Modeling Perspectives in NLP: Parameter-Efficient Perspective Conditioning for Span Extraction and Summarization

Harikrishnan Gurushankar Saisudha, Sabine Bergler
Concordia University,
Montreal, Quebec, Canada,
h_gurush@live.concordia.ca, sabine.bergler@concordia.ca

Abstract

Understanding text through multiple perspectives is essential in healthcare community question answering, where answers frequently contain heterogeneous viewpoints, including experiences, suggestions, causes, follow-up questions, and informational claims. We present a unified perspective-conditioned framework for both span identification and perspective-aware summarization on the PerAnsSumm dataset. We approach explicit perspective samples in transformer models using two parameter-efficient mechanisms: prefix-conditioned representations and perspective-aware attention layers. First, we use multi-label perspective classification to identify relevant viewpoints, which serve as conditioning signals for downstream tasks. Span identification for perspective-specific extraction is modeled as a conditioned binary sequence labeling problem. Summarization, finally, is guided by perspective-enriched encoder representations. Experiments demonstrate that explicit perspective conditioning substantially improves span detection performance while achieving competitive summarization quality. Notably, perspective-aware attention achieves strong results using only a small fraction of the trainable parameters required by full fine-tuning. Our findings highlight the importance of structured viewpoint modeling and show that explicit perspective control enables efficient and interpretable multi-perspective text understanding.

Keywords: attention, summarization, span identification, BIO tags, perspective conditioning, multi-label classification, parameter-efficient approach, prefix tuning

1. Introduction

Text carries multiple viewpoints, such as personal experiences or factual claims (Cabitza et al., 2023). Understanding text through multiple viewpoints or perspectives is an essential yet underexplored challenge in NLP. Modeling these perspectives explicitly is crucial for tasks that require nuanced comprehension and generation (Frenda et al., 2025).

We address this challenge in the context of the PerAnsSumm shared task (Agarwal et al., 2025), which operates on the PUMA dataset (Naik et al., 2024), a corpus of healthcare community question-answering (CQA) threads annotated with five perspective types: cause, suggestion, experience, question, and information. This mirrors recent interest in Data Perspectivism (Cabitza et al., 2023), which explores here giving different answers to the same question, depending on a set of different perspectives that are defined by samples in the training data. Perspectives are pertinent in healthcare CQA, where subjects answer from fundamentally different epistemic positions — medical professionals, patients, and caregivers each bring distinct knowledge and lived experience to the same question. The five perspective categories in PUMA are not arbitrary topical bins but operationalization of different human standpoints, making perspective-aware summarization and span identification an instance of the perspectivist paradigm. The task has two subtasks: *Task A: perspective span identification*, where perspective-indicating text spans must be detected and labeled, and *Task B: perspective summarization*, where summaries must be generated conditioned on each perspective type (see Figure 1 for sample input-output patterns).

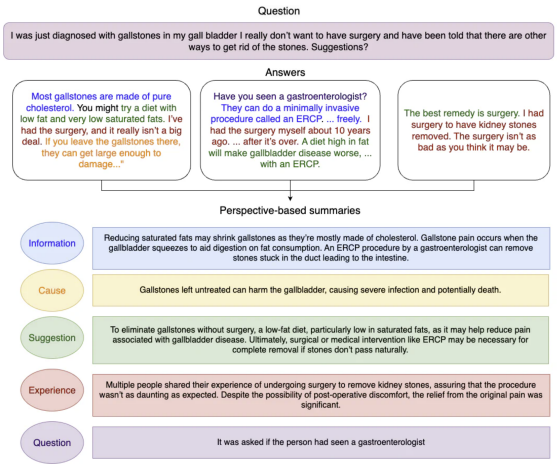

Figure 1: Image from PerAnsSumm (Agarwal et al., 2025) Task.[2] Task A: Span Identification and Classification (color-highlighted spans in answers); Task B: Summary Generation (Perspective-based summaries).

Since both tasks are inherently perspective-conditioned, they require models that are explicitly aware of these perspectives. We propose a unified

[2] https://peranssumm.github.io/docs/

Proceedings of NLPerspectives @ LREC 2026, pages 124–135
12 May 2026. ©ELRA Language Resources Association (ELRA), 2026

framework centered on *perspective conditioning*. We first introduce a multi-label perspective classifier using BART (Lewis et al., 2020), whose outputs form perspective signals for downstream models. These signals are then used in two alternate architectures: one using perspective-conditioned prefix vectors inspired by PLASMA (Naik et al., 2024), the other using a perspective-aware attention layer (inspired by AWAN (Tahaei and Bergler, 2025) and label attention (Vu et al., 2020)) that encodes the active perspective into token representations. Both mechanisms allow the model to selectively attend to and generate content that is grounded in a specific viewpoint, rather than aggregating indiscriminately across perspectives. This paper explores how conditional representations can improve both span-level identification and abstractive summarization in multi-perspective settings.

2. Prior Work

Perspective summarization for healthcare CQA settings was formally introduced by (Naik et al., 2024), who proposed the PUMA dataset, a collection of 3167 CQA threads from Yahoo! Answers annotated with five perspective types: cause, suggestion, experience, question, and information. They also proposed PLASMA, a prompt-driven controllable summarization model built on Flan-T5 with a prefix tuner and an energy-controlled perspective loss that enforces perspective-specific attributes in the generated summary. PLASMA outperformed five baselines across ROUGE, METEOR, BERTScore, and BLEU metrics, establishing a strong benchmark for the task on the PUMA dataset.

The PerAnsSumm 2025 Shared Task (Agarwal et al., 2025) had two subtasks: perspective span identification and classification (Task A), and perspective-based answer summarization (Task B), both evaluated on the PUMA dataset supplemented with a new test set of 50 samples. Large Language Models dominated the competition; 18 of 23 teams used LLMs in some capacity, including all top-10 teams. The top-performing system, WisPerMed (Pakull et al., 2025), achieved high performance with DeepSeek-R1 for span extraction and instruction-tuned Mistral-7B for summarization, while YALENLP (Jang et al., 2025) leveraged GPT-4o in a zero-shot setup, achieving the best scores on both summarization and span recognition. In general, there was a shift from fine-tuning paradigms toward in-context learning and prompt-based inference.

A different approach to multi-perspective answer summarization in CQA forums is *AnswerSumm* (Fabbri et al., 2021), which used an automated pipeline for creating bullet-point abstractive summaries by clustering relevant answer sentences and using cluster centroids as summary targets. To improve coverage and faithfulness, they proposed a multi-reward reinforcement learning objective combining ROUGE (Lin, 2004), NLI-based entailment (Bowman et al., 2015), and semantic area rewards, alongside a sentence-relevance prediction auxiliary loss. Their analysis showed that supervision from multi-perspective data inherently leads models to generate diverse, multi-viewpoint summaries, and that the quality of the NLI model significantly affects downstream performance.

3. Data and Tasks

We use the PerAnsSumm (Perspective-aware Healthcare Answer Summarization) dataset (Agarwal et al., 2025). The dataset consists of healthcare community question answering (CQA) threads, where each instance contains a question Q, a set of answers $\mathcal{A}$, and annotated perspective information.

The predefined set of perspective categories is:

$\mathcal{P} = \{$ *cause, suggestion, experience, question, information* $\}$

The shared task consists of two complementary objectives: (i) identifying and classifying perspective-specific spans in answer texts (Task A), and (ii) generating perspective-specific summaries (Task B) (see Figure 1).

In addition to these tasks, we introduce a *multi-label perspective classification task* as a preliminary step for both Task A and Task B[3]. This task involves classifying each question–answer pair given in the input into one or more perspective categories. The classification of question–answer pairs into perspective categories in a multi-label setting acts as a first step for perspective span recognition and perspective summarization. The quality of the classification models directly influences the performance of the downstream tasks, as the predicted perspectives serve as conditioning signals for both span identification and summarization.

The objective of Task A is to identify spans in the answer text that reflect a particular perspective and classify each span into the corresponding perspective category. These perspective spans represent fine-grained semantic units that characterize how different viewpoints are expressed within answer texts.

The objective of Task B is to generate a concise summary of a question–answer thread that reflects a specific target perspective $p \in \mathcal{P}$. Given a question Q, its associated set of candidate answers $\mathcal{A}$, and a target perspective p, the model is required to generate a summary Y_p that captures only the information relevant to perspective p from $\mathcal{A}$.

[3]We are using the PerAnsSumm 2025 data and evaluation but did not participate in the competition.

The dataset contains 2,533 training instances, 959 validation instances, a test-seen split of 640 instances (a subset of the validation set), and 50 instances in the official test set.

4. Preprocessing

We perform task-specific preprocessing for classification, span recognition, and summarization.

Multi-label Classification For multi-label perspective classification, we construct question–answer pairs (q, a) consisting of a question a and one answer q from the corresponding answer set $\mathcal{A}$ belonging to the CQA thread. Using the span label annotation in the training data, we label each answer with the set of perspectives present in it. An answer a is assigned a perspective p_i if at least one annotated span in that answer corresponds to perspective category p_i. This results in a dataset of question–answer pairs with multi-label perspective targets for classification.

Span Identification For span recognition, we construct perspective-conditioned instances for each perspective p_i identified in an answer. For every answer and its associated perspectives, we extract the spans labeled with that perspective along with their character-level offsets. These spans serve as target labels and are used only during training. Since the provided offsets in the dataset are defined over the raw JSON text, we realign the span offsets to match the processed answer text used during model training. The corrected character offsets are then converted into token-level BIO labels to formulate the task as a sequence tagging problem. For the baseline perspective span identification model, we adopt a joint tagging formulation where all perspective spans are predicted simultaneously. In this setup, we define separate B–I label pairs for each perspective category (e.g., B-info, I-info, B-suggestion, I-suggestion, etc.), while the O label remains shared across all perspectives. Unlike our perspective-conditioned models, which process one perspective at a time using question–answer–perspective triples, the baseline model operates on question–answer pairs without explicit conditioning. For each question–answer pair, we associate the full set of perspectives identified in that answer. The token-level labels, therefore, include spans from all perspectives within the same sequence, each annotated with its corresponding BIO tags. This formulation requires the model to jointly identify and distinguish spans belonging to multiple perspectives within a single tagging space.

Summarization For summarization, the model generates summaries across the entire set of answers for a given question, conditioned on a target perspective. The input is constructed by concatenating a short perspective-specific prompt, the question, and all associated answers into a single sequence. During training, the dataset is expanded such that each question–answer thread is paired separately with each of its perspective-specific gold summaries. This ensures that the model learns to generate one summary per perspective for each question–answer thread. The preprocessing does not change for the baseline summarization model.

5. Perspective Classification

5.1. MLC: BART Encoder-based Classification

This system is developed and fine-tuned for multi-label classification. We employ the encoder component only of the transformer-based BART (Lewis et al., 2020) as the backbone, because BART has the ability to process longer input sequences compared to architectures such as BERT and RoBERTa. The BART encoder can accommodate the larger context of the full question–answer pairs with longer answers.

The input question–answer pair is concatenated and fed into the BART encoder. For classification, we use the final hidden representation of the last token (EOS), which serves a role analogous to the [CLS] token in BERT-based models.

Since perspective identification is formulated as a multi-label classification task, a linear classification layer projects the encoder representation into a vector of dimension $|\mathcal{P}|$, corresponding to the number of perspective categories. A sigmoid activation function is applied to obtain independent probability scores for each perspective, and the model is trained using binary cross-entropy loss with class weights.

5.2. LLC: LLM-based Classification

In addition to the supervised classifier, we develop an LLM-based classification system for identifying perspective categories. This system serves as a robustness baseline by leveraging the few-shot prompting techniques with large language models to perform multi-label perspective identification without task-specific fine-tuning.

The LLM is prompted to assign one or more perspective categories to each question–answer pair. Comparing the supervised and LLM-based classification systems allows us to analyze their impact on downstream span identification and perspective-aware summarization on the test set. The full

prompt used for the LLM-based classifier is provided in Appendix 13.1

5.3. Performance Comparison

	CM-F1	CW-F1
MLC	71.26	79.91
LLC	72.47	82.17

Table 1: Perspective Classification results. Column header definitions: CM-F1: Classification Macro F1, CW-F1: Classification Weighted F1. Row definitions: MLC: BAT-based Multi-label Classification, LLC: LLM-based Classification

Table 1 compares the performance of the two classifiers. The LLM-based few-shot system outperforms our BART-derived system, and we use it for all other experiments exclusively.

6. Baselines

6.1. BLA: Perspective Span Detection

A BART encoder paired with a CRF layer serves as the baseline for the perspective span identification task. We fine-tune a BART encoder (Lewis et al., 2020) followed by a token-level classification layer and a Conditional Random Field (CRF) (Lafferty et al., 2001).

The CRF layer is applied to model dependencies between adjacent labels and enforce valid BIO tag transitions during decoding, which is commonly used in sequence labeling tasks (Huang et al., 2015).

This baseline model does not incorporate perspective-conditioned signals. Instead, it performs joint multi-perspective span identification using a unified tagging space. Specifically, the model predicts spans for all perspectives simultaneously within a single sequence tagging formulation.

6.2. BLB: Perspective Summarization

This system serves as the baseline for the perspective summarization task. We fine-tune a BART sequence-to-sequence model (Lewis et al., 2020) without incorporating any perspective-conditioning modules. The model operates in a standard encoder–decoder setting, where the input consists of the question and all associated answers concatenated into a single sequence.

The only explicit perspective signal is provided through a perspective-specific prompt that is prepended to the input text (Appendix 13.1.2). Apart from this prompt-based conditioning, no additional architectural modifications or perspective-aware mechanisms are introduced.

7. Span Identification

7.1. PTA: Prefix-Conditioned Span Identification

We implement a prefix-conditioned span identification model that extends the baseline BART+CRF architecture by introducing a prefix module, following the prefix-tuning paradigm (Li and Liang, 2021). For each target perspective, a prefix representation is generated using the prefix module containing a prefix encoder and a prefix MLP.

The prefix encoder maps a given perspective prompt into a fixed-dimensional embedding using a sentence transformer (Reimers and Gurevych, 2019) model. This 768-dimensional embedding is then transformed by a learnable prefix MLP into a sequence of k dense prefix vectors, where k denotes the prefix length. These k prefix vectors are prepended to the input token embeddings before being passed to the BART encoder.

Unlike standard prefix-tuning approaches (Naik et al., 2024; Li and Liang, 2021), where the backbone transformer model remains frozen, we jointly optimize both the prefix modules and the encoder parameters. Since span labels are defined only over the original input tokens, we discard the first k encoder representations corresponding to the prefix tokens and apply the classifier and CRF only to the remaining token representations.

We conduct an ablation study by varying the prefix length to analyze its impact on span extraction performance (see Table 2).

7.2. PAA: Perspective-aware Attention for Span Identification

In this alternate approach, we extend a baseline *BART+CRF* sequence labeling model by introducing a *perspective-aware attention layer* between the encoder and the token-level classifier. This layer explicitly conditions token representations on a target perspective by injecting a learned perspective embedding through cross-attention. Because each token attends to the perspective embedding, the resulting representations become perspective-dependent, encouraging the model to emphasize tokens that are most indicative of perspective-specific span prediction.

To preserve the pretrained knowledge of BART, the encoder parameters are kept frozen during training. Only the perspective-aware attention modules, the token-level classification layer, and the CRF decoding layer are trained. This design allows the model to learn perspective-specific span extraction while limiting the number of trainable parameters.

Given an input answer A_i, the encoder produces contextualized token representations:

$$H = \{h_1, h_2, \ldots, h_n\}, \qquad h_i \in \mathbb{R}^d$$

For a target perspective p, we obtain a perspective embedding $z_p \in \mathbb{R}^d$, which is learned as a trainable parameter.

We implement a cross-attention mechanism where the encoder outputs act as the Query (Q), while the perspective embedding provides the Key (K) and Value (V):

$$Q = HW_Q, \quad K = z_p W_K, \quad V = z_p W_V,$$

where W_Q, W_K, W_V are learnable projection matrices. The attention output is computed as:

$$\text{Attn}(H, z_p) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d}}\right)V.$$

This attention mechanism injects perspective-specific information into the token representations by allowing each token to attend to the target perspective embedding. The resulting representation is combined with the original encoder representations using a residual connection:

$$\tilde{H} = H + \text{Attn}(H, z_p).$$

To capture perspective-specific patterns, we train a separate attention module for each perspective. That is, for each $p \in \mathcal{P}$, a distinct attention layer with its own parameters is optimized independently. This formulation models span recognition as a perspective-conditioned binary sequence labeling task.

The enriched token representations $\tilde{H}$ are then passed through a linear classification layer to predict token-level BIO labels for span extraction.

We conduct an ablation study comparing a single key vector with multiple key vectors per perspective to evaluate how the number of perspective keys affects span extraction performance (see Table 2).

8. Perspective Summarization

8.1. PTB: Prefix-Conditioned Summarization

We implement a prefix-conditioned summarization model by extending the baseline BART encoder–decoder architecture with a prefix module. The prefix module follows the same design as in the prefix-conditioned span identification model, where a given perspective prompt is encoded and transformed into a sequence of k prefix vectors that are prepended to the encoder input embeddings. Each prefix vector is also transformed to the BART-large embedding dimension of 1024 in the prefix MLP layer.

Unlike the span identification setting, we do not discard the first k encoder representations, as summarization is a generation task. The decoder attends over the full set of encoder representations, including the prefix tokens, allowing the enriched perspective-conditioned signals to influence content selection during generation.

We conduct the same ablation study as performed for the span identification task (see Table 3).

8.2. PAB: Perspective-aware Attention for Summarization

We extend a baseline BART encoder-decoder summarization model by adding a perspective-aware attention layer between the encoder and the decoder. The approach is similar to perspective-aware attention for span recognition. The primary difference from the span recognition model lies in the backbone transformer and the role of the attention outputs in generation.

Given a question Q, its associated answers $\mathcal{A}$, and a target perspective p, the objective is to generate a summary Y_p that captures content aligned with the specified perspective. The input sequence is first encoded by the BART encoder to produce contextual representations

$$H = \{h_1, h_2, \ldots, h_n\}, \qquad h_i \in \mathbb{R}^d$$

As in the span recognition model, the perspective-aware attention module conditions these encoder representations on a learned perspective embedding z_p, producing enriched representations $\tilde{H}$. Unlike the span recognition setting, where these representations are used for token classification, the enriched encoder states $\tilde{H}$ are provided to the BART decoder to generate the summary autoregressively.

To preserve the pretrained knowledge of the summarization model, both the BART encoder and decoder are kept frozen during training. Only the perspective-aware attention modules are fine-tuned. As in the span recognition system, we train separate attention modules for each perspective, resulting in perspective-specific parameters that are optimized independently.

During inference, the multi-label classification system predicts the set of relevant perspectives P_i for a given input. For each predicted perspective $p \in P_i$, the corresponding attention module is activated to produce perspective-conditioned encoder representations $\tilde{H}$, from which the decoder generates a perspective-specific summary.

We further conduct an ablation study comparing the use of a single key vector with multiple key vectors per perspective to analyze how the number of perspective keys influences summarization performance (see Table 3).

9. Implementation

9.1. Training Setup

All models are implemented using the HuggingFace Transformers library and optimized using the AdamW optimizer (Loshchilov and Hutter, 2019). Parameters belonging to the BART backbone are trained with a learning rate of 5×10^{-5}, a commonly used setting for fine-tuning pretrained transformers (Lewis et al., 2020; Devlin et al., 2019). Unless otherwise specified, models are trained for 30 epochs.

Two learning rate scheduling strategies are used. Prefix-conditioned systems use ReduceLROnPlateau to adapt the learning rate when validation loss plateaus. Perspective-aware attention systems use a LambdaLR scheduler with a warmup followed by cosine decay, which gradually increases the learning rate early in training and then smoothly decays it, improving training stability (Vaswani et al., 2017; Loshchilov and Hutter, 2017).

Span identification models are trained using a joint objective combining Conditional Random Field (CRF) loss and token-level cross entropy (CE):

$$L = L_{CRF} + \lambda_{CE} L_{CE}$$

where $\lambda_{CE} = 0.7$. Weighted CE is used to address class imbalance in BIO labels with weights $[0.524, 32.61, 1.944]$ for O, B, and I classes. All span identification systems use the BART-base encoder (Lewis et al., 2020) with a maximum sequence length of 899 tokens. Summarization systems use BART-large-CNN with a maximum length of 512 tokens, except for prefix-based summarization which uses 899 tokens. Gradient clipping is applied for attention-based span identification and summarization models.

Prefix-Conditioned Span Identification The prefix encoder is implemented using a sentence-transformer (all-mpnet-base-v2 (Song et al., 2020)), with the prefix encoder and the first six layers of the BART encoder frozen to stabilize training. The prefix encoder uses a learning rate of 1×10^{-5} with weight decay 0.1, while the prefix projection MLP uses 3×10^{-4} with weight decay 1×10^{-4}. The CRF layer is trained with 1×10^{-2} and the classification layer with 2×10^{-4}.

Perspective-aware Attention for Span Identification The BART encoder remains frozen while only the attention and span prediction layers are trained. Due to slower convergence observed in validation, the model is trained for 50 epochs.

BART Encoder with CRF The baseline span identification system uses the BART-base model's encoder followed by a CRF layer for sequence labeling. Similar to other span systems, training uses the combined CRF and weighted CE loss.

Prefix-conditioned Summarization The prefix encoder uses a sentence-transformer backbone where the first six layers are frozen. Within the BART encoder-decoder backbone, most layers are frozen while the final two layers of both encoder and decoder are unfrozen for task adaptation. The prefix encoder uses a learning rate of 1×10^{-5} with weight decay 0.1, and the prefix MLP uses 3×10^{-4} with weight decay 1×10^{-4}.

Perspective-aware Attention for Summarization This system introduces a perspective-aware attention layer between the encoder and decoder of the BART-large-CNN model with a maximum input length of 512 tokens. The loss for the EOS token is down-weighted by 0.2 to reduce bias toward early sequence termination while still allowing the decoder to learn appropriate stopping behavior.

Baseline Summarization The baseline summarization system fine-tunes the standard BART-large-CNN encoder-decoder architecture for perspective summarization with a maximum token length of 512.

BART-based Multi-label Classification Perspective classification is performed using a BART-base encoder with a token length of 899 and a multi-label classifier applied to the EOS token representation. The model is trained using weighted binary cross entropy (BCE) loss to address class imbalance across perspective categories, with a learning rate of 1×10^{-5}.

LLM-based Classification We additionally evaluate a prompt-based classification system using the Qwen3-8B-AWQ (Yang et al., 2025) model. The model is hosted via vLLM[4] for faster and efficient inference. A few-shot prompting strategy is used to classify question-answer pairs into their respective perspective categories.

9.2. Evaluation and Outcomes

We follow the evaluation protocol defined in the PerAnsSumm shared task (Agarwal et al., 2025). Classification performance in Table 1 is measured using Macro F1 and Weighted F1 scores. Macro F1 treats all perspective classes equally, while Weighted F1 accounts for class imbalance by weighting each class according to its support.

[4] https://docs.vllm.ai/en/latest/

	K/PL	CM-F1	CW-F1	SM-F1	PM-F1
PAA	1	74.54	82.0	8.18	**61.96**
PAA	5	**75.92**	**82.65**	9.91	59.92
PAA	16	**75.34**	**82.43**	**10.62**	58.80
PTA	1	72.97	80.50	**10.89**	41.87
PTA	5	73.68	**82.47**	**11.00**	52.14
PTA	16	73.12	81.63	**11.06**	52.50
BL	-	50.94	62.62	4.73	28.99

Table 2: Perspective Span Classification and Identification results. Column header definitions: K/PL: Key/Prefix Length, CM-F1: Classification Macro F1, CW-F1: Classification Weighted F1, SM-F1: Strict Matching F1, PM-F1: Proportional Matching F1 (Only F1 scores, the precision and recall metrics are available in Table 4 in Appenedix 13.2) Table row definitions: PAA: Perspective-Attention, PTA: Prefix Tuning, BLA: Baseline (BART+CRF)

	K/PL	R-1	BS	MT	BU
PAB	1	24.97	79.32	18.93	4.23
PAB	5	34.20	**81.17**	26.85	8.24
PAB	16	37.47	**81.75**	**30.03**	9.68
PTB	1	36.41	**81.13**	**30.18**	**11.57**
PTB	5	**39.09**	**81.64**	**32.15**	**12.24**
PTB	16	36.15	**81.33**	**29.52**	10.58
BLB	-	**36.70**	**81.70**	29.02	8.97

Table 3: Perspective Summarization results. Column header definitions: K/PL: Key or Prefix Length, R-1: ROUGE-1, BS: BERTScore, MT: METEOR, BU: BLEU. Row definitions: PAB: Perspective-Attention, PTB: Prefix Tuning, BLB: Baseline BART encoder-decoder. ROUGE-L and 2 are available in Table 5 (Appendix 13.2)

Span recognition in Table 4 is evaluated using token-level F1 scores under two matching criteria: strict matching and proportional matching. Strict matching requires exact alignment between predicted and gold span boundaries, while proportional matching measures maximum token-level overlap between predicted and gold spans, allowing partial credit for boundary mismatches.

Summarization systems are evaluated only using ROUGE (Lin, 2004), BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), and BERTScore (Zhang et al., 2020) in Table 5. These scores evaluate the lexical and semantic metrics similarity between the reference and prediction summaries.

For evaluation, we use the test-seen data to evaluate locally, as the Codabench server[5] of the shared task does not work at times in the post-evaluation phase. We use LLM classification for all the systems, as we observed a slight improvement in performance from the BART-based Multi-label Classification (Table 1). The results are available in Tables 2, 3, 4, and 5.

Classification Task We first analyze the classification results derived from span predictions. A perspective is considered present if at least one non-'O' span exists. If the gold annotation contains no span for a perspective and the model predicts one, the prediction is counted as incorrect.

We observed that the BART-based classifier performs slightly worse than the LLM-based classifier on the test set (Table 1). Therefore, we use the LLM classifier for both the span identification and summarization systems to maintain consistency across tasks. However, the baseline span identification does not use the classification output, as they are trained to predict all perspective spans jointly.

Span Identification Table 4 shows that the baseline underperforms in almost all span metrics except Proportional Matching Precision (PM-P). While it achieves high PM-P (74.25), its Proportional Matching Recall (18.01) is very low, leading to poor overall PM-F1 (28.99). This suggests that the baseline predicts very few spans and does so conservatively.

Both PAA and PTA substantially improve strict and proportional F1 scores. Strict Matching F1 improves from 4.73 in the baseline to 10.62 for PAA ($K = 16$) and 11.06 for PTA ($PL = 16$). Proportional Matching F1 improves from 28.99 in the baseline to 61.96 for PAA ($K = 1$) and 52.50 for PTA ($PL = 16$).

A significant difference is observed in Proportional Matching Recall (PM-R) between PAA and PTA. PAA consistently achieves higher PM-R (above 50 across key sizes), whereas PTA shows substantially lower PM-R, particularly at smaller prefix lengths (e.g., 30.54 at $PL = 1$). This indicates that PAA is better at recovering overlapping spans with the gold annotations, while PTA tends to be more conservative in recall despite comparable strict F1 at higher prefix lengths.

PAA uses approximately 2.36M trainable parameters across key sizes ($K = 1$: 2.36M, $K = 5$: $\sim$2.36M, $K = 16$: 2.37M), compared to 139.42M in the fully fine-tuned baseline. Despite this large reduction in trainable parameters, PAA significantly outperforms the baseline across nearly all metrics. As the number of keys increases, Strict Matching F1 improves, while Proportional Matching F1 slightly decreases. Empirically, $K = 5$ provides the best trade-off between strict and proportional performance, indicating that too few keys underrepresent perspective cues, while too many may introduce redundancy given the dataset size.

Prefix tuning uses substantially more trainable parameters ($PL = 1$: 86.23M, $PL = 5$: 87.42M,

[5] https://www.codabench.org/competitions/4312/

$PL = 16$: 90.67M). As prefix length increases, Strict F1 slightly improves, and Proportional Matching Recall increases, indicating greater token overlap between predicted and gold spans. However, even at higher prefix lengths, PTA does not match the proportional PM-R levels achieved by PAA in Table 4.

The span systems were not explicitly trained with negative spans (i.e., cases where all tokens are labeled 'O' for a perspective). Despite this, both PAA and PTA remain robust as key or prefix length increases, as reflected in stable classification scores and improved recall. To further experimentally assess to what extent the perspectives are formed during training and to what extent the LLM can extrapolate to unseen perspectives, we created synthetic data introducing a new unseen perspective termed *reassurance*. We tested the trained prefix-based span system without retraining. Out of three synthetic examples, two showed reasonable overlap between predicted and synthetic gold spans. While this indicates partial generalization to unseen perspectives, performance was inconsistent. This may be due to the limited dataset size, imbalance between existing perspectives, and the fact that only five perspectives were used during training. To better demonstrate the capacity of the prefix MLP to encode new perspectives, experiments with a larger number of perspectives and more balanced data would be necessary.

Summarization Table 3 presents the results for perspective-aware summarization. In contrast to the span identification task, the differences between the baseline, PAB, and PTB systems are less pronounced. Most configurations perform very close to the fully fine-tuned baseline, with several slightly underperforming it.

The baseline achieves strong overall performance (R-1: 36.70, BERTScore: 81.70, METEOR: 29.02, BLEU: 8.97). PAB at $K = 16$ (R-1: 37.47, METEOR: 30.03) and PTB at $PL = 5$ (R-1: 39.09, METEOR: 32.15, BLEU: 12.24) slightly outperform the baseline on multiple metrics. However, the improvements are modest, and overall performance remains within a narrow range across systems.

What is particularly noteworthy is the parameter efficiency. The baseline uses approximately 406M trainable parameters, whereas PAB uses only about 4.2M parameters, which is roughly 1% of the baseline. Despite this drastic reduction in trainable parameters, PAB performs very close to the fully fine-tuned model. This highlights the effectiveness of attention-based perspective conditioning as a parameter-efficient fine-tuning strategy and emphasizes the importance of explicitly modeling perspectives in summarization.

We also observe that, unlike in span identification, PAB benefits from increasing the number of keys in summarization, with performance steadily improving from $K = 1$ to $K = 16$. This suggests that richer perspective embeddings help guide generation more effectively in a sequence-to-sequence setting. PTB exhibits a similar but less stable trend, with the best performance at $PL = 5$ and slight degradation at $PL = 16$, indicating that longer prefixes may introduce redundancy or noise during generation. PTB uses substantially more trainable parameters ($PL = 1$: 70.14M, $PL = 5$: 72.24M, $PL = 16$: 78.02M).

Despite incorporating a weighted EOS loss in the attention-based summarization system, we do not observe consistent improvements over PTB. This suggests that the choice of the EOS weighting hyperparameter λ_{eos} may require further tuning. Overall, the results indicate that perspective-aware modeling achieves competitive performance with significantly fewer trainable parameters, demonstrating strong parameter efficiency while maintaining summarization quality.

10. Conclusion

This work demonstrates that explicitly modeling perspective is not merely an auxiliary enhancement but a structural principle for multi-view text understanding. By separating perspective signals from surface lexical cues and injecting them directly into representation learning, our framework reframes span identification and summarization as conditioned reasoning tasks rather than generic text processing problems. The results show that perspective conditioning reshapes token-level and sequence-level representations in meaningful ways, enabling models to distinguish overlapping perspectives within the same discourse. Importantly, the effectiveness of lightweight attention modules suggests that perspective control does not require extensive parameter updates but instead benefits from targeted representational steering. This highlights that explicit perspective conditioning introduces modular and controllable structure into transformer models, enabling targeted analysis of perspective-specific behavior while maintaining parameter efficiency. Future work will explore perspective-specific adapter modules and supervised contrastive objectives to encourage stronger separation between perspective representations. We also plan to explore prefix-based conditioning strategies that better support generalization to previously unseen perspectives. Additionally, evaluating the model across domains will help determine whether it captures abstract viewpoint structures or relies on domain-specific patterns, thereby providing deeper insight into how perspective-sensitive knowledge is represented and transferred.

11. Acknowledgements

We thank Narjes Tahaei for her comments on this paper. The work was conducted with the support of a NSERC DG grant.

Parts of this paper were edited for clarity and fluency with the assistance of an AI language tool. All scientific content and conclusions remain the responsibility of the authors.

12. Bibliographical References

Siddhant Agarwal, Md. Shad Akhtar, and Shweta Yadav. 2025. Overview of the PerAnsSumm 2025 Shared Task on Perspective-aware Healthcare Answer Summarization. In *Proceedings of the Second Workshop on Patient-Oriented Language Processing (CL4Health)*, pages 445–455, Albuquerque, New Mexico. Association for Computational Linguistics.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proceedings of the ACL workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72.

Samuel Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A Large Annotated Corpus for Learning Natural Language Inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642.

Federico Cabitza, Andrea Campagner, and Valerio Basile. 2023. Toward a Perspectivist Turn in Ground Truthing for Predictive Computing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 6860–6868.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for Computational Linguistics: Human Language Technologies, volume 1 (long and short papers)*, pages 4171–4186.

Alexander R Fabbri, Xiaojian Wu, Srini Iyer, and Mona Diab. 2021. Multi-Perspective Abstractive Answer Summarization. *arXiv preprint arXiv:2104.08536*.

Simona Frenda, Gavin Abercrombie, Valerio Basile, Alessandro Pedrani, Raffaella Panizzon, Alessandra Teresa Cignarella, Cristina Marco, and Davide Bernardi. 2025. Perspectivist Approaches to Natural Language Processing: a survey. *Language Resources and Evaluation (LREC)*, 59(2):1719–1746.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *ArXiv*, abs/1508.01991.

Dongsuk Jang, Haoxin Li, and Arman Cohan. 2025. YaleNLP @ PerAnsSumm 2025: Multi-Perspective Integration via Mixture-of-Agents for Enhanced Healthcare QA Summarization. In *Proceedings of the Second Workshop on Patient-Oriented Language Processing (CL4Health)*, pages 415–427.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. *ICML*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising Sequence-to-Sequence pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th annual meeting of the Association for Computational Linguistics*, pages 7871–7880.

Xiang Lisa Li and Percy Liang. 2021. Prefix-Tuning: Optimizing Continuous Prompts for Generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597.

Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Proceedings of Text Summarization Branches Out Workshop at ACL 2004*, pages 74–81.

Ilya Loshchilov and Frank Hutter. 2017. Sgdr: Stochastic gradient descent with warm restarts. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*.

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, Bangkok, Thailand. Association for Computational Linguistics.

Tabea Pakull, Hendrik Damm, Henning Schäfer, Peter Horn, and Christoph Friedrich. 2025. WisPerMed @ PerAnsSumm 2025: Strong Reasoning Through Structured Prompting and Careful Answer Selection Enhances Perspective Extraction and Summarization of Healthcare Forum Threads. In *Proceedings of the Second Workshop on Patient-Oriented Language Processing (CL4Health)*, pages 359–373.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. MPNet: Masked and Permuted Pre-training for Language Understanding. *Advances in neural information processing systems (NEURIPS)*, pages 16857–16867.

Narjes Tahaei and Sabine Bergler. 2025. Beyond Consensus: Use of Demographics for Datasets that Reflect Annotator Disagreement. In *First Workshop on Bridging NLP and Public Opinion Research at COLM2025*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems (NeurIPS)*.

Thanh Vu, Dat Quoc Nguyen, and Anthony Nguyen. 2020. A label attention model for icd coding from clinical text. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 3335–3341. International Joint Conferences on Artificial Intelligence Organization. Main track.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. 2025. Qwen3 Technical Report. *arXiv preprint arXiv:2505.09388*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating Text Generation with BERT. In *International Conference on Learning Representations, ICLR 2020*.

13. Optional Supplementary Materials

13.1. Appendix A

13.1.1. Perspective Classification Prompt

```
You are a perspective
classification agent for a given
text.
You need to identify different
perspectives in the given text
and classify them into one of the
following categories:
1. EXPERIENCE
2. INFORMATION
3. CAUSE
4. SUGGESTION
5. QUESTION
Guidelines: - A perspective can
occur only once in a text.
- One text can have up to 5
unique perspectives.
- Do not repeat the same
perspective again.
Refer to the following examples
for guidance.
Example 1:
Question: what is parkinesonism?
Answer: u spelt it wrong
!!Parkinson's disease is one of
the most common neurologic
disorders of the elderly. The
term "parkinsonism" refers to any
condition that causes any
combination of the types of
movement abnormalities seen in
Parkinson's disease by damaging
or destroying dopamine neurons in
a certain area of the brain.
Output: ["INFORMATION"]
Example 2:
Question: I scream, shout and
swear in my sleep. How do I
stop?
Answer: I think that you have a
stress on your daily life. I
think that is not bad to do some
following things, If you caould
find any way, you will be happy,
otherwise, you have to go to a
psyc.
1) Keep out yourself from stress,
during your day.
2) Try to sleep, just when you
are really tired. (excuse me for
this example!) Like after a
sweet sex! try to find that are
you in this situation after sex,
or not? If No, it shows that you
try to sleep, before it needs.
3) Read Book, or newsletter
before sleep.
4) Drink one glass warm (NOT HOT)
milk.
5) Do some sort of excersice
before sleeping.
6) If you have any problem in
your dream, try to solve it by
someone that you are fighting
with. I mean, before your
sleeping (Specially when your
husband is not at home, because I
want to nobody awake you) try to
solve your problem with your
dream fighter!! Yes, it is funny
but true. Try to find a logical
way for treating out this
conflict with your dream.
i wish a good dream and sweet
night, beside of your sweet
husband.
Output: ["SUGGESTION", "CAUSE"]
Example 3:
Question: Are their any good
home remedies for tooth pain?
Answer: yes- keep water in your
mouth for 24 hours a day.
i drank about 5 each 16 oz
bottles for a few days (the
coolness of the water actually
moderated the pain). After a few
days, no pain. I had flushed it
clean and was able to function
until I could get to a dentist.
Output: ["EXPERIENCE"]
Example 4:
Question: Is 24 too old to
consider become pregant?
Answer: Only the 24 year old can
ask themselves that question --
are they personally ready?
Typically it's good to make sure
you're financially lined up to
have a baby (makes life easier),
and that your home environment
would be condusive to a baby
being around, but it's the
parents that need to know if
they're ready. I know 35+ year
olds that weren't ready yet.
Good luck! Output: ["QUES-
TION","SUGGESTION","EXPERIENCE"]
With the examples above, classify
the answer for the given question
into at least one of the given
categories. The answer should be
in the same format as the output
of the examples. There can be
more than one category for each
answer.
Think step-by-step before
deciding the output.
The text is: Question:
{question} Answer: {answer}
Output:
** Return only the output, do not
return anything else **
```

13.1.2. Summarization Prompt

```
For the {perspective}
perspective, summarize the given
answer for the question below:
Question: {question}
Answers: {answers}
```

13.2. Appendix B

Table 4 shows detailed span matching metrics.

	K/PL	SM-P	SM-R	PM-P	PM-R
PAA	1	9.37	7.25	**68.98**	**56.24**
PAA	5	10.76	9.18	67.02	54.19
PAA	16	**11.03**	10.23	67.09	52.33
PTA	1	10.80	10.98	66.58	30.54
PTA	5	10.49	11.55	62.39	44.78
PTA	16	10.58	**11.59**	63.27	44.86
BLA	-	8.96	3.22	74.25	18.01

Table 4: Perspective Span Classification and Identification results. Column header definitions: K/PL: Key/Prefix Length, SM-P: Strict Matching Precision, SM-R: Strict Matching Recall, PM-P: Proportional Matching Precision, PM-R: Proportional Matching Recall. Table row definitions: PAA: Perspective-Attention, PTA: Prefix Tuning, BLA: Baseline

Table 5 shows ROUGE-2 and ROUGE-L (Lin, 2004) metrics for Summarization.

	K/PL	ROUGE-2	ROUGE-L
PAB	1	9.61	22.0
PAB	5	15.55	30.72
PAB	16	**18.84**	**34.01**
PTB	1	17.33	32.39
PTB	5	**18.90**	**34.84**
PTB	16	**17.84**	32.29
BLB	-	**18.12**	32.28

Table 5: ROUGE-2 and ROUGE-L scores for summarization. Column header definitions: K/PL: Key/Prefix Length. Table row definitions: PAB: Perspective-Attention, PTB: Prefix Tuning, BLB: Baseline