YNU-HPCC at SemEval-2026 Task 5: Rating Plausibility of Word Senses in Ambiguous Stories through Narrative Understanding

Mingyu Bai, Jin Wang, and Xuejie Zhang
School of Information Science and Engineering
Yunnan University
Kunming, China
Contact: mybai@stu.ynu.edu.cn, {wangjin, xjzhang}@ynu.edu.cn

Abstract

This paper introduces our approach to SemEval 2026 Task 5, which evaluates the rationality of word-sense scores in ambiguous stories through narrative comprehension. This task requires models to assess the consistency between a given word-sense definition and the meaning of an ambiguous target word in a short narrative context, and to infer a rationality score on a 1-5 scale. We experimented and compared multiple methods. These methods include multi-head ensembles that simulate the behavior of individual annotators, ordinal classification and regression methods that treat scores as ordered categories, and direct regression using mean squared error (MSE) or L1 loss to predict human-average consensus scores. Additionally, we investigated instructional fine-tuning (Raganato et al., 2023) with low-rank adaptation (LoRA) on large language models (LLMs) such as Qwen3-4B-Instruct and Phi-4-mini. Our experimental results show that the direct MSE regression method performs best. This study indicates that directly optimizing to approach human consensus scores is effective for this task, while methods that model individual annotator differences are less applicable.

1 Introduction

Evaluating the Rationality of Word Sense in Ambiguous Stories through Narrative Understanding is crucial for advancing natural language understanding, as it goes beyond traditional word-sense disambiguation and requires determining the meaning of ambiguous words in context. Computers find it difficult to understand words with multiple meanings accurately (Navigli, 2009). Although LLMs perform well, they often fail on less frequent senses and show a bias toward more common meanings (Navigli, 2026). Especially in some short stories, these ambiguities may be caused not only by polysemy but also by the author's intentional puns or metaphors. This can help computers generate coherent stories, perform nuanced reading comprehension, and handle intentional ambiguities such as puns and metaphors, better understand the core of the story, and avoid falling into carefully designed textual traps.

Existing word sense disambiguation methods mainly employ supervised learning, including Naive Bayes, Support Vector Machine (SVM), Bi-directional Long Short-Term Memory (BiLSTM), and BERT, with a small portion using clustering and self-training (Raganato et al., 2017; Lee et al., 2004; Huang et al., 2019).

This paper proposes applying direct regression, fine-tuning a pre-trained language model to predict a continuous confidence score by minimizing the MSE between the predicted value and the average human annotator score for each sample. This approach prioritizes alignment with human consensus, directly penalizes large deviations, and leverages the model's existing language knowledge to capture the subtle relationship between narrative context and meaning confidence.

From a quantitative perspective, our regression system based on MSE achieved the best performance, with an accuracy within one standard deviation (Acc_wSD) of 0.72 and a Spearman correlation coefficient of 0.54, significantly outperforming our baseline and alternative structured methods, such as ordinal classification. This result indicates that directly regressing to human average judgments is an effective strategy for this task. However, our system tends to be conservative and avoids extreme ratings. In our local validation of 588 samples, the system gave a score of 1 to only 20 records and a score of 5 to 70 records, while the gold data had 121 and 125 records for these scores, respectively. Because human ratings in AmbiStory are naturally subjective and vary, the model prefers safe, middle-ground scores to minimize potential error.

The rest of the paper is organized as follows. Re-

Proceedings of the The 20th International Workshop on Semantic Evaluation (2026), pages 1163–1168

lated work is reviewed in Section 2. The proposed methods and the system architecture are introduced in Section 3. The experimental results and analysis are presented in Section 4. Finally, the conclusion and future work are provided in Section 5.

2 Background

The task of selecting a word's sense in a given context is defined as WSD. Gehring and Roth (2025) designed the AmbiStory dataset to evaluate word sense plausibility in short stories. It was found that most models correlate poorly with human judgments in complex narratives. Furthermore, Meconi et al. (2025) addressed the gap in understanding whether LLMs truly grasp word senses. Their study compared instruction-tuned LLMs with state-of-the-art specialized systems.

Pretrained Language Models (PLMs) are widely utilized to extract semantic features. Masethe et al. (2025) investigated hybrid Transformer-based LLMs for WSD in low-resource languages. Their work combined different architectures to improve disambiguation accuracy. However, models still face difficulties when stories contain indirect clues or distracting information.

The performance of LLMs is often improved through fine-tuning on specific datasets. Gehring and Roth (2025) demonstrated that fine-tuning on AmbiStory increases the agreement with human ratings. Meconi et al. (2025) also evaluated LLMs in generative settings, such as definition and example generation. Furthermore, the LoRA method is often used to update model parameters efficiently.

3 LoRA Fine-tuning for LLM Instructions

Four scoring methods are introduced and analyzed to train the DeBERTa-v3-large model. These models convert story text into scores. Additionally, the Qwen3-4B-Instruct model is fine-tuned using the LoRA method.

3.1 Multi-head ensemble

Utilizing the choices array within the dataset, each column is interpreted as representing an individual. A classification head is trained for each individual to simulate their unique scoring habits. During the inference phase, a soft voting mechanism is employed to average the final predictions for each person.

This method employs cross-entropy as the loss function, with the formula presented as follows:

$$L_k = -\frac{1}{N}\sum_{i=1}^{N} \log p_{y_i}^{(k)}(x_i) \tag{1}$$

where p_{y_i} represents the true distribution, which, for classification tasks, corresponds to the true classification label.

3.2 Ordered Classification

Unlike conventional classification, which ignores the order of categories or label scaling, ordinal regression models discrete ordered categories as continuous latent variables (Bellmann and Schwenker, 2020).

This method constructs an ordinal regression architecture based on the DeBERTa-v3-large pretrained model (see Figure 1). The model first encodes the text into a high-dimensional representation, and then predicts a continuous rating tendency value, denoted y, using a regression head. This y value is not the final score itself, but needs to be mapped through a set of ordered segmentation points. We set four segmentation points (corresponding to five rating levels), calculate the cumulative probability that y^* values are less than each segmentation point using the Sigmoid function, and then estimate the probability distribution for each specific level to which the sample belongs using the difference in probabilities. During training, we use the negative log-likelihood loss function to maximize the probability corresponding to the true rating level.

In the inference stage, the final integer prediction score is obtained by calculating the expected value of the scoring level.

This method employs negative log-likelihood (NLL) as the loss function, with the formula as follows:

$$L = -\frac{1}{N}\sum_{i=1}^{N} \log(p_{y_i}^{(i)}) \tag{2}$$

at this point, p_{y_i} differs from cross-entropy. It represents the negative logarithm of the model's probability of the true label, that is $-\log q(t_{true})$, where $q(y_{true})$ is the probability of the model predicting the true class.

3.3 Ordinal Regression

The K-1 threshold method is a core idea for solving ordered classification problems. Unlike directly

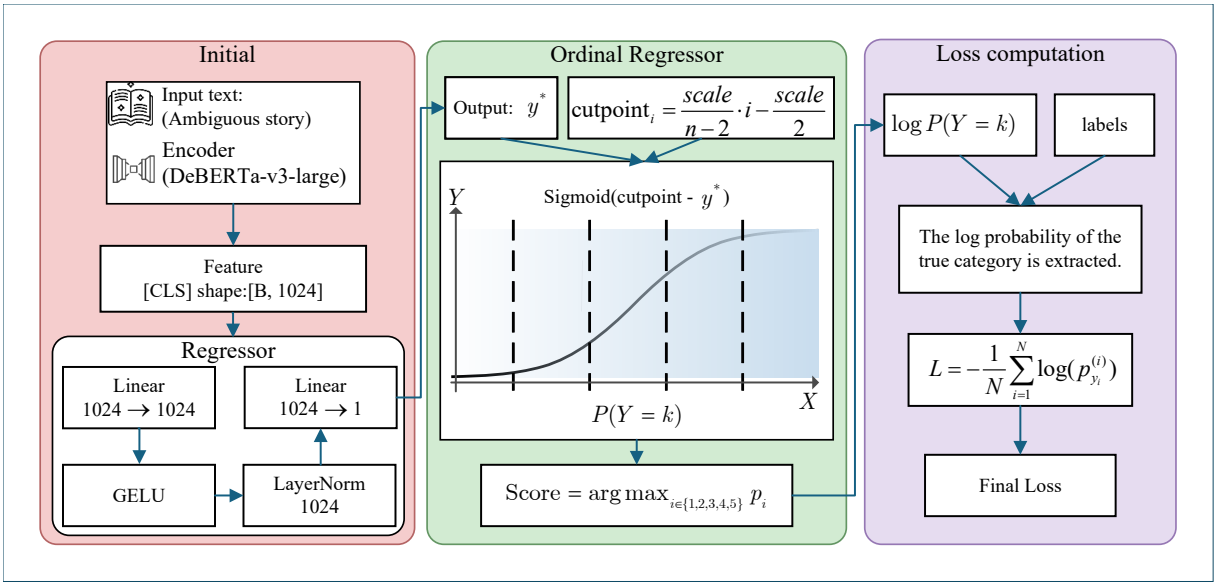

Figure 1: Procedure of ordered classification

asking the model to guess a label on a scale of 1-5, this algorithm reconstructs the problem as a sequentially progressive **climbing** decision-making process. The model answers four key questions in turn through four parallel binary classification heads: Is the score of this sample greater than 1?, Is it greater than 2?, and so on, up to Is it greater than 4? Each classification head specifically learns a decision threshold.

During training, the true labels are automatically converted into a **ramp target sequence** consisting of 0s and 1s (e.g., [1, 1, 0, 0], indicating greater than 1 and 2, but not greater than 3 and 4). The model is guided to learn this sequence through binary cross-entropy loss. During inference, the model accumulates the probability outputs from these four greater-than-or-equal-to conditions to determine the final grade. The core advantage of this method lies in explicitly encoding the sequential relationships between categories as structural knowledge for the model.

In the K-1 ordinal regression model, the loss function employs binary cross-entropy (BCE). The formula is as follows:

$$L_{BCE} = -\frac{1}{N}\sum_{j=1}^{N}\Big[t_j \log(\sigma(z_j)) + (1 - t_j)\log(1 - \sigma(z_j))\Big] \quad (3)$$

$t_j = [y > j]$ represents whether the true score is greater than the threshold, and z_j without Sigmoid represents the original output value of the j classification head.

3.4 MSE and L1

The MSE loss defines the model's optimization objective. MSE guides training by calculating the squared error between the predicted value and the true average score. It imposes severe penalties for large prediction deviations, thereby driving the model output to be as close as possible to the average annotator score. In contrast, L1 loss focuses more on capturing the overall trend rather than absolute accuracy. During inference, the model's continuous outputs are rounded and post-processed into integer scores ranging from 1 to 5.

Assuming the actual value is y, and the model's predicted value is $\hat{y}$, where i is the index. The formula is as follows:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad (4)$$

3.5 Qwen3 with LoRA Fine-tuning

The Unsloth framework is utilized to achieve underlying optimization and acceleration for instruction learning. Only a small number of new parameters are added to the frozen pre-trained weights. Sufficient expressive power is ensured by this parameter-efficient design. Efficient training of the 4-billion-parameter model is enabled on a single consumer-grade GPU.

3.6 Section Summary

An experimental framework is established by combining DeBERTa-v3-large and Large Language Models. The experimental details and results for each method are presented in the next section.

4 Experimental Result

4.1 Dataset

The task organizer provides a dedicated English dataset for training and evaluation, which contains multiple short story fragments. Each sample has multiple human labels, so the model can learn the consensus and also capture the disagreement.

The system input includes three key narrative sentences: a **preface**, a **sentence** (containing the target word), and a **conclusion**. Additionally, a **judgment-meaning** sentence is used to ask if the word has this potential meaning.

4.2 Evaluation Metrics

We use Acc_wSD as one of the main evaluation metrics. For a word sense prediction task involving n samples, let the human score for each sample i be $L_i = [L_{i,1}, L_{i,2}, ..., L_{i,k}]$ (Scores from k annotators). The predicted score of the model is p_i, the accuracy within standard deviation is defined as:

$$Acc\_wSD = \frac{1}{n}\sum_{i=1}^{n} I(within\_SD(p_i, L_i)) \quad (5)$$

where $I(\cdot)$ is an indicator function that returns 1 when the predicted value satisfies any of the following conditions, and returns 0 otherwise:

1. Standard deviation condition: The predicted value should fall within one standard deviation of the average human score.

$$\mu_i - \sigma_i < p_i < \mu_i + \sigma_i \quad (6)$$

2. Absolute difference condition: The absolute difference between the predicted value and the average human score is less than 1.

$$|p_i - \mu_i| < 1 \quad (7)$$

where $\mu_i = \frac{1}{k}\sum_{j=1}^{k} l_{ij}$ and σ_i are the mean and standard deviation of human scores, respectively.

Spearman's Rank Correlation Coefficient is another core evaluation metric. For n samples, let $g = [g_1, g_2, \ldots, g_n]$ denote the average rating sequence labeled by humans, and $p = [p_1, p_2, \ldots, p_n]$ represent the system's predicted rating sequence. The Spearman correlation coefficient is calculated using the following formula:

$$\rho = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{n(n^2-1)} \quad (8)$$

where d_i represents the difference between the ranks of the i-th pair of sentences, n is the number of pairs of sentences, and ρ is Spearman's rank correlation.

4.3 Implementation Details

The DeBERTaV2Tokenizer processes input texts. A structured sequence is constructed by joining the homonym, definition, example, and context with [SEP] tokens. All sequences are padded to a fixed maximum length of 512. An input_ids vector and an attention_mask are created for every sample. And the [CLS] token is selected to represent the meaning of the whole story.

Final scores are directly generated by two baseline methods:

1. The majority: All predictions achieved a score of 4.

2. Randomly: All predictions are generated randomly.

4.4 Parameters Fine-tuning.

4.4.1 DeBERTa-based Model Configurations

DeBERTa-based Model Configurations The DeBERTa-v3-large model is trained for 10 epochs to ensure stability. The best performance is reached with a total batch size of 48. Gradient explosion is easily caused by training the cutpoints in the ordered classification method. Therefore, a scale value of 6 is used to solve this problem. Four cutpoints are set to -3, -1, 1, and 3 to achieve the best separation.

4.4.2 Qwen3 Learning Configurations

The Qwen3-4B-Instruct models are fine-tuned using LoRA.Higher accuracy is obtained by increasing the rank (r) and alpha (α) values. This trend is clearly shown in the development set results. More training information is stored when these parameters are expanded. Therefore, the largest possible values are selected until the GPU memory limit is reached. Best performance is achieved by using these maximized settings to learn the narrative scores. The growth trend is shown in Figure 2.

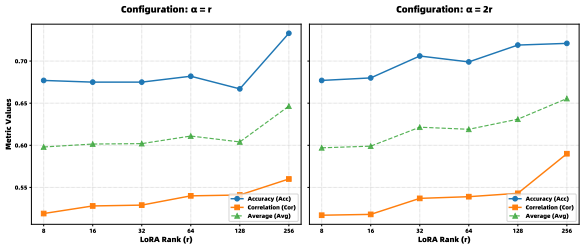

Figure 2: Impact of r and α on the performance

Method	Acc	Cor	Avg
Multi-head	0.45	0.02	0.24
Ordered Classification	0.60	0.28	0.44
Ordered Regression	0.63	0.54	0.59
MSE	0.72	0.54	0.63
L1	0.54	0.20	0.27
Baseline-Majority	0.57	0	0.29
Baseline-Random	0.45	-0.08	0.22

Table 1: Performance of DeBERTa-based models on the task

4.5 Comparative Results

The performance of all models is compared in this section. Detailed numerical results are presented in the figures below. For comparison with the baseline, the results reported in our paper are based on the official dev.json set.

The MSE regression method achieves the best performance among the four DeBERTa-based tasks. This approach achieves high accuracy (0.72) and a Spearman correlation of 0.54. The Multi-head ensemble obtains the lowest scores. Furthermore, the ordered classification and ordered regression methods yield moderate results. The experimental results are presented in Table 1.

Qwen3 and Phi4 perform poorly in the zero-shot setting. However, performance improves significantly after LoRA fine-tuning. A clear growth in accuracy is observed as the LoRA parameters are increased. The Qwen3-lora model reaches the same top accuracy as the best DeBERTa method. The experimental results are presented in Table 2.

4.6 Discussion

The low performance of the Multi-head ensemble is due to the failure to account for individual differences. Many diverse factors influence human subjectivity in ambiguous stories. Therefore, significant differences arise within the voting mechanism when each person's habits are imitated (Xue and Hauskrecht, 2019). This complexity makes it difficult for the model to find a clear consensus.

LLM	Acc	Cor	Avg
Qwen3-4B (zero-shot)	0.49	0.26	0.375
Phi4-mini (zero-shot)	0.47	0.28	0.375
Qwen3-4B-lora	0.71	0.54	0.625
Phi4-mini-lora	0.66	0.52	0.59
Baseline-Majority	0.57	0	0.29
Baseline-Random	0.45	-0.08	0.22

Table 2: Experimental results of instruction-tuned LLMs

The ordered classification and regression methods obtain moderate scores. These methods assume a natural order between categories. However, the meanings of ambiguous words often represent distinct and unrelated values. Therefore, noise is introduced by forcing these meanings into a fixed sequence. Furthermore, the Sigmoid function's smooth transitions reduce the ability to distinguish difficult samples.

In contrast, the best results are achieved by the direct MSE regression method. Large prediction deviations are heavily penalized by the MSE loss function. This approach allows the model to utilize pre-trained linguistic knowledge more effectively. Therefore, directly reaching human consensus is shown to be the most stable strategy for this task.

Fine-grained semantic distinctions are crucial for this task. The Qwen3-4B model with LoRA fine-tuning achieves performance comparable to the best DeBERTa method. However, the same approach yields lower performance on Phi4-mini. First, the smaller overall capacity of Phi4-mini may limit its ability. Second, the LoRA parameters may still be suboptimal for this specific task. Furthermore, the training process focuses on numerical scoring rather than deep narrative understanding.

5 Conclusion

This study explored several methods for evaluating the rationality of word meanings in ambiguous stories to address narrative comprehension in the presence of lexical ambiguity. The results indicate that the regression method based on MSE achieves the best overall performance, surpassing traditional baseline methods and more structured methods such as ordinal classification and regression. These results significantly outperform the majority baseline (0.57) and the random baseline

(0.45). Furthermore, the LoRA-enhanced Qwen3 model reaches the same top accuracy as the best DeBERTa method.

Future work will attempt to use neuro-symbolic methods (Dong and Sifa, 2024) or Graph Neural Networks (GNNs) (Caruso et al., 2026) to represent word meanings thereby reducing semantic uncertainty (Qiao et al., 2025).

We release our code at `https://github.com/white-Alopex-lagopus/SemEval_task5`

Acknowledgement

This work was supported by the National Natural Science Foundation of China (NSFC) under Grant Nos.61966038 and 62266051. The authors would like to thank the anonymous reviewers for their constructive comments.

References

Peter Bellmann and Friedhelm Schwenker. 2020. Ordinal classification: Working definition and detection of ordinal structures. *IEEE Access*, 8:164380–164391.

Alessandro Caruso, Jacopo Venturin, Lorenzo Giambagli, Edoardo Rolando, Zakariya El-Machachi, Frank Noé, and Cecilia Clementi. 2026. Extending the range of graph neural networks with global encodings. *Nature Communications*, 17(1):1855.

Tiansi Dong and Rafet Sifa. 2024. Word sense disambiguation as a game of neurosymbolic darts. In *Proceedings of the Workshop: Bridging Neurons and Symbols for Natural Language Processing and Knowledge Graphs Reasoning (NeusymBridge)@ LREC-COLING-2024*, pages 22–32.

Janosch Gehring and Michael Roth. 2025. Ambistory: A challenging dataset of lexically ambiguous short stories. In *Proceedings of the 14th Joint Conference on Lexical and Computational Semantics (*SEM 2025)*, pages 152–171. Association for Computational Linguistics.

Luyao Huang, Chi Sun, Xipeng Qiu, and Xuanjing Huang. 2019. Glossbert: Bert for word sense disambiguation with gloss knowledge. 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 3507–3512. Association for Computational Linguistics.

Yoong Keok Lee, Hwee Tou Ng, and Tee Kiah Chia. 2004. Supervised word sense disambiguation with support vector machines and . . . *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of T.*

Hlaudi Daniel Masethe, Mosima Anna Masethe, Sunday O. Ojo, Pius A. Owolawi, and Fausto Giunchiglia. 2025. Hybrid transformer-based large language models for word sense disambiguation in the low-resource sesotho sa leboa language. *Applied Sciences (Switzerland)*, 15.

Domenico Meconi, Simone Stirpe, Federico Martelli, Leonardo Lavalle, and Roberto Navigli. 2025. Do large language models understand word senses? In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 33885–33904. Association for Computational Linguistics.

Roberto Navigli. 2009. Word sense disambiguation: A survey. *ACM Computing Surveys*, 41.

Roberto Navigli. 2026. Is word sense disambiguation dead in the llm era? *Proceedings of the AAAI Conference on Artificial Intelligence*, 40:39753–39762.

Wenbo Qiao, Peng Zhang, and Qinghua Hu. 2025. Quantum visual word sense disambiguation: Unraveling ambiguities through quantum inference model. *arXiv preprint arXiv:2512.24687*.

Alessandro Raganato, Iacer Calixto, Asahi Ushio, Jose Camacho-Collados, and Mohammad Taher Pilehvar. 2023. Semeval-2023 task 1: Visual word sense disambiguation. In *Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 2227–2234. Association for Computational Linguistics.

Alessandro Raganato, Jose Camacho-Collados, and Roberto Navigli. 2017. Word sense disambiguation: A unified evaluation framework and empirical comparison. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 99–110, Valencia, Spain. Association for Computational Linguistics.

Yanbing Xue and Milos Hauskrecht. 2019. Active learning of multi-class classification models from ordered class sets. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:5589–5596.