregating scores preserves more information than aggregating prediction sets. Our contribution is the formalization of the recall-style coverage guarantee setting which is more applicable to NLP tasks. Other relevant score-level works include conformal score averaging (Luo and Zhou, 2025), symmetric aggregation (Alami et al., 2026), efficiency/validity first conformal prediction (Yang and Kuchibhotla, 2025), and conformal aggregation of one-shot predictors (Waldron, 2026).

Conformal prediction can be seen as a type of uncertainty quantification. By applying it, users gain more information about how confident their model’s predictions are, as larger sets indicate greater uncertainty (Angelopoulos and Bates, 2023). However, when applying conformal prediction in practice, one must be aware of potential negative interactions with other aspects of trust (Cresswell, 2025). In particular, researchers have considered whether conformal prediction can impact fairness. When datapoints are stratified by group identities, Romano et al. (2020) proposed that ensuring each group receives prediction sets with equal coverage should be a fairness goal (Zhou and Sesia, 2024). This is a type of procedural fairness, since it relates to properties of the algorithm itself. However, Gao et al. (2026) found that there can be inherent tradeoffs between fairness and cov-

erage disparity.

In contrast to the above, (Cresswell et al., 2024, 2025) proposed that fairness should consider the downstream impacts of conformal prediction, that is, how sets are used, not just how they are produced. They found that equalizing set size, not coverage, resulted in the most fair outcomes on downstream tasks. This was further confirmed in larger scale experiments (Liu et al., 2026).

Other works have borrowed traditional notions of fairness in machine learning to apply them to conformal prediction, such as demographic parity (Liu et al., 2022b), Equal Opportunity (Wang et al., 2024), while Vadlamani et al. (2025) create a framework that supports many definitions. Counterfactual fairness (Kusner et al., 2017) has also been applied in a conformal setting (Guldogan et al., 2026). Practitioners should thus be aware to evaluate their deployments of conformal prediction for fairness impacts, and employ suitable methods to improve fairness where possible.

B Detailed Exchangeability Argument

This appendix extends the validity argument in Section 3.2 for greater clarity.

For fixed-ICL scorers, ICL examples are drawn from the pool Π , which is disjoint from the calibration and test sets. Once drawn and added to the LLM’s prompt, each relevance scoring function $R_j(c; x)$ is a deterministic function of the input, so exchangeability of calibration and test samples directly implies exchangeability of conformal scores. The ensemble case is immediate: $\bar{R}(c; x) = \frac{1}{K} \sum_j R_j(c; x)$ is also a deterministic function of the input once all K ICL example sets are fixed, and the conformal guarantee applies to any fixed scoring function.

B.1 Per-Sample ICL Scorers

For retrieval-based ICL selection (e.g., BM25), the ICL examples depend on the input x via the selection function $f(x, \text{pool})$. Conditioning on the fixed pool, this is still a deterministic function of x , and exchangeability of calibration and test data is preserved.

Formal argument (tower property). Let Π denote the pool draw. For any fixed realization π of Π , the relevance scoring function $R_j(\cdot; \cdot \mid \Pi = \pi)$ is a deterministic function, so the conformal guarantee holds conditionally:

$$P[S_\beta \leq \hat{q} \mid \Pi = \pi] \leq \alpha \quad \text{for all } \pi. \quad (9)$$

By the tower property:

$$P[S_\beta \leq \hat{q}] = \mathbb{E}_\Pi[P[S_\beta \leq \hat{q} \mid \Pi]] \leq \alpha, \quad (10)$$

so the guarantee holds unconditionally.

B.2 What Would Break Exchangeability

Two scenarios would break the argument:

1. **Data leakage:** drawing ICL examples from the calibration or test set rather than the disjoint pool, which couples the scoring function to the calibration data.
2. **Data-driven aggregation:** letting the aggregation function depend on calibration data, introducing drift in the conformal score distribution between calibration and test.

The three-way split (pool drawn first, calibration and test drawn independently) prevents scenario 1. The fixed aggregation prevents scenario 2.

This is precisely [Ochoa Rivera et al. \(2025\)](#)’s concern: their data-driven quantile envelopes require a two-stage calibration split to restore exchangeability, which our fixed mean ensembling avoids.

C Ensemble Theory — Full Proofs and Additional Results

This appendix contains full proofs of all theoretical results stated in Section 3, along with additional results: the ensemble floor inequality, the $\beta < 1$ extension (generalized Comp_β decomposition, sacrifice fragmentation characterization, and the β dichotomy), set containment bounds, conciseness guarantees, and variance reduction.

C.1 Setup and Notation

We first recall the notation from Section 3 for convenience. We have K scoring functions $R_1, \dots, R_K$, each assigning a relevance score in $[0, 1]$ to every content span c . The mean ensemble scoring function averages them per-span:

$$\bar{R}(c; x) = \frac{1}{K} \sum_{j=1}^K R_j(c; x). \quad (11)$$

The positive set y^* contains $|y^*|$ ground truth relevant content spans. For a given scoring function (either one of the R_j or $\bar{R}^{(K)}$) and a given x , sort the scores for positive spans in ascending order: $R_{(1)} \leq R_{(2)} \leq \dots \leq R_{(|y^*|)}$. Then:

$$S_\beta(x, y^*) = R_{(r)} \quad \text{where } r = |y^*| - \lceil \beta |y^*| \rceil + 1.$$

At $\beta = 1$: $r = 1$, so $S(x, y^*) = \min_{c \in y^*} R(c; x)$.

C.2 Ensemble floor advantage proof

We present the full proof of Lemma 1 from Section 3.3.

Lemma 1. (*Ensemble floor advantage*) At $K = 2$ and $\beta = 1$,

$$S^{(2)} = \frac{1}{2}(S_{\max} + S_{\min} + \text{Comp}), \quad (7)$$

and $S^{(2)} > S_{\max}$ iff $\text{Comp} > S_{\max} - S_{\min}$.

Proof. For convenience, we recall the definition of complementarity from Definition 1,

$$\begin{aligned} \text{Comp}(R_1, R_2) &= \min_{c \in y^*} [R_1(c) + R_2(c)] \\ &\quad - [S(R_1) + S(R_2)]. \end{aligned}$$

We note that non-negativity of Comp follows from superadditivity of the minimum. The mean ensemble’s conformal score is

$$S^{(2)} = \frac{1}{2} \min_{c \in y^*} [R_1(c) + R_2(c)]. \quad (12)$$

Substituting the definition of Comp directly gives

$$\begin{aligned} S^{(2)} &= \frac{1}{2} [S(R_1) + S(R_2) + \text{Comp}] \\ &= \frac{1}{2} [S_{\max} + S_{\min} + \text{Comp}]. \end{aligned} \quad (13)$$

The mean ensemble has an advantage when $S^{(2)} > S_{\max}$, which occurs exactly when

$$S^{(2)} - S_{\max} = \frac{1}{2} [\text{Comp} - (S_{\max} - S_{\min})] > 0, \quad (14)$$

or $\text{Comp} > S_{\max} - S_{\min}$. $\square$

Remark (Disjoint minimizers). Let $A_j = \{c \in y^* : R_j(c) = \min_{c'} R_j(c')\}$ be the set of positive content spans with minimal score according to R_j . If $\bigcap_j A_j = \emptyset$ (no content span is a minimizer for all R_j simultaneously), then $S(\bar{R}) > \frac{1}{K} \sum_j S(R_j)$ strictly. This is because for every c , since $c \notin \bigcap_j A_j$, there exists some scoring function R_j with $R_j(c) > A_j$ strictly, giving $\bar{R}(c) > \frac{1}{K} \sum_j R_j(a)$ for all $a \in A_j$.

Worked example. Consider $K = 2$ and $p = 3$ relevant content spans:

Content span	R_1	R_2	$\bar{R}$
c_1	0.8	0.6	0.70
c_2	0.4	0.9	0.65
c_3	0.7	0.5	0.60

Here $S(R_1) = 0.4$, $S(R_2) = 0.5$, so $S_{\max} = 0.5$ and $S_{\min} = 0.4$. The complementarity measure is $\text{Comp} = \min\{1.4, 1.3, 1.2\} - 0.9 = 0.3$, which exceeds the floor gap of $S_{\max} - S_{\min} = 0.1$. The ensemble achieves $S^{(2)} = 0.60 > 0.50 = S_{\max}$: R_1 's weakness on c_2 is compensated by R_2 , and vice versa for c_1 . If both scorers shared the same weakness ($R_2(c_2) = 0.4$), Comp would merely match the floor gap (both being zero) and the ensemble would not improve.

C.3 $O(\frac{1}{K})$ diminishing returns proof

We present the full proof of Proposition 1 from Section 3.4.

Proposition 1. ($O(\frac{1}{K})$ diminishing returns) *We have $S^{(K+1)} > S^{(K)}$ iff $R_{K+1}(c) > S^{(K)} - K \delta^K(c)$ for every $c \in y^*$. When the condition holds,*

$$0 < S^{(K+1)} - S^{(K)} \leq \frac{1 - S^{(K)}}{K + 1} = O\left(\frac{1}{K}\right). \quad (8)$$

Proof. The new ensemble average is:

$$\bar{R}^{(K+1)}(c) = \frac{K \bar{R}^{(K)}(c) + R_{K+1}(c)}{K + 1}. \quad (15)$$

To raise the ensemble floor $S^{(K+1)} > S^{(K)}$, we require $\bar{R}^{(K+1)}(c) > S^{(K)}$ for every $c \in y^*$. Substituting and rearranging:

$$\begin{aligned} K \cdot \bar{R}^{(K)}(c) + R_{K+1}(c) &> (K + 1) \cdot S^{(K)} \\ R_{K+1}(c) &> S^{(K)} - K \cdot \delta^K(c). \end{aligned} \quad (16)$$

For the converse direction, we simply follow the algebraic manipulations in reverse.

Next we consider the bound in Equation (8). Let $c^* = \arg \min_{c \in y^*} \bar{R}^{(K)}(c)$ such that $\bar{R}^{(K)}(c^*) = S^{(K)}$. We then have

$$\begin{aligned} S^{(K+1)} &= \min_{c \in y^*} \bar{R}^{(K+1)}(c) \\ &\leq \bar{R}^{(K+1)}(c^*) = \frac{K S^{(K)} + R_{K+1}(c^*)}{K + 1}. \end{aligned} \quad (17)$$

Since relevance scores are in $[0, 1]$ by definition, $R_{K+1}(c^*) \leq 1$:

$$S^{(K+1)} \leq \frac{K S^{(K)} + 1}{K + 1}. \quad (18)$$

Subtracting $S^{(K)}$ from both sides yields the upper bound:

$$S^{(K+1)} - S^{(K)} \leq \frac{1 - S^{(K)}}{K + 1} = O\left(\frac{1}{K}\right), \quad (19)$$

where we note that $0 \leq S^{(K)} \leq 1$ again by our definitions. $\square$

C.4 Extension to $\beta < 1$

This subsection formalizes the extension of results from Section 3 to $\beta < 1$, now focusing on the conformal score function $S_\beta(x, y^*)$ from Equation (3).

Generalized Comp_β decomposition. For $\beta < 1$ and $K = 2$ scorers, define

$$\begin{aligned} \text{Comp}_\beta(R_1, R_2) &= \text{OrderStat}_r(\{R_1(c) + R_2(c) : c \in y^*\}) \\ &\quad - [S_\beta(R_1) + S_\beta(R_2)], \end{aligned} \quad (20)$$

using the r 'th order statistic over positive.

Lemma 2 (General ensemble floor advantage). *With $S_{\beta, \max} = \max(S_\beta(R_1), S_\beta(R_2))$ and $S_{\beta, \min} = \min(S_\beta(R_1), S_\beta(R_2))$, the mean ensemble floor*

$$S_\beta^{(2)}(x, y^*) = \bar{R}_{(r)}^{(2)}(c; x) \quad (21)$$

improves, $S_\beta^{(2)} > S_{\beta, \max}$, iff $\text{Comp}_\beta > S_{\beta, \max} - S_{\beta, \min}$.

Proof. If we use the OrderStat notation, the ensemble scoring function is $S_\beta^{(2)} = \frac{1}{2} \text{OrderStat}_r(\{R_1(c) + R_2(c)\})$. Substituting the Comp_β definition gives $S_\beta^{(2)} = \frac{1}{2}(S_{\beta, \max} + S_{\beta, \min} + \text{Comp}_\beta)$. Hence we immediately find that $S_\beta^{(2)} - S_{\beta, \max} > 0$ exactly when $\text{Comp}_\beta > S_{\beta, \max} - S_{\beta, \min}$. $\square$

We can equivalently write this complementarity condition as $S_\beta^{(2)} > S_{\beta, \max}$ iff at least $\lceil \beta |y^*| \rceil$ positives satisfy $\bar{R}(c) > S_{\beta, \max}$.

Proof. $S_\beta^{(2)}$ exceeds $S_{\beta, \max}$ iff at most $r - 1$ values are $\leq S_{\beta, \max}$, i.e., at least $|y^*| - r + 1 = \lceil \beta |y^*| \rceil$ values are $> S_{\beta, \max}$. At $\beta = 1$ this reduces to Lemma 1's "for all $c \in y^*$ " condition. $\square$

Note that the min function (order statistic $r = 1$) is concave, so $\text{Comp}_1 \geq 0$ always holds; higher-order statistics ($r > 1$) are not concave, and Comp_β can be strictly negative. Averaging scoring functions at $\beta = 1$ cannot lower the floor, but we have no such guarantee at $\beta < 1$.

Proposition 2 (General $O(\frac{1}{K})$ diminishing returns). Adding R_{K+1} gives $S_{\beta}^{(K+1)} > S_{\beta}^{(K)}$ iff at least $\lceil \beta |y^*| \rceil$ positives satisfy $R_{K+1}(c) > S_{\beta}^{(K)} - K\delta_{\beta}^K(c)$, with $\delta_{\beta}^K(c) = \bar{R}^{(K)}(c) - S_{\beta}^{(K)}$. When this condition holds,

$$0 < S_{\beta}^{(K+1)} - S_{\beta}^{(K)} \leq \frac{1 - S_{\beta}^{(K)}}{K+1} = O\left(\frac{1}{K}\right). \quad (22)$$

Proof. Because $S_{\beta}^{(K)}$ is defined as the r -th order statistic of scoring functions on $c \in y^*$, for all $i \in \{1, 2, \dots, r\}$ with $r \geq 1$, we have

$$\bar{R}_{(i)}^{(K)} = \bar{R}^{(K)}(c_{(i)}) \leq S_{\beta}^{(K)} \quad (23)$$

Under the expanded ensemble $\bar{R}^{(K+1)}$, the result for any $i \leq r$ is

$$\bar{R}^{(K+1)}(c_{(i)}) = \frac{K\bar{R}^{(K)}(c_{(i)}) + R_{K+1}(c_{(i)})}{K+1}. \quad (24)$$

Using $R_{K+1}(c) \leq 1$ (since $R_{K+1}(c) \in [0, 1]$ by definition),

$$\bar{R}^{(K+1)}(c_{(i)}) \leq \frac{KS_{\beta}^{(K)} + 1}{K+1}. \quad (25)$$

Because all r spans in $\{c_{(1)}, \dots, c_{(r)}\}$ satisfy this inequality, there exist at least r positive content spans whose new ensemble scores do not exceed $\frac{KS_{\beta}^{(K)} + 1}{K+1}$. By definition of the r -th order statistic, the r -th smallest score $S_{\beta}^{(K+1)}$ cannot exceed the maximum score among any subset of r elements. Thus,

$$S_{\beta}^{(K+1)} \leq \frac{KS_{\beta}^{(K)} + 1}{K+1}. \quad (26)$$

Subtracting $S_{\beta}^{(K)}$ from both sides yields

$$S_{\beta}^{(K+1)} - S_{\beta}^{(K)} \leq \frac{1 - S_{\beta}^{(K)}}{K+1}. \quad (27)$$

□

D Reproducibility Details

This appendix documents the settings required to reproduce our pipeline: model, sampling temperature, random seed, prompt templates, data splits, and calibration size.

Model and inference settings. All experiments use Gemini-2.5-Flash-Lite (Google Developers, 2025), accessed January–April 2026) at sampling temperature $T=0$. The random seed is fixed for ICL pool sampling, stratified selection, DPP subset sampling, negative sub-sampling in contrastive formatting, and calibration/test splits. Scoring and retry prompts are given verbatim in Appendix E.1 and Appendix G.

ICL embedding model. Both embedding-based ICL selection strategies (anchor_dpp, pattern_dpp) use all-MiniLM-L6-v2 (Reimers and Gurevych, 2019) via the sentence-transformers library. This 22M-parameter model produces 384-dimensional embeddings.

Data splits and sample counts. Data split sizes and per-dataset sample counts are reported in Table 2. Calibration size is fixed at $n = 100$ across all datasets

Pool and calibration sizing rationale. The ICL pool size is set to 50 for single-intent datasets, or 20 per intent for multi-intent datasets (see Table 2). With $k=2$ examples per sub-scoring function and $K=4$ sub-scoring functions, only 8 pool examples are consumed per test sample, but a larger pool lets the diversity-maximizing strategies (anchor_dpp, pattern_dpp) select from a richer candidate set per input, improving complementarity. A calibration size of $n = 100$ balances conformal threshold variance against annotation cost, and gives reasonably precise coverage estimates in our experiments. Both sizes are fixed across all datasets, making the budget requirement predictable for practitioners.

Code availability. All experiment code, ICL strategy implementations, and data processing scripts are available at github.com/layer6ai-labs/conformal-relevance.

D.1 Dataset Reformulations

Four of the seven datasets have their native task reformulated to per-sentence binary relevance. The mapping from native task to sentence-level positive definition is:

- **HotpotQA** (Yang et al., 2018) — Native: multi-hop QA with supporting-fact annotations. Sentence-level relevance: a context sentence is positive when it is annotated as a supporting

fact for the question. The question text fills the `intent` field.

- **ECTSum** (Mukherjee et al., 2022) — Native: abstractive financial-call summarization. Sentence-level relevance: a transcript sentence is positive when it is included in the human-written summary (matched by ROUGE-based alignment). No intent / single-intent dataset.
- **ContractNLI** (Koreeda and Manning, 2021) — Native: contract NLI (entailment/contradiction/neutral hypothesis classification). Sentence-level relevance: a contract span is positive when it appears in the gold evidence set for the (contract, hypothesis) pair. The 17 hypotheses act as intents (stratified ICL); only Entailment + Contradiction pairs are retained as usable.
- **Evidence Inference** (DeYoung et al., 2020) — Native: clinical-trial outcome inference. Sentence-level relevance: an article sentence is positive when it is annotated as an evidence span for the (intervention, comparator, outcome) prompt. The 8 user splits act as intents (stratified ICL).

The remaining three datasets (SubSumE, PUMA, PhysioNet) carry sentence-level relevance labels in their native form; no reformulation is applied. PhysioNet sentences come from the PhysioNet Deidentified Medical Text v1.0 corpus (Douglass et al., 2007; Neamatullah et al., 2008), and per-sentence PHI labels follow the i2b2 2014 deidentification schema (Stubbs et al., 2015) applied on top of the corpus.

E Prompting and ICL Specification

In this appendix, we provide information on prompts and other details required to run our method.

E.1 Contrastive ICL Format Specification

This appendix provides the full contrastive ICL format specification used to present labeled examples to the LLM, the scoring prompt template, and the retry logic for handling malformed LLM outputs. The contrastive format lists positive sentences (“Sentences selected”) and a balanced sample of negative sentences (“Sentences NOT selected”), with a near-1:1 ratio and a minimum of two negatives.

Scoring prompt template. The base prompt instructs the LLM to infer a selection distinction from the ICL block and apply it to the evaluation section. Placeholders: `{icl_examples}` is replaced by the formatted ICL block; `{task_line}` is an optional one-line hint prepended for domain-specific tasks (empty string otherwise); `{intent}` is the per-sample query or perspective; `{sentences}` is the numbered sentence list. For no-intent datasets (ECTSum, PhysioNet), the `Intent :` line is omitted entirely and a variant template is used.

```
In the examples below, some sentences were selected
↪ and others were not. Identify what
↪ distinguishes the selected sentences from the
↪ non-selected ones. Then apply that SAME
↪ distinction to score each sentence in the
↪ evaluation section.
```

```
Each score should be a two decimal float between 0
↪ and 1. A sentence that clearly matches the
↪ demonstrated distinction should score close to 1
↪ (> 0.8). A sentence that does not match should
↪ score close to 0 (< 0.2). Use intermediate
↪ scores (0.3, 0.5, 0.7) for partial matches.
```

```
IMPORTANT: Output must be valid JSON format with
↪ sentence indices as keys.
```

```
{icl_examples}
Now evaluate the following:
```

```
{task_line}Intent: {intent}
Sentences to evaluate:
{sentences}
```

```
Scores (JSON format):
```

Contrastive ICL example format. Each pool example is formatted as: the full sentence list (numbered 0-indexed), then the selected positives as a comma-separated list of quoted sentence texts, then a balanced sample of negatives under the “NOT selected” header. Negatives are sub-sampled to achieve a near-1:1 ratio with positives, with a minimum of two negatives; this is capped by the number of available negatives. Sub-sampling uses a per-example deterministic seed derived from the global seed. Multiple examples are concatenated with double newlines and numbered (Example 0:, Example 1:, ...), separated by “--” dividers.

An example block for an intent-bearing dataset looks as follows (shortened for illustration):

```
Example 0:
Intent: History
Sentences to evaluate:
0. The city was founded in 1832.
1. The annual rainfall averages 800 mm.
2. It was incorporated as a borough in 1901.
3. The main industry is tourism.
Sentences selected for "History": "The city was
↪ founded in 1832.", "It was incorporated as a
↪ borough in 1901."
Sentences NOT selected for "History" (examples):
↪ "The annual rainfall averages 800 mm.", "The
↪ main industry is tourism."
---
```

For no-intent datasets, the `Intent :` line is omitted and the labels read `Sentences`

selected: and Sentences NOT
selected (examples):.

Retry prompt. When the LLM response omits sentence indices, the missing sentences are re-submitted using the retry template below. The same ICL block is included to maintain consistency with the original scoring context.

You previously scored sentences but missed some. In
→ the examples below, some sentences were
→ selected and others were not. Identify what
→ distinguishes the selected sentences from the
→ non-selected ones. Then apply that SAME
→ distinction to score each MISSED sentence
→ below.

Each score should be a two decimal float between 0
→ and 1. A sentence that clearly matches the
→ demonstrated distinction should score close to 1
→ (> 0.8). A sentence that does not match should
→ score close to 0 (< 0.2). Use intermediate
→ scores (0.3, 0.5, 0.7) for partial matches.

IMPORTANT: Output must be valid JSON format with
→ sentence indices as keys.

{icl_examples}
Now evaluate the following:

{task_line} Intent: {intent}
Missing sentences:
{missing_sentences}

Scores (JSON format with ONLY the missing sentence
→ numbers as keys):

Format as controlled variable. The contrastive format is held constant across all experimental conditions: ICL-free baselines (ICL0), individual strategies, and ensembles. All comparisons in Table 3 (main result) and Table 7 (composition ablation) are therefore measured within the same format, and ensemble gains over single strategies are attributable to strategy diversity, not prompt formatting. A format–strategy interaction is theoretically possible: different strategies select examples with different positive/negative ratios, and the contrastive sub-sampling handles these differently. However, such an interaction would need to *systematically* favor diverse strategy sets over individual strategies across all seven datasets—a strong and specific assumption unsupported by any proposed mechanism. Preliminary experiments during development confirmed that the contrastive layout outperforms both positive-only (no negatives shown) and full-document (all sentences labeled, no sub-sampling) alternatives for individual strategies; a full format × ensemble factorial ablation is left to future work.

E.2 Windowed Context Algorithm and Ablation

This appendix describes the windowed ICL compression algorithm used for long-document datasets

and presents ablation results comparing windowed and full ICL on SubSumE (average 461 sentences per document). The algorithm retains all positive sentences, asymmetric random-width context windows around each positive, and a sample of remote negatives, substantially reducing prompt length while preserving ICL quality.

Algorithm. Windowed context is applied automatically when a dataset’s average document length exceeds 30 sentences or its average positive rate falls below 15%; both SubSumE and Evidence Inference trigger this condition, while PUMA, ECT-Sum, PhysioNet, HotpotQA, and ContractNLI do not. For each ICL pool example selected by a retrieval strategy, the full sentence list is replaced by a compressed version retaining: (i) all positive sentences; (ii) a local context window around each positive with independently sampled width in each direction (left width $\sim \mathcal{U}(1, w_L)$, right width $\sim \mathcal{U}(1, w_R)$, with $w_L = w_R = w$); and (iii) a uniform random subsample of sentences outside all windows (“remote negatives”), with subsample size set to match the number of context-window negatives. The context-window half-width w is fixed at $w = 2$ throughout. For test documents, no windowing is applied: the full sentence list is scored in a single LLM call.

F ICL Selection Strategy Details

This appendix provides detailed algorithmic descriptions for the four ICL selection strategies σ_j used in the Ens4 ensemble: `anchor_dpp` (centroid anchor with conditional DPP), `pattern_dpp` (DPP over relevance-direction vectors), `bm25` (lexical retrieval with the BM25 algorithm), and `random` (uniform sampling from the ICL). We include DPP kernel formulations, embedding model specifications, and BM25 parameter settings.

Anchor-DPP (`anchor_dpp`). A two-phase selection procedure that combines centroid proximity with conditional diversity. First, the query document x is embedded via a language embedding model (see App. D). In Phase 1, the example document from the ICL pool which has embedding nearest to the query embedding as measured by cosine similarity is selected as the anchor, and included in the final ICL set. In Phase 2, the remaining $k - 1$ examples are drawn from a conditional Determinantal Point Process (DPP) whose kernel

is the cosine Gram matrix of pool embeddings, ridge-regularized to ensure positive definiteness. Conditioning on the anchor biases the sample toward examples that are both semantically close to the query document and mutually diverse relative to it. This strategy targets ICL pool examples that are representative of the test domain while remaining internally diverse, reducing the chance that all k examples provide redundant information to the LLM scorer.

Pattern-DPP (`pattern_dpp`). This DPP operates in the space of sentence-level embeddings rather than full-document embeddings. For each ICL pool example, a “relevance direction” is computed as the difference between the centroid embedding of its positive spans ($c \in y^*$) and the centroid embedding of its negative spans ($c \notin y^*$). The relevance direction captures the specific distinction of content relevance the document exemplifies. The k ICL examples are then selected by running a DPP over the L2-normalized relevance directions to maximize diversity. Because this strategy selects examples whose relevance directions are maximally orthogonal, the resulting scoring function captures a different cross-section of the relevance signal than `anchor_dpp`.

BM25 (`bm25`). This strategy uses lexical retrieval with the Okapi BM25 algorithm (Robertson and Zaragoza, 2009). For each intent within the ICL pool, or the full pool for uni-intent datasets, a BM25 index is built over the concatenated text of documents, tokenized by whitespace. At scoring time, the query document x is used as a query to the index and the k highest-scoring documents are retrieved. BM25 captures lexical overlap that semantic embeddings may miss—recurring domain-specific terminology, abbreviations, or proper nouns—and is entirely independent of the embedding model used by `anchor_dpp` and `pattern_dpp`, making it a structurally distinct signal within the ensemble.

Random (`random`). This strategy samples documents uniformly at random without replacement from the intent-stratified pool. Given a query document x , k examples are drawn from the ICL pool Π using a fixed seed, and independently of the query document’s content or embedding. This strategy introduces no retrieval bias and serves as an ensemble regularizer: its scores are uncorrelated with any document-specific signal, so averaging with

embedding-based scorers dampens their shared biases rather than reinforcing them.

G ICL0 Prompts

This appendix contains the task-specific prompt templates used for the ICL0 baseline. The ECTSum prompt was adapted from the Importance Scoring prompt of Kuwahara et al. (2025), with minor modifications for sentence-level extraction and output formatting; prompts for the remaining tasks followed the same structure with manually written task descriptions. For six of seven datasets, the prompts were written once without empirical tuning on labeled data. For Evidence Inference, candidate variants were evaluated on the 100-sample calibration set, and the selected prompt was evaluated once on the untouched test set.

G.1 ECTSum

Financial earnings call summarization. The scoring function evaluates sentence-level alignment with the overall message of the transcript. No intent placeholder; the task is fully specified by the prompt framing.

```
Evaluate the importance of each input sentence in
→ the original text, based on how the information
→ carried in the sentence is aligned with the
→ overall message. and provide a importance score
→ for EACH input sentence. Each output score
→ should be a two decimal float number ranged
→ between 0 and 1, indicating how important the
→ corresponding input sentence is in the context
→ of the text document. For example, if sentence
→ 1's information is highly aligned with that of
→ the input text, and very likely to be included
→ in the summary, then score 1 should be close to
→ 1, say greater than 0.8; if information carried
→ in sentence 3 is trivial or only remotely
→ related to the central message of the text, and
→ is not worthy of inclusion in the summary, then
→ score 3 should be close to 0, say less than
→ 0.2.
```

```
IMPORTANT: Output must be valid JSON format with
→ sentence indices as keys.
```

```
Sentences to evaluate:
{sentences}
```

```
Importance scores (JSON format):
```

G.2 SubSumE

Query-focused summarization over Wikipedia articles. `{intent}` is filled at runtime with one of 10 query intents (e.g., “History”, “Geography”).

```
Evaluate each sentence based on how well it answers
→ or provides information relevant to the given
→ query. Provide a score for EACH sentence. Each
→ score should be a two decimal float between 0
→ and 1.
```

```
For example, if a sentence directly answers or
→ supports the query, its score should be close
→ to 1 (greater than 0.8); if a sentence is
→ unrelated to the query or provides only
→ irrelevant background, its score should be
→ close to 0 (less than 0.2).
```

IMPORTANT: Output must be valid JSON format with
→ sentence indices as keys.

Query: {intent}
Sentences to evaluate:
{sentences}

Scores (JSON format):

G.3 PUMA

Perspective-based QA over medical discussion threads. {intent} is filled with one of 5 perspective labels (e.g., “Information”, “Suggestion”).

Evaluate each sentence based on how well it
→ expresses or addresses the given perspective in
→ this discussion. Provide a score for EACH
→ sentence. Each score should be a two decimal
→ float between 0 and 1.

For example, if a sentence directly addresses the
→ {intent} perspective, its score should be close
→ to 1 (greater than 0.8); if a sentence is
→ unrelated to the {intent} perspective or
→ addresses a different aspect, its score should
→ be close to 0 (less than 0.2).

IMPORTANT: Output must be valid JSON format with
→ sentence indices as keys.

Perspective: {intent}
Sentences to evaluate:
{sentences}

Scores (JSON format):

G.4 PhysioNet

PHI entity detection in de-identified clinical notes. No intent placeholder; PHI categories are enumerated directly in the prompt.

Evaluate each sentence for the presence of
→ Protected Health Information (PHI). PHI
→ includes: patient names, doctor names, dates
→ (birth, admission, discharge, specific times),
→ locations (cities, hospitals, addresses),
→ medical record numbers, phone numbers, and
→ ages. Provide a PHI score for EACH sentence.
→ Each score should be a two decimal float
→ between 0 and 1.

For example, if a sentence contains clear PHI such
→ as a person's name, specific date, or location,
→ its score should be close to 1 (greater than
→ 0.8); if a sentence contains only general
→ medical information with no personal
→ identifiers, its score should be close to 0
→ (less than 0.2).

IMPORTANT: Output must be valid JSON format with
→ sentence indices as keys.

Sentences to evaluate:
{sentences}

PHI scores (JSON format):

G.5 HotpotQA

Multi-hop QA supporting fact identification. {intent} is the per-sample question (each sample has a unique question, so no intent stratification is applied).

Evaluate each sentence based on how useful it is as
→ a supporting fact for answering the given
→ question. Provide a score for EACH sentence.
→ Each score should be a two decimal float
→ between 0 and 1.

For example, if a sentence provides information
→ needed to answer the question, its score should
→ be close to 1 (greater than 0.8); if a sentence
→ provides unrelated information that doesn't
→ help answer the question, its score should be
→ close to 0 (less than 0.2).

IMPORTANT: Output must be valid JSON format with
→ sentence indices as keys.

Question: {intent}
Sentences to evaluate:
{sentences}

Scores (JSON format):

G.6 Evidence Inference

Statistical evidence extraction from clinical trial reports. {intent} is one of 8 annotator perspective identifiers (User0–User7).

From {intent}'s perspective, evaluate each sentence
→ for its value as statistical evidence of the
→ treatment's effect on the outcome. Evidence
→ sentences typically contain: statistical
→ results (p-values, confidence intervals),
→ outcome measurements, comparative data between
→ treatment and control groups, hazard ratios, or
→ effect sizes. Provide an evidence score for
→ EACH sentence. Each score should be a two
→ decimal float between 0 and 1.

For example, if a sentence reports statistical
→ results or quantitative treatment outcomes, its
→ score should be close to 1 (greater than 0.8);
→ if a sentence contains only background
→ information, methodology description, or
→ non-quantitative text, its score should be
→ close to 0 (less than 0.2).

IMPORTANT: Output must be valid JSON with ALL
→ sentence indices as keys. Every sentence must
→ be scored and every index must appear.

Perspective: {intent}
Sentences to evaluate:
{sentences}

Evidence scores (JSON format):

G.7 ContractNLI

Legal NDA clause relevance scoring. {intent} is one of 17 natural language inference hypotheses (e.g., “Receiving Party shall not use any Confidential Information for any purpose other than the Authorized Purpose”).

Evaluate each contract clause (sentence or list
→ item) based on whether it provides evidence for
→ or against the given hypothesis about this NDA.
→ A clause is evidence if it directly supports,
→ contradicts, or qualifies the hypothesis.
→ Provide a relevance score for EACH clause. Each
→ score should be a two decimal float between 0
→ and 1.

For example, if a clause directly addresses the
→ hypothesis (e.g., explicitly states an
→ obligation or right that the hypothesis
→ describes), its score should be close to 1
→ (greater than 0.8); if a clause is about
→ unrelated obligations, definitions, or
→ boilerplate unrelated to the hypothesis, its
→ score should be close to 0 (less than 0.2).

IMPORTANT: Output must be valid JSON format with
→ sentence indices as keys.

Hypothesis: {intent}

Contract clauses to evaluate:
{sentences}

Relevance scores (JSON format):

Table 5: Main empirical results for MAP comparing all ICL selection strategies. **Bold** marks the best configuration per row. Avg Single is the mean MAP across the four strategies.

Dataset	ICL0	<i>anchor_dpp</i>	<i>bm25</i>	<i>pattern_dpp</i>	<i>random</i>	Avg Single	Ens4
ECTSum	0.350	0.437	0.368	0.426	0.339	0.393	0.516
Evidence Inf.	0.206	0.205	0.199	0.196	0.203	0.201	0.304
HotpotQA	0.749	0.749	0.730	0.739	0.741	0.740	0.839
PhysioNet	0.764	0.767	0.764	0.749	0.782	0.766	0.879
PUMA	0.648	0.777	0.737	0.741	0.751	0.752	0.814
SubSumE	0.289	0.356	0.364	0.354	0.340	0.354	0.464
ContractNLI	0.718	0.706	0.713	0.717	0.712	0.712	0.828

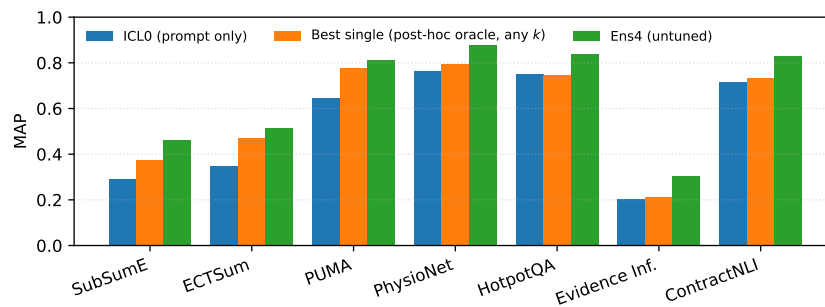


Figure 2: MAP comparison across seven datasets: Ens4 improves over both baselines on all 7 datasets. Numerical values are in Table 3.

H Main Empirical Results

Table 5 reports MAP for each of the four Ens4 ICL selection strategies and the Ens4 ensemble at $k \in \{1, 2, 3, 5, 8\}$, with $K=4$ fixed for Ens4 runs. Figure 2 visualizes the MAP comparisons from Table 3.

Computational cost. Ens4 issues $K=4$ scoring-LLM calls per document versus ICL0’s single call, independent of dataset, model, or document length. There is also an ICL-selection overhead, run once per document on the labeled pool, which is negligible compared to the LLM API call latencies. Each Ens4 call uses the same template prompt as ICL0 augmented with $k=2$ contrastive examples (Appendix E.1), so input tokens decompose as

$$\underbrace{T_{\text{system}} + T_{\text{hint}}}_{\text{shared with ICL0}} + \underbrace{2k \cdot T_{\text{ex}}}_{\text{ICL block}} + T_{\text{doc}},$$

where the ICL block is windowed to ± 2 sentences on long-document datasets (Appendix E.2) and output tokens are identical across ICL0 and every Ens4 sub-call. The input-token cost ratio

$$\begin{aligned} & \frac{4(T_{\text{shared}} + 2k \cdot T_{\text{ex}} + T_{\text{doc}})}{T_{\text{shared}} + T_{\text{doc}}} \\ &= 4 \left(1 + \frac{2k \cdot T_{\text{ex}}}{T_{\text{shared}} + T_{\text{doc}}} \right) \end{aligned}$$

approaches $4\times$ from above as T_{doc} grows, so the multiplier is closest to $4\times$ on long-document datasets.

Wall-clock latency is harder to isolate because it also depends on provider-side effects such as throttling and retries. We benchmark with local caching disabled so that every request goes to the LLM’s API, using 150 samples per configuration, identical hardware, and serial execution, even though

Ens4’s calls are embarrassingly parallel. As expected, Ens4 is more expensive than ICL0, but the resulting measurements in Table 6 should be interpreted as directionally indicative rather than as intrinsic latency multipliers. In practice, the four Ens4 scoring calls are independent and can be parallelized.

Table 6: Latency for ICL0 and Ens4 with serial execution of LLM API calls. The multiplier is Ens4 relative to ICL0. Measurements include provider-side retries and should be interpreted as directionally indicative.

Dataset	ICL0 (s/sample)	Ens4 (s/sample)	Multiplier
ECTSum	1.64	8.03	$4.9\times$
Evidence Inf.	0.76	7.69	$10.1\times$
HotpotQA	0.99	7.61	$7.7\times$
PhysioNet	0.17	1.78	$10.7\times$
PUMA	1.83	4.16	$2.3\times$
SubSumE	2.51	45.19	$18.0\times$
ContractNLI	1.46	14.03	$9.6\times$

Table 7: MAP for all non-singleton subsets of the four ICL selection strategies at $k=2$; the oracle best single strategy ($K=1$) is shown as a reference. **Bold** marks the best subset per K . $\overline{\text{MAP}}$ is the mean across 7 datasets. Strategy abbreviations: a=anchor_dpp, p=pattern_dpp, b=bm25, r=random.

K	Subset	SubS	ECT	PUMA	Phys	HotQ	EviI	CNLI	$\overline{\text{MAP}}$
1	Best Single [†]	0.373	0.469	0.777	0.795	0.749	0.213	0.733	0.587
2	a+p	0.417	0.492	0.797	0.842	0.812	0.260	0.786	0.629
2	a+b	0.426	0.482	0.796	0.837	0.810	0.261	0.787	0.628
2	a+r	0.420	0.474	0.800	0.843	0.806	0.259	0.780	0.626
2	p+b	0.424	0.473	0.780	0.837	0.804	0.255	0.787	0.623
2	b+r	0.427	0.431	0.785	0.855	0.797	0.253	0.785	0.619
2	p+r	0.421	0.468	0.787	0.854	0.800	0.249	0.786	0.623
3	a+p+b	0.450	0.509	0.807	0.867	0.832	0.288	0.816	0.652
3	a+p+r	0.445	0.506	0.808	0.872	0.830	0.285	0.813	0.651
3	a+b+r	0.450	0.492	0.807	0.868	0.827	0.287	0.814	0.649
3	p+b+r	0.452	0.488	0.800	0.874	0.824	0.280	0.814	0.647
4	a+p+b+r (Ens4)	0.464	0.516	0.814	0.879	0.839	0.304	0.828	0.664

[†]Oracle post-hoc best single strategy per dataset (varies by dataset).

Table 8: MAP for each individual sub-strategy at $k=2$ and 8. Mean Δ is the average change from $k=2$ to $k=8$.

Dataset	anchor_dpp		bm25		pattern_dpp		random		Ens4
	$k=2$	$k=8$	$k=2$	$k=8$	$k=2$	$k=8$	$k=2$	$k=8$	
ECTSum	0.437	0.397	0.368	0.371	0.426	0.339	0.339	0.392	0.516
SubSumE	0.356	0.347	0.364	0.369	0.354	0.359	0.340	0.361	0.304
HotpotQA	0.749	0.742	0.730	0.719	0.739	0.722	0.741	0.719	0.839
PhysioNet	0.767	0.702	0.764	0.735	0.749	0.739	0.782	0.784	0.879
PUMA	0.777	0.737	0.737	0.731	0.741	0.744	0.751	0.731	0.814
Evidence Inf.	0.205	0.184	0.199	0.186	0.196	0.191	0.203	0.186	0.464
ContractNLI	0.706	0.697	0.713	0.711	0.717	0.703	0.712	0.696	0.828
Mean Δ	−0.027		−0.008		−0.018		+0.000		

I Robustness Evaluations

I.1 Ensemble Size (K) and Composition

The results in Section 5.3 show the main outcomes. Here, Table 7 gives the full breakdown of MAP for each non-singleton strategy combination over the seven datasets. Ens4 ranks first among all 11 subsets on every dataset. Mean MAP rises from 0.587 ($K=1$, oracle best single) to 0.625 ($K=2$) to 0.650 ($K=3$) to 0.664 ($K=4$), with strictly diminishing marginal gains, consistent with Proposition 1’s intuition. Figure 3 visualizes the per-dataset best MAP at each $K \in \{1, 2, 3, 4\}$.

I.2 Complementarity vs. ICL Quantity

The default Ens4 configuration uses $K \cdot k = 8$ ICL examples for each query x , spread out over 4 scoring functions. This can be directly compared to Best Single at $k=8$, using the entire ICL budget in a single retrieval signal, as we show in Table 8. Ens4 outperforms by +0.070 to +0.119 MAP on every dataset. In fact, increasing the number of ICL examples from $k = 2$ to 8 for a single strategy is not helpful.

I.3 Complementarity vs. Labeled Data.

Compared to ICL0, Ens4 uses additional labeled data as ICL examples. To assess whether the additional labeled examples alone explain Ens4’s gains, we train a logistic regression classifier using all-MiniLM-L6-v2 sentence and intent embeddings, the same embedding model used for DPP retrieval in Ens4. The classifier is trained on the entire ICL pool together with the 100 calibration samples. This gives an advantage to the classifier, since each scorer in the Ens4 ensemble sees only two examples, for a maximum of eight distinct ICL

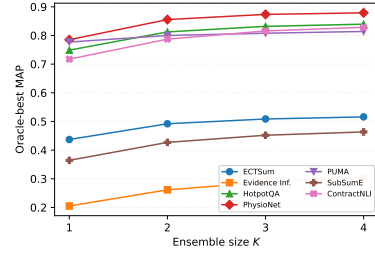


Figure 3: Best MAP at each ensemble size $K \in \{1, 2, 3, 4\}$ across all 7 datasets. All 7 datasets show monotonically increasing, flattening curves, consistent with Proposition 1’s intuition.

Table 9: MAP comparison between the supervised classifier baseline and Ens4. Bold indicates the better method.

Dataset	Classifier	Ens4
ECTSum	0.615	0.516
Evidence Inf.	0.281	0.304
HotpotQA	0.281	0.839
PhysioNet	0.732	0.879
PUMA	0.574	0.814
SubSumE	0.322	0.464
ContractNLI	0.293	0.828

examples used at a time.

Ens4 outperforms the supervised classifier on 6/7 datasets. The classifier performs best only on ECTSum. Results are shown in Table 9.

I.4 Complementarity vs. Stochasticity and Compute

Following from Section 5.3, we test whether stochastic diversity is as effective as retrieval diversity. Instead of ensembling four ICL selection strategies, we ensemble one with four different settings of temperature in the LLM scoring function. Table 10 shows the MAP for temperature ensembles for each of the four strategies on four different datasets. The default Ens4 outperforms in 12/16 cases with non-overlapping bootstrap 95% CIs (1000 resamples), while having overlap in the other 4 cases. In no case does temperature ensembling clearly outperform. Table 11 further shows individual strategy results for each T , showing minimal differences in MAP.

To more directly control for test-time compute, we also ensemble four ICL0 calls at temperatures $T \in \{0.3, 0.5, 0.8, 1.0\}$. This matches Ens4’s four LLM calls without introducing ICL examples or retrieval diversity. As shown in Table 12, Ens4 significantly outperforms on 6/7 datasets; ContractNLI is statistically tied.

Table 10: MAP values and 95% bootstrap CIs for temperature ensembles of single strategy scoring functions. The Ens4 column is reproduced from Table 3. The final column indicates whether the CIs overlap, or which method clearly outperforms

Dataset	Strategy	MAP	95% CI	Ens4	Overlap
PhysioNet	anchor_dpp	0.845	[0.8223, 0.8678]	0.879	No, Ens4 better
PhysioNet	pattern_dpp	0.842	[0.8216, 0.8621]	0.879	No, Ens4 better
PhysioNet	bm25	0.855	[0.8341, 0.8754]	0.879	No, Ens4 better
PhysioNet	random	0.882	[0.8625, 0.8997]	0.879	Yes
PUMA	anchor_dpp	0.817	[0.8109, 0.8228]	0.814	Yes
PUMA	pattern_dpp	0.783	[0.7762, 0.7898]	0.814	No, Ens4 better
PUMA	bm25	0.794	[0.7873, 0.8001]	0.814	No, Ens4 better
PUMA	random	0.804	[0.7982, 0.8108]	0.814	No, Ens4 better
SubSumE	anchor_dpp	0.435	[0.4229, 0.4473]	0.464	No, Ens4 better
SubSumE	pattern_dpp	0.438	[0.4251, 0.4505]	0.464	No, Ens4 better
SubSumE	bm25	0.450	[0.4370, 0.4624]	0.464	No, Ens4 better
SubSumE	random	0.441	[0.4283, 0.4541]	0.464	No, Ens4 better
ECTSum	anchor_dpp	0.518	[0.5065, 0.5293]	0.516	Yes
ECTSum	pattern_dpp	0.516	[0.5047, 0.5268]	0.516	Yes
ECTSum	bm25	0.478	[0.4667, 0.4901]	0.516	No, Ens4 better
ECTSum	random	0.447	[0.4367, 0.4582]	0.516	No, Ens4 better

Table 11: MAP of each single-strategy run by temperature T . **span** = max — min across T .

Dataset	ICL Strategy	$T=0.3$	$T=0.5$	$T=0.8$	$T=1.0$	span
PhysioNet	anchor_dpp	0.763	0.768	0.779	0.764	0.016
PhysioNet	pattern_dpp	0.743	0.735	0.742	0.752	0.017
PhysioNet	bm25	0.769	0.760	0.762	0.769	0.009
PhysioNet	random	0.798	0.792	0.789	0.798	0.009
PUMA	anchor_dpp	0.777	0.771	0.772	0.768	0.009
PUMA	pattern_dpp	0.739	0.739	0.740	0.739	0.001
PUMA	bm25	0.739	0.740	0.739	0.732	0.008
PUMA	random	0.758	0.757	0.756	0.752	0.006
SubSumE	anchor_dpp	0.339	0.346	0.350	0.352	0.013
SubSumE	pattern_dpp	0.344	0.356	0.358	0.351	0.014
SubSumE	bm25	0.359	0.361	0.366	0.374	0.015
SubSumE	random	0.343	0.355	0.359	0.362	0.019
ECTSum	anchor_dpp	0.449	0.443	0.432	0.423	0.026
ECTSum	pattern_dpp	0.442	0.444	0.426	0.425	0.019
ECTSum	bm25	0.372	0.370	0.364	0.368	0.008
ECTSum	random	0.321	0.335	0.340	0.344	0.023

Table 12: Equal-compute comparison between ICL0 and Ens4. ICL0 $\times$ 4 ensembles four calls at $T \in \{0.3, 0.5, 0.8, 1.0\}$. 95% CIs computed via paired bootstrap.

Dataset	ICL0	ICL0 $\times$ 4	Ens4	Ens4 — ICL0 [95% CI]
ECTSum	0.350	0.397	0.516	+0.120 [+0.109, +0.130]
Evidence Inf.	0.206	0.274	0.304	+0.019 [+0.015, +0.023]
HotpotQA	0.749	0.788	0.839	+0.047 [+0.043, +0.051]
PhysioNet	0.764	0.827	0.879	+0.034 [+0.018, +0.050]
PUMA	0.648	0.663	0.814	+0.144 [+0.138, +0.150]
SubSumE	0.289	0.370	0.464	+0.090 [+0.081, +0.098]
ContractNLI	0.718	0.818	0.828	+0.010 [−0.006, +0.008]

I.5 Cross-Model Validation

Section 5.3 introduced cross-model evidence as one of three robustness checks. We run the same untuned Ens4 default configuration on two open-weight LLMs, Llama3-8B and Qwen3-8B, across four representative datasets spanning medical, financial, and general-QA domains: PhysioNet, PUMA, HotpotQA, and ECTSum. We additionally evaluate GPT-5.6-terra on seeded 500-sample test subsets of all seven datasets. Table 13 reports the same four representative datasets for direct cross-model comparison.

I.6 Task-Hint Format Ablation

The Ens4 default configuration uses a one-line task hint (e.g., “*Task: Identify sentences relevant to the intent.*”) prepended to the prompt. The hint is not part of the ensemble framework—it is an input to every constituent scoring function—and is therefore optional, but its presence can help the LLM scorer by preparing it for ICL examples. This section tabulates the importance of the task hint across all 7 datasets and three hint regimes: *short* (the default one-line hint), *full* (a multi-sentence task description), and *no hint* (ICL examples only).

Across the 7 datasets we observe three tiers of sensitivity to the task hint (Table 14):

Non-semantic (PhysioNet): the Ens4 MAP collapses from 0.879 to 0.549 without the hint. Personal health information detection cannot be inferred from the contrastive ICL examples alone, because the positive class (personal health information spans) is defined by a regulatory criterion

rather than a semantic theme. The scoring LLM is not provided regulatory criteria as part of its context.

Domain-specific (Evidence Inference, ECTSum): each loses ~ 0.04 MAP without the hint, yet the no-hint Ens4 still beats *every* baseline on these datasets, so the hint is helpful but not critical for the Conformal Relevance framework.

Transparent (HotpotQA, PUMA, SubSumE, ContractNLI): the relevance criterion is recoverable from ICL examples alone, and the hint’s MAP impact is ≤ 0.01 in either direction. Expanding the hint to a multi-sentence task description (the *full* column) does not consistently help: it improves the MAP on 2/7 datasets and slightly hurts it on 5/7.

I.7 Stratified ICL Selection Ablation

For multi-intent datasets, ICL examples can be drawn from the same intent as the query document x (*stratified*) or from the full pool regardless of intent (*non-stratified*). Table 15 shows that stratification helps on three of four multi-intent datasets (+0.018 to +0.028 MAP). The exception is Evidence Inference, where non-stratified selection is marginally better (-0.006), possibly because its eight user-defined intents are weakly correlated with relevance patterns.

I.8 Windowed ICL Selection Ablation

Table 16 compares MAP with and without windowed ICL on SubSumE across three ICL budgets. Windowed ICL consistently improves MAP; the gain is largest where full-document ICL examples overflow the context window and degrade coherence. At $k = 2$ the improvement is +0.10 MAP; at $k = 5$ it is +0.16 MAP.

The key design insight is to decouple signal density—which pool examples are selected—from context compression—how they are rendered in the prompt. Windowing compresses rendering without altering selection, so all retrieval strategies benefit.

Table 13: Cross-model validation. Default Ens4 configuration run on Llama3-8B, Qwen3-8B, and GPT-5.6-terra. GPT-5.6-terra results use seeded 500-sample test subsets; the other models use the original test sets. The Gemini-2.5-Flash-Lite rows reproduce entries of Table 3 for reference. *Best Single* is the post-hoc best individual ICL strategy.

Model	Dataset	ICL0	Best Single (Strategy)	Ens4	$\Delta/\text{ICL0}$	Δ/Best
Gemini-2.5-Flash-Lite	PhysioNet	0.764	0.795 (random)	0.879	+0.115	+0.080
Gemini-2.5-Flash-Lite	PUMA	0.648	0.777 (anchor_dpp)	0.814	+0.165	+0.036
Gemini-2.5-Flash-Lite	ECTSum	0.350	0.469 (anchor_dpp)	0.516	+0.166	+0.047
Gemini-2.5-Flash-Lite	HotpotQA	0.749	0.749 (anchor_dpp)	0.839	+0.090	+0.090
Llama3-8B	PhysioNet	0.488	0.405 (random)	0.524	+0.036	+0.119
Llama3-8B	PUMA	0.555	0.603 (random)	0.663	+0.109	+0.061
Llama3-8B	ECTSum	0.220	0.178 (anchor_dpp)	0.258	+0.038	+0.080
Llama3-8B	HotpotQA	0.372	0.235 (pattern_dpp)	0.370	−0.002	+0.135
Qwen3-8B	PhysioNet	0.437	0.455 (random)	0.600	+0.163	+0.145
Qwen3-8B	PUMA	0.490	0.530 (anchor_dpp)	0.597	+0.106	+0.066
Qwen3-8B	ECTSum	0.225	0.188 (anchor_dpp)	0.258	+0.033	+0.070
Qwen3-8B	HotpotQA	0.427	0.323 (anchor_dpp)	0.471	+0.043	+0.148
GPT-5.6-terra	PhysioNet	0.843	0.881	0.921	+0.078	+0.040
GPT-5.6-terra	PUMA	0.712	0.838	0.856	+0.143	+0.018
GPT-5.6-terra	ECTSum	0.478	0.591	0.612	+0.134	+0.021
GPT-5.6-terra	HotpotQA	0.910	0.905	0.923	+0.014	+0.018

Table 14: Task-hint ablation. **Bold** marks the best Ens4 variant per dataset. ICL0 is shown for reference (uses the same short hint).

Dataset	ICL0	Ens4 (short)	Ens4 (full)	Ens4 (no hint)
ECTSum	0.350	0.516	0.500	0.476
Evidence Inf.	0.206	0.304	0.316	0.266
HotpotQA	0.749	0.839	0.832	0.841
PhysioNet	0.764	0.879	0.872	0.549
PUMA	0.648	0.814	0.806	0.804
SubSumE	0.289	0.464	0.460	0.457
ContractNLI	0.718	0.828	0.829	0.821

Table 15: Effect of stratified ICL selection on multi-intent datasets. **Bold** marks the better configuration.

Dataset	Intents	Stratified	Non-stratified	Δ
PUMA	5	0.814	0.787	+0.028
SubSumE	10	0.464	0.444	+0.020
ContractNLI	17	0.828	0.811	+0.018
Evidence Inf.	8	0.304	0.310	−0.006

Table 16: MAP for Windowed vs. full ICL on SubSumE (random strategy, Gemini-2.5-Flash-Lite). Windowing is beneficial at all ICL budgets; the gap grows with k as full ICL examples become longer.

ICL budget k	Full ICL	Windowed ICL
2	0.206	0.309
3	0.174	0.294
5	0.112	0.267

J Empirical Validation of Ensemble Theory

This appendix expands on the per-sample validation summarized in Section 5.3 by checking the condition from Lemma 1 with bootstrap confidence intervals. We restrict to test samples with more than one positive span.

Lemma 1 condition. Lemma 1 states that $\text{Comp}(R_j, R_k) > S_{\max} - S_{\min} \iff S^{(2)} > S_{\max}$. Table 17 reports the co-occurrence of these two events per dataset with 95% bootstrap CIs, using a variant scoring function with $K = 2$, as required by the theory. Although the agreement level is high, it is not exactly 100%. We observed that all 1,648 disagreements across all datasets together can be attributed to floating-point ties where $|S^{(2)} - S_{\max}| < 10^{-12}$ or $\text{Comp} - (S_{\max} - S_{\min}) < 10^{-12}$, which often occurs when $S_{\max} - S_{\min}$ is exactly zero.

Table 17: Co-occurrence of $\text{Comp}(R_j, R_k) > S_{\max} - S_{\min}$ and $S^{(2)} > S_{\max}$. 95% CIs are bootstrapped with 1000 resamples.

Dataset	n	Agreement (95% CI)
ECTSum	11,514	0.995 [0.994, 0.997]
Evidence Inf.	35,580	0.998 [0.998, 0.999]
HotpotQA	59,922	0.999 [0.998, 0.999]
PhysioNet	1,482	0.999 [0.998, 1.000]
PUMA	27,354	0.994 [0.993, 0.995]
SubSumE	10,812	0.995 [0.994, 0.996]
ContractNLI	18,252	0.999 [0.999, 1.000]
Pooled	471,378	0.9955

K Conformal Conciseness Across α

Figure 1b showed the conciseness metric for a range of recall levels β , and a fixed target coverage $1 - \alpha = 0.8$. Here we expand those results to additional values of α . Table 18 reports the default Ens4 configuration at $\alpha \in \{0.05, 0.10, 0.20\}$ with $\beta = 0.8$ fixed, while Figure 4 shows sweeps over β for the smaller α values. Ens4 improves conciseness over ICL0 on all seven datasets at $\alpha = 0.20$. and on most datasets at $\alpha = 0.05$ and $\alpha = 0.10$.

The biggest outlier is Evidence Inference at $\alpha = 0.05$. Evidence Inference’s relevance criterion — whether a stated outcome agrees with experimental evidence — is non-semantic. We showed in the task hint ablation from App. I.6, Table 14, that Evidence Inference is one of two non-semantic datasets that benefits from a lengthier task hint describing what relevance means in that context. Figure 5 shows that including a more descriptive hint (Ens4-FullTask) recovers a positive conciseness gap across all β .

K.1 Alternative Conformal Ensemble Rules

We compare score-level mean ensembling with alternative conformal ensemble rules while keeping the four Ens4 constituent scoring functions fixed. The rules we test are:

- **Majority Vote** (Gasparin and Ramdas, 2024b)
This method uses each scoring function to generate a conformal set, and then merges those sets keeping only elements that appear in the majority. If each individual scorer’s set guarantees $1 - \alpha$ coverage, the majority vote set only guarantees $1 - 2\alpha$ coverage, which can lead to undercoverage.
- **COncidence-Level Allocation** (Xu et al.,

Table 18: Mean conciseness (fraction of irrelevant sentences excluded from the prediction set) for ICL0 and Ens4 across $\alpha \in \{0.05, 0.10, 0.20\}$ at $\beta = 0.8$.

Dataset	$\alpha = 0.05$		$\alpha = 0.10$		$\alpha = 0.20$	
	ICL0	Ens4	ICL0	Ens4	ICL0	Ens4
ContractNLI	0.097	0.335	0.204	0.606	0.849	0.940
ECTSum	0.046	0.075	0.156	0.153	0.272	0.355
Evidence Inf.	0.305	0.168	0.542	0.611	0.755	0.803
HotpotQA	0.268	0.657	0.368	0.760	0.724	0.868
PhysioNet	0.152	0.616	0.426	0.716	0.717	0.820
PUMA	0.042	0.352	0.121	0.451	0.300	0.515
SubSumE	0.052	0.445	0.122	0.536	0.228	0.768

2025) This method uses each scoring function to generate a conformal set, and then takes the intersection of those sets. Using the same $1 - \alpha$ target coverage for each scorer in this way would lead to severe undercoverage. Instead COLA defines a different α_j for each scorer by optimizing them jointly against a set-size metric.

- **Union of Sets** (Yang and Kuchibhotla, 2025) This simple method generates a conformal set with each scoring function, and returns the union. The union guarantees $1 - \alpha$ coverage, but is typically very conservative and overcovers.
- **Average Single Scorer** This baseline runs each scoring function independently through the standard conformal pipeline, with its own calibrated threshold and prediction sets. We report the average coverage deviation and conciseness across the four scorers and seven datasets. No ensembling occurs; each scorer is evaluated independently, and only the resulting evaluation metrics are averaged.

Following Section 5.4, we repeat evaluation over 400 random calibration/test splits, with $\beta=0.8$ and $\alpha \in \{0.05, 0.10, 0.20\}$. Table 19 reports coverage deviation from the target, averaged across all seven datasets, and conciseness at $(\alpha, \beta) = (0.2, 0.8)$.

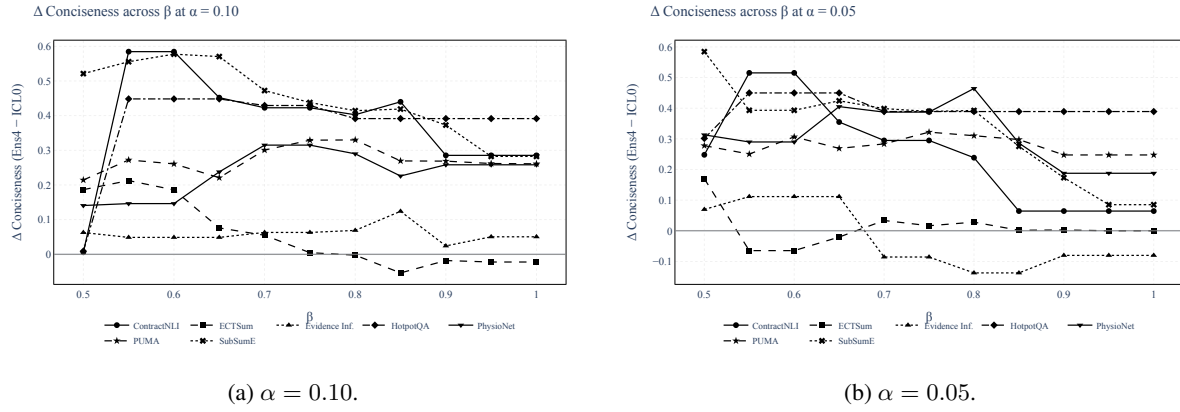


Figure 4: Conciseness improvement $\text{Ens4} - \text{ICL0}$ over β at stricter coverage targets, complementing Figure 1b.

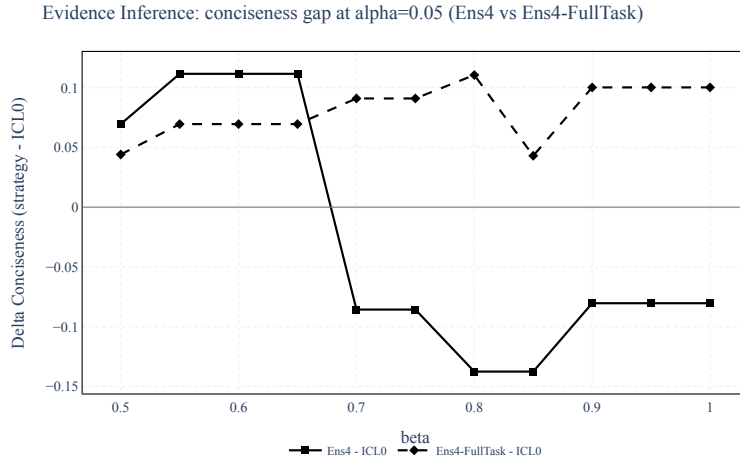


Figure 5: Conciseness improvement $\text{Ens4} - \text{ICL0}$ at $\alpha = 0.05$ on Evidence Inference. This dataset benefits from a lengthier manually crafted prompt describing the task, here shown as Ens4-FullTask (dashed). This variant recovers a positive gap across all β .

Table 19: Comparison of conformal ensemble rules using the same four Ens4 constituent scoring functions. Coverage deviation is empirical minus target coverage, averaged across seven datasets (percentage points; positive values indicate over-coverage). Conciseness is the fraction of sentences removed at $(\alpha, \beta) = (0.2, 0.8)$; higher is better.

Method	Coverage deviation (pp)			Conciseness
	$\alpha=0.05$	$\alpha=0.10$	$\alpha=0.20$	
Majority vote	+1.3	+0.5	-1.4	0.664
Confidence-Level Allocation	+1.7	+3.6	+5.5	0.426
Union of sets	+4.8	+9.0	+15.1	0.364
Average single scorer	+0.2	+0.2	+0.3	0.586
Mean Ensembling (Ens4)	+0.0	+0.1	+0.2	0.711