How Reliable are AMR Parser Outputs? Uncertainty Estimation for AMR Parsing

Anuj Kumar[1,2], **Shintaro Ozaki**[2],
Yamuna Prasad[1], **Katsuhiko Hayashi**[2,3], **Hidetaka Kamigaito**[2], **Taro Watanabe**[2]
[1]Indian Institute of Technology Jammu, India
[2]Nara Institute of Science and Technology (NAIST), Japan
[3]The University of Tokyo, Japan
{anuj,yamuna.prasad}@iitjammu.ac.in, katsuhiko-hayashi@g.ecc.u-tokyo.ac.jp
{ozaki.shintaro.ou6,kamigaito.h,taro.watanabe}@naist.ac.jp

Abstract

Recent advances have significantly improved the performance of Abstract Meaning Representation (AMR) parsers, yet determining when their outputs can be trusted remains largely unexplored. This work presents the first study of uncertainty estimation for AMR parsers in a black-box setting using top-k beam-search candidates. Uncertainty is estimated from pairwise candidate similarities and evaluated using the Prediction Rejection Ratio. This work characterises the relationship between parsing performance and uncertainty, evaluates the reliability of beam candidate distributions, analyses the effect of evaluation metrics on uncertainty assessment, and identifies parsers that provide the strongest balance between performance and reliability. Experimental results show the value of uncertainty estimation as a complementary dimension for evaluating AMR parsers.

1 Introduction

Abstract Meaning Representation (AMR) represents sentence meaning as a rooted semantic graph that captures concepts and their relations beyond surface text (Banarescu et al., 2013). By modelling predicate–argument structure, AMR supports meaning-centric NLP applications including information extraction (Han et al., 2025), question answering (Rakshit and Flanigan, 2025), summarisation (Qiu et al., 2024), multilingual semantic parsing (Vanroy and Van de Cruys, 2024), and trustworthy AI (Regan et al., 2024; Pust et al., 2015).

However, manually constructing AMR graphs is expensive and difficult to scale, motivating the development of automatic AMR parsers that convert text into structured semantic representations. Advances in pretrained language models have substantially improved parsers such as SPRING (Bevilacqua et al., 2021), IBM-Parser (Fernandez Astudillo et al., 2020), HMG-AMR (Sataer et al., 2024), MBSE (Lee et al., 2022), and IBM-Neural (Drozdov et al., 2022), making AMR parsing increasingly important for downstream applications where graph quality directly affects reasoning and decision-making.

Despite recent advances in AMR parsing, an important question remains unresolved: how confident should a parser be in its prediction? Recent studies have explored uncertainty estimation for large language models using probabilistic, diversity-based, and graph-based approaches (Kuhn et al., 2023; Lin et al., 2024; Hashimoto et al., 2025; Vashurin et al., 2025), highlighting the importance of estimating prediction reliability beyond output generation. However, Uncertainty Estimation (UE) for AMR parsing remains largely unexplored. Existing evaluation protocols rely on graph similarity metrics (Cai and Knight, 2013; Opitz et al., 2020; Anchiêta et al., 2019; Song and Gildea, 2019; Opitz, 2023; Opitz et al., 2021), which compare predicted graphs against gold annotations and measure aggregate parser performance. Nevertheless, these metrics provide limited insight into the reliability of individual predictions and into whether a parser's confidence aligns with its actual performance. To the best of our knowledge, this work presents the first systematic study of uncertainty estimation for AMR parsing in a black-box setting and addresses the following research questions:

RQ1: Do performance and uncertainty correlate in AMR parsing?

RQ2: How reliable are beam candidates?

RQ3: Is relying on a single metric enough to judge the uncertainty of AMR parsers?

RQ4: Which parser is best in terms of both performance and reliability?

Proceedings of the 2026 Conference on Empirical Methods in Natural Language Processing, pages 31144–31159
October 24-29, 2026 ©2026 Association for Computational Linguistics

Our experiments[1] evaluate uncertainty estimation across multiple AMR parsers in a black-box setting. For each input, the parser generates a set of beam candidates, from which an uncertainty estimator assigns a scalar uncertainty score; higher scores indicate greater uncertainty. In this work, *Uncertainty Estimation* (UE) refers to the process of assigning these scores, while *Reliability* refers to how well the scores reflect the actual quality of predictions. The reliability of these uncertainty estimates is evaluated using the Prediction Rejection Ratio (PRR) (Malinin et al., 2017), which measures how effectively the uncertainty scores identify low-quality predictions: a reliable estimator should assign higher uncertainty to predictions with lower parsing quality, such that rejecting the most uncertain predictions improves the remaining average quality. Results show a general tendency for higher-performing parsers to produce more reliable uncertainty estimates, although parser architecture and candidate-generation behaviour also influence reliability. Moderate beam sizes yield the most reliable uncertainty estimates, while different evaluation metrics can lead to different conclusions about their reliability. These findings demonstrate that uncertainty estimation provides a complementary perspective beyond parsing accuracy for evaluating AMR parsers.

2 Problem Definition

Let $\mathcal{D} = \{(s_i, g_i^*)\}_{i=1}^n$ denote a dataset of size n, where s_i denotes an input sentence and g_i^* its corresponding gold AMR graph. Given sentence s_i, an AMR parser generates k beam candidates

$$\mathcal{C}_i = \{g_i^{(1)}, \ldots, g_i^{(k)}\}, \qquad g_i^{(1)} = \hat{g}_i, \tag{1}$$

where $\mathcal{C}_i$ denotes the set of k candidate AMR graphs, $g_i^{(j)}$ the j-th beam prediction, and $\hat{g}_i$ the top-1 prediction. In the black-box setting, parsing is defined as $P : s_i \mapsto \mathcal{C}_i$ without access to model probabilities or internal states. The objective is to learn an uncertainty estimator U that maps beam candidates to a scalar uncertainty score,

$$U : \mathcal{C}_i \mapsto u_i, \qquad u_i \in \mathbb{R} \tag{2}$$

yielding one uncertainty score per input sentence and uncertainty vector $\boldsymbol{u} = (u_1, \ldots, u_n) \in \mathbb{R}^n$. Prediction quality is computed between the top-1 prediction and gold graph producing quality vector $\boldsymbol{q} = (q_1, \ldots, q_n) \in \mathbb{R}^n$. A desirable estimator satisfies $u_i \propto (1 - q_i)$ and maximises PRR.

[1] https://github.com/naist-nlp/UE_Method_AMR_Parser

3 Methodology

This work estimates uncertainty for AMR parsing in a black-box setting, where model probabilities and internal states are unavailable. The key intuition is that a set of beam candidates with high semantic agreement indicates lower uncertainty, whereas greater disagreement among candidates indicates higher uncertainty. For each input sentence, pairwise similarities among the candidate AMR graphs are computed to construct a similarity matrix (§3.1), which is then converted into a scalar uncertainty score (§3.2). The quality of these uncertainty estimates is evaluated using the Prediction Rejection Ratio (PRR) (§3.3), which measures how effectively the uncertainty scores identify low-quality predictions and whether rejecting the most uncertain predictions improves the quality of the remaining predictions.

3.1 Pairwise Similarity Matrix

For each candidate set $\mathcal{C}_i$, pairwise candidate agreement is represented by

$$[\mathbf{S}_i]_{jl} = F_{\text{sim}}(g_i^{(j)}, g_i^{(l)}), \qquad \mathbf{S}_i \in \mathbb{R}^{k \times k}, \tag{3}$$

where F_{sim} denotes an AMR similarity function, $[\mathbf{S}_i]_{jl} \in [0, 1]$ measures similarity between beam candidates $g_i^{(j)}$ and $g_i^{(l)}$, and $[\mathbf{S}_i]_{jj} = 1$. This procedure produces one similarity matrix $\mathbf{S}_i$ per sentence.

3.2 Uncertainty Estimation

Each uncertainty estimator maps beam candidates to an uncertainty score as $U : \mathcal{C}_i \mapsto \mathbf{S}_i \mapsto u_i$, where $\mathbf{S}_i \in \mathbb{R}^{k \times k}$ is the pairwise similarity matrix, $u_i \in \mathbb{R}$ is the scalar uncertainty score for sentence s_i, and $\boldsymbol{u} = (u_1, \ldots, u_n) \in \mathbb{R}^n$ is the uncertainty vector. We use the following estimators:

LexSim Following Kuhn et al. (2023), lexical uncertainty is estimated from pairwise ROUGE-L (Lin, 2004) similarity:

$$u_i^{\text{LexSim}} = -\frac{1}{k(k-1)} \sum_{j \neq l} [\mathbf{R}_i]_{jl} \tag{4}$$

where $[\mathbf{R}_i]_{jl} \in [0, 1]$ denotes the ROUGE-L similarity between beam candidates $g_i^{(j)}$ and $g_i^{(l)}$.

Graph-Based Methods Following Lin et al. (2024), $\mathbf{S}_i$ is interpreted as a weighted graph. The symmetrized adjacency matrix $\mathbf{A} \in \mathbb{R}^{k\times k}$ and degree matrix $\mathbf{D} \in \mathbb{R}^{k\times k}$ are defined as

$$[\mathbf{A}]_{jl} = \frac{[\mathbf{S}_i]_{jl} + [\mathbf{S}_i]_{lj}}{2}, \qquad [\mathbf{D}]_{jj} = \sum_l [\mathbf{A}]_{jl} \tag{5}$$

and transformed into the normalised Laplacian

$$\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}, \tag{6}$$

where $\mathbf{L} \in \mathbb{R}^{k\times k}$ and $\lambda_1, \ldots, \lambda_k$ denote its eigenvalues. The normalised Laplacian provides a spectral representation of candidate connectivity: the multiplicity of its zero eigenvalue equals the number of connected components in the graph (von Luxburg, 2007, Proposition 4). This property motivates the use of the Laplacian eigenvalues and eigenvectors in the following **EigV** of Equation 7 and **Ecc** of Equation 8.

EigV estimates uncertainty from eigenvalue deviations, where larger deviations $(1 - \lambda_j)$ correspond to greater candidate disagreement:

$$u_i^{\text{EigV}} = \sum_{j=1}^{k} \max(0, 1 - \lambda_j). \tag{7}$$

Ecc estimates uncertainty from candidate spread in the Laplacian eigenspace. Let $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k \in \mathbb{R}^k$ denote the eigenvectors of $\mathbf{L}$. Each candidate $g_i^{(j)}$ is embedded as $\boldsymbol{e}_j = (v_{1,j}, \ldots, v_{k,j})^\top \in \mathbb{R}^k$ and mean-centred as $\tilde{\boldsymbol{e}}_j = \boldsymbol{e}_j - \frac{1}{k}\sum_{l=1}^{k} \boldsymbol{e}_l$; uncertainty is the Frobenius norm of the centred embeddings:

$$u_i^{\text{Ecc}} = \sqrt{\sum_{j=1}^{k} \|\tilde{\boldsymbol{e}}_j\|^2}. \tag{8}$$

DegMat does not rely on the normalized Laplacian and estimates uncertainty from average candidate agreement, where smaller $\sum_{j,l}[\mathbf{A}]_{jl}$ indicates higher uncertainty:

$$u_i^{\text{DegMat}} = 1 - \frac{\text{tr}(\mathbf{D})}{k^2} = 1 - \frac{1}{k^2}\sum_{j,l}[\mathbf{A}]_{jl}. \tag{9}$$

3.3 Prediction Rejection Evaluation

An uncertainty estimator should assign a low uncertainty score to reliable predictions and a high uncertainty score to unreliable predictions. PRR (Malinin et al., 2017) evaluates this behaviour by measuring quality improvement after rejecting uncertain outputs.

Prediction quality is computed between the top-1 prediction $\hat{g}_i$ and the gold graph g_i^*:

$$q_i = \mathcal{F}_{\text{qual}}(\hat{g}_i, g_i^*) \in \mathbb{R} \tag{10}$$

yielding quality vector $\boldsymbol{q} = (q_1, \ldots, q_n) \in \mathbb{R}^n$. Given uncertainty vector $\boldsymbol{u} \in \mathbb{R}^n$, predictions are sorted by increasing uncertainty $\boldsymbol{\pi} = \text{argsort}(\boldsymbol{u}) \in \mathbb{Z}^n$. At rejection rate $\alpha \in [0, 1]$, the $\lfloor \alpha n \rfloor$ most uncertain predictions are removed, and the remaining mean quality is

$$Q(\alpha) = \frac{1}{n - \lfloor \alpha n \rfloor} \sum_{i=1}^{n-\lfloor \alpha n \rfloor} q_{\pi_i} \in \mathbb{R} \tag{11}$$

where π_i is the i-th element of $\boldsymbol{\pi}$. The area under the resulting prediction–rejection curve (AUCPR) is normalised against oracle and random rejection:

$$\text{PRR} = \frac{\text{AUCPR}_{\text{unc}} - \text{AUCPR}_{\text{random}}}{\text{AUCPR}_{\text{oracle}} - \text{AUCPR}_{\text{random}}} \in \mathbb{R} \tag{12}$$

where $\text{PRR} = 1$ denotes oracle performance, $\text{PRR} = 0$ random rejection, and $\text{PRR} < 0$ harmful uncertainty estimates. Further details on the algorithmic framework and AUCPR derivation are provided in Appendix A.

4 Experiments and Results

4.1 Experimental Setup

Experiments are conducted on the AMR 3.0 test set (Knight et al., 2020), comprising $n = 1,898$ sentence–graph pairs, using six AMR parsers[2]: SPRING (Bevilacqua et al., 2021), IBM Transition Parser (Fernandez Astudillo et al., 2020), HMG-AMR-BART (Sataer et al., 2024), HMG-AMR-T5 (Sataer et al., 2024), MBSE (Lee et al., 2022), and IBM-Neural (Drozdov et al., 2022). For each parser, beam search generates candidate sets with $k \in \{3, 5, 8, 10\}$. Fewer than 1% of generated beam candidates are invalid across parsers and beam sizes and are assigned zero similarity when computing graph-based similarity measures.

AMR similarity function defined in Equation (3) is measured using Smatch (Cai and Knight, 2013), Smatch++ (Opitz, 2023), S^{2}Match (Opitz et al., 2020), SEMA (Anchiêta et al., 2019), and WLK (Opitz et al., 2021), while prediction quality defined in Equation (10) is evaluated against gold

[2]Parsers were selected based on reported performance and code availability. LLM baselines were excluded due to substantially lower parsing performance. Additional details are provided in Appendix C.1

Parser	Smatch	Smatch++
SPRING (Bevilacqua et al., 2021)	0.8449	0.8438
MBSE (Lee et al., 2022)	0.8374	0.8378
HMG-AMR-BART (Sataer et al., 2024)	0.8358	0.8380
IBM-Parser (Fernandez Astudillo et al., 2020)	0.8128	0.8181
IBM-Neural (Drozdov et al., 2022)	0.8115	0.8174
HMG-AMR-T5 (Sataer et al., 2024)	0.7693	0.7797

Table 1: Top-1 AMR parsing performance measured using Smatch and Smatch++.

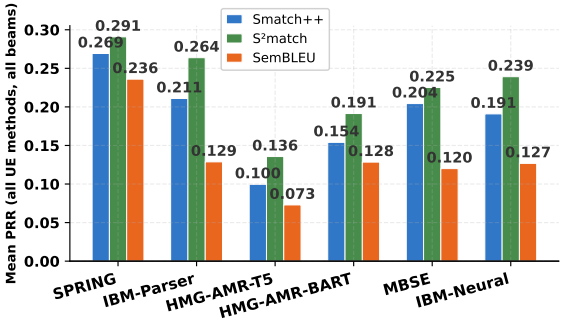

Figure 1: Mean PRR for each parser and prediction quality metric, averaged across all beam sizes and uncertainty estimators.

graphs using Smatch++ (Opitz, 2023), S^2Match, and SemBLEU (Song and Gildea, 2019). Additional implementation details, parser checkpoints, similarity metrics, and hardware configuration are provided in Appendix B.

4.2 Results and Analysis

RQ1: Do performance and uncertainty correlate in AMR parsing? Table 1 and Figure 1 compare parsing performance and PRR score across parsers. Overall, parsers with higher top-1 Smatch scores tend to achieve higher PRR, indicating a positive relationship between parsing accuracy and reliability. However, parsers with comparable Smatch scores exhibit substantially different PRR values; for instance, MBSE and HMG-AMR-BART have mean PRRs of 0.183 and 0.154, respectively, under Smatch++, despite near-identical parsing accuracy, suggesting that candidate-generation behaviour and architectural design also independently influence reliability. Additional results are provided in Appendix C.1 and Appendix C.2, where positive correlations are observed consistently across evaluation metrics. Quantitatively, the Spearman correlation between parsing performance and mean PRR is $\rho = 0.388$ for Smatch++, $\rho = 0.317$ for S^2Match, and $\rho = 0.360$ for SemBLEU, with the strongest correlation observed under Smatch++. These results further support the positive relationship between parsing performance and uncertainty-estimation reliability.

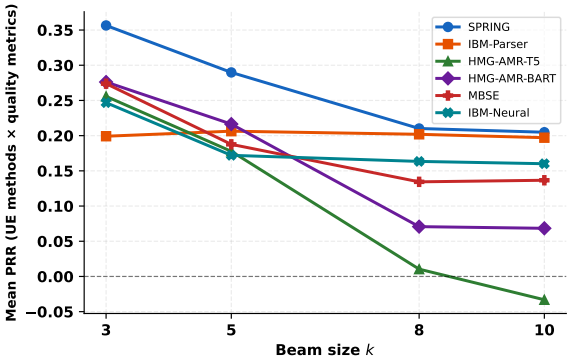

Figure 2: Mean PRR averaged across uncertainty estimation methods and prediction quality metrics for beam sizes $k \in \{3, 5, 8, 10\}$.

RQ2: How reliable are beam candidates? Beam candidates provide meaningful uncertainty estimates, as shown in Figure 2. However, PRR declines across most parsers as beam size increases, suggesting that larger beams introduce a more diverse set of candidates, weakening the relationship between uncertainty and prediction quality. The rate of degradation varies substantially: HMG-AMR-T5 falls below zero at $k = 10$, indicating that the uncertainty estimates are actively harmful, whereas transition-based parsers remain comparatively stable, potentially because their architecture explores structured action sequences rather than relying solely on graph linearisation. Overall, moderate beam sizes provide the strongest trade-off between candidate diversity and uncertainty quality. Additional analysis across beam sizes and uncertainty estimators is provided in Appendix C.3, where LexSim is the only estimator whose PRR consistently increases with beam size, whereas Ecc exhibits the steepest degradation. These results show that increasing the beam size does not affect all uncertainty estimators equally, as their ability to capture candidate agreement varies.

RQ3: Is relying on a single metric enough to judge the uncertainty of AMR parsers? No, the uncertainty of AMR parsers cannot be judged by using a single evaluation metric. As illustrated in Figure 3, PRR varies across metrics, with transition-based parsers achieving stronger PRR under S^2Match than Smatch++, indicating that uncertainty assessment depends on the notion of graph similarity being measured. Additional analysis is provided in Appendix C.4, where graph-based

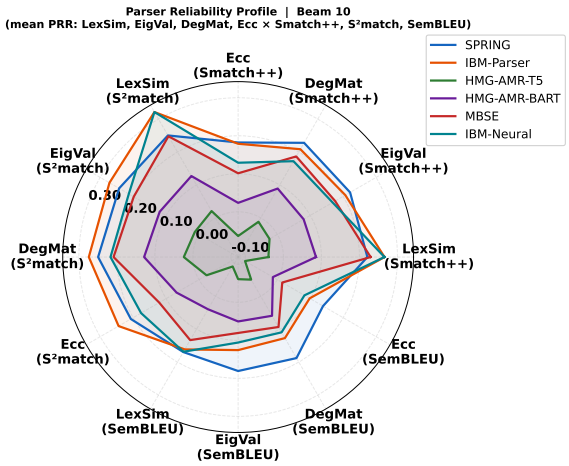

Figure 3: Each value shows the mean PRR averaged across uncertainty estimation methods and prediction quality metrics.

estimators show strong mutual agreement under Smatch++, while LexSim shows near-zero or negative correlation with most graph-based methods. Under SemBLEU, estimator agreement becomes more uniform, indicating that the relationship between uncertainty estimators also changes with the evaluation metric. These observations reinforce that reliability is jointly determined by the uncertainty estimator and the metric used to assess prediction quality, making multi-metric evaluation important for a robust assessment.

RQ4: Which parser is best in terms of both performance and reliability? SPRING (Bevilacqua et al., 2021) achieves the strongest overall balance, obtaining the highest mean PRR (0.265) while maintaining positive uncertainty estimates across all beam sizes and evaluation metrics as shown in Figure 1 and 2. Its seq2seq formulation with graph-isomorphic linearisation reduces reliance on heuristic processing, preserving semantically coherent candidate sets as beam diversity increases. Additional parser reliability results across beam sizes are provided in Appendix C.5, where IBM-Parser remains the most stable across all metrics. At the same time, HMG-AMR-T5 exhibits the steepest degradation, approaching zero or negative PRR at $k = 8$ and $k = 10$. Detailed estimation results at $k = 10$ are provided in Appendix C.6, where DegMat-SEMA achieves the strongest PRR for SPRING and HMG variants, while LexSim outperforms graph-based methods for IBM-Parser and IBM-Neural. These results highlight that parser selection involves a trade-off between overall parsing performance and the stability of uncertainty estimation across beam sizes. While SPRING provides the strongest overall balance, IBM-Parser may be preferable when stable uncertainty estimation across different beam sizes is prioritised.

4.3 Additional Analysis

To further investigate factors affecting uncertainty estimation and reliability, the analysis examines sentence length and agreement between uncertainty estimators. Short sentences (≤ 9 tokens) receive substantially higher mean uncertainty scores than medium- and long-length sentences; however, PRR indicates that uncertainty estimates are most reliable for medium-length sentences, followed by long and short sentences. The analysis further shows strong agreement among graph-based estimators. Additional analysis is provided in Appendix C.3 and C.6 showing that parser reliability varies substantially with beam size: some parsers remain stable as the beam expands, whereas others exhibit pronounced degradation. At larger beam sizes, the most reliable UE also varies across parsers, indicating that no single estimator consistently provides the strongest reliability.

5 Conclusion

This work presents UE for AMR parsers in a black-box setting. Across multiple AMR parsers, beam sizes, and evaluation metrics, it is observed that more accurate parsers generally yield more reliable uncertainty estimates. At the same time, parser reliability also varies with beam size and parser architecture. The results further show that the reliability of UE depends on the evaluation metric and that moderate beam sizes tend to yield more reliable UE than larger beams. Importantly, the experiments demonstrate that parsing performance and uncertainty estimation with reliability provide complementary information for comparing AMR parsers. Parsing accuracy measures the quality of the predicted AMRs, whereas PRR evaluates how effectively the estimated uncertainty identifies potentially low-quality predictions. Since parsers with similar parsing accuracy can exhibit substantially different PRR, considering both performance and reliability provides a more comprehensive assessment of parser behaviour. Overall, the findings support the use of both conventional accuracy-based evaluation and uncertainty-based reliability evaluation when comparing AMR parsers.

Limitations

This work studies uncertainty estimation for AMR parsing under a black-box setting using beam-search candidates as the sole source of uncertainty. Although beam decoding is adopted by most publicly available AMR parsers and therefore provides a consistent evaluation setting across models, the behaviour of uncertainty under alternative decoding strategies (e.g., sampling-based or diverse decoding methods) remains unexplored and may yield different candidate distributions and reliability characteristics.

A second limitation concerns evaluation. Uncertainty quality in this study is measured using the PRR alongside existing AMR quality metrics. However, current AMR evaluation metrics capture distinct aspects of graph quality and can yield different conclusions about uncertainty, as observed in our experiments. This suggests that uncertainty-aware evaluation for AMR remains an open problem and that future work may benefit from developing reliability-oriented evaluation protocols that complement conventional parsing metrics.

Finally, this study focuses on post-hoc uncertainty estimation without modifying parser architectures or training objectives. Integrating uncertainty directly into AMR parser training and decoding may further improve reliability.

Acknowledgment

The authors would like to thank the Japan Science and Technology Agency (JST) for supporting the research activity for this project at the Nara Institute of Science and Technology (NAIST) through the LOTUS Programme (India-Japan Circulation of Talented Youths in Science Programme).

Ethical Considerations

AI-assisted tools were used to support parts of the research workflow. ChatGPT[3] was used for writing support and language refinement, Claude Code[4] was used to accelerate research workflows, and Grammarly[5] was used for grammar correction and proofreading. All experiments, analyses, research decisions, and final manuscript content were conducted and verified by the authors.

[3] https://chat.openai.com/

[4] https://www.anthropic.com/claude-code

[5] https://www.grammarly.com/

References

Rafael Torres Anchiêta, Marco Antonio Sobrevilla Cabezudo, and Thiago Alexandre Salgueiro Pardo. 2019. Sema: an extended semantic evaluation metric for amr. *ArXiv*, abs/1905.12069.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract Meaning Representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186.

Michele Bevilacqua, Rexhina Blloshmi, and Roberto Navigli. 2021. One spring to rule them both: Symmetric amr semantic parsing and generation without a complex pipeline. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(14):12564–12573.

Shu Cai and Kevin Knight. 2013. Smatch: an evaluation metric for semantic feature structures. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 748–752.

Andrew Drozdov, Jiawei Zhou, Radu Florian, Andrew McCallum, Tahira Naseem, Yoon Kim, and Ramón Astudillo. 2022. Inducing and using alignments for transition-based AMR parsing. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1086–1098.

Ramón Fernandez Astudillo, Miguel Ballesteros, Tahira Naseem, Austin Blodgett, and Radu Florian. 2020. Transition-based parsing with stack-transformers. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1001–1007.

Peitao Han, Lis Pereira, Fei Cheng, Wan Jou She, and Eiji Aramaki. 2025. AMR-RE: Abstract Meaning Representations for retrieval-based in-context learning in relation extraction. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 4: Student Research Workshop)*.

Wataru Hashimoto, Hidetaka Kamigaito, and Taro Watanabe. 2025. Decoding uncertainty: The impact of decoding strategies for uncertainty estimation in large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 14601–14613.

Kevin Knight, Bianca Badarau, Laura Baranescu, Claire Bonial, Madalina Bardocz, Kira Griffitt, Ulf Hermjakob, Daniel Marcu, Martha Palmer, Tim O'Gorman, and Nathan Schneider. 2020. Abstract Meaning Representation (AMR) Annotation Release 3.0. LDC2020T02.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation.

In *The Eleventh International Conference on Learning Representations*.

Young-Suk Lee, Ramón Astudillo, Hoang Thanh Lam, Tahira Naseem, Radu Florian, and Salim Roukos. 2022. Maximum Bayes Smatch ensemble distillation for AMR parsing. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5379–5392.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. 2024. Generating with confidence: Uncertainty quantification for black-box large language models. *Transactions on Machine Learning Research*.

Alexander H. Liu, Kartik Khandelwal, Sandeep Subramanian, Victor Jouault, Abhinav Rastogi, Adrien Sadé, Alan Jeffares, Albert Jiang, Alexandre Cahill, Alexandre Gavaudan, Alexandre Sablayrolles, Amélie Héliou, Amos You, Andy Ehrenberg, Andy Lo, Anton Eliseev, Antonia Calvi, Avinash Sooriyarachchi, Baptiste Bout, and 101 others. 2026. Ministral 3. *Preprint*, arXiv:2601.08584.

Andrey Malinin, Anton Ragni, Kate Knill, and Mark Gales. 2017. Incorporating uncertainty into deep learning for spoken language assessment. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 45–50.

Juri Opitz. 2023. SMATCH++: Standardized and extended evaluation of semantic graphs. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1595–1607.

Juri Opitz, Angel Daza, and Anette Frank. 2021. Weisfeiler-leman in the bamboo: Novel AMR graph metrics and a benchmark for AMR graph similarity. *Transactions of the Association for Computational Linguistics*, 9:1425–1441.

Juri Opitz, Letitia Parcalabescu, and Anette Frank. 2020. AMR similarity metrics from principles. *Transactions of the Association for Computational Linguistics*, 8:522–538.

Michael Pust, Ulf Hermjakob, Kevin Knight, Daniel Marcu, and Jonathan May. 2015. Parsing English into Abstract Meaning Representation using syntax-based machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1143–1154.

Haoyi Qiu, Kung-Hsiang Huang, Jingnong Qu, and Nanyun Peng. 2024. AMRFact: Enhancing summarization factuality evaluation with AMR-driven negative samples generation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*.

Geetanjali Rakshit and Jeffrey Flanigan. 2025. ASQ: Automatically generating question-answer pairs using AMRs. In *Proceedings of the 15th International Conference on Recent Advances in Natural Language Processing - Natural Language Processing in the Generative AI Era*, pages 1003–1011.

Michael Regan, Shira Wein, George Baker, and Emilio Monti. 2024. MASSIVE multilingual Abstract Meaning Representation: A dataset and baselines for hallucination detection. In *Proceedings of the 13th Joint Conference on Lexical and Computational Semantics (*SEM 2024)*.

Yikemaiti Sataer, Yunlong Fan, Bin Li, Miao Gao, Chuanqi Shi, and Zhiqiang Gao. 2024. Hierarchical information matters! improving amr parsing with multi-granularity representation interactions. *Information Processing & Management*, 61(3):103698.

Linfeng Song and Daniel Gildea. 2019. SemBleu: A robust metric for AMR parsing evaluation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4547–4552.

Bram Vanroy and Tim Van de Cruys. 2024. Less is enough: Less-resourced multilingual AMR parsing. In *Proceedings of the 20th Joint ACL - ISO Workshop on Interoperable Semantic Annotation @ LREC-COLING 2024*, pages 82–92.

Roman Vashurin, Ekaterina Fadeeva, Artem Vazhentsev, Lyudmila Rvanova, Daniil Vasilev, Akim Tsvigun, Sergey Petrakov, Rui Xing, Abdelrahman Sadallah, Kirill Grishchenkov, Alexander Panchenko, Timothy Baldwin, Preslav Nakov, Maxim Panov, and Artem Shelmanov. 2025. Benchmarking uncertainty quantification methods for large language models with LM-polygraph. *Transactions of the Association for Computational Linguistics*, 13.

Ulrike von Luxburg. 2007. A tutorial on spectral clustering. *Preprint*, arXiv:0711.0189.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Algorithm 1: Black-box Uncertainty Estimation for AMR Parsing

Input: Dataset $\mathcal{D} = \{(s_i, g_i^*)\}_{i=1}^n$, parser P, beam size k
Output: PRR $\mathrm{PRR} \in \mathbb{R}$
foreach $(s_i, g_i^*) \in \mathcal{D}$ **do**
  Generate k beam candidates via beam search;

$$\mathcal{C}_i = P(s_i) = \{g_i^{(1)}, \ldots, g_i^{(k)}\}$$

  Compute pairwise candidate similarity matrix;

$$\mathbf{S}_i = F_{\mathrm{sim}}(\mathcal{C}_i) \in \mathbb{R}^{k \times k}, \qquad [\mathbf{S}_i]_{jj} = 1,$$

$$[\mathbf{S}_i]_{jl} \in [0, 1]$$

  UE methods;

$$u_i^{\mathrm{LexSim}} = 1 - \frac{1}{k(k-1)} \sum_{j \neq l} \mathrm{ROUGE}(g_i^{(j)}, g_i^{(l)}),$$

$$u_i^{\mathrm{DegMat}} = 1 - \frac{2}{k(k-1)} \sum_{j<l} [\mathbf{S}_i]_{jl},$$

$$u_i^{\mathrm{EigV}} = \frac{1}{k} \sum_{r=1}^{k} (\lambda_r - 1)^2,$$

$$u_i^{\mathrm{Ecc}} = \left(\sum_{j=1}^{k} \left\| \mathbf{e}_j - \frac{1}{k} \sum_{l=1}^{k} \mathbf{e}_l \right\|_2^2 \right)^{1/2}.$$

  Higher u_i indicates greater uncertainty;
  Compute prediction quality against the gold graph;

$$q_i = \mathcal{F}_{\mathrm{qual}}(g_i^{(1)}, g_i^*) \in \mathbb{R}$$

Construct uncertainty and quality vectors;

$$\boldsymbol{u} = (u_1, \ldots, u_n) \in \mathbb{R}^n, \qquad \boldsymbol{q} = (q_1, \ldots, q_n) \in \mathbb{R}^n$$

Sort by increasing uncertainty (π_1 = least uncertain);

$$\boldsymbol{\pi} = \mathrm{argsort}(\boldsymbol{u}) \in \mathbb{Z}^n$$

At rejection rate α, remove $\lfloor \alpha n \rfloor$ most uncertain predictions and compute remaining mean quality;

$$Q(\alpha) = \frac{1}{n - \lfloor \alpha n \rfloor} \sum_{r=1}^{n - \lfloor \alpha n \rfloor} q_{\pi_r}$$

Compute area under the prediction–rejection curve;

$$\mathrm{AUCPR} = \int_0^1 Q(\alpha)\, \mathrm{d}\alpha$$

Normalise against oracle and random baselines;

$$\mathrm{PRR} = \frac{\mathrm{AUCPR}_{\mathrm{unc}} - \mathrm{AUCPR}_{\mathrm{random}}}{\mathrm{AUCPR}_{\mathrm{oracle}} - \mathrm{AUCPR}_{\mathrm{random}}}$$

return PRR

A Algorithm

A.1 Algorithm

Algorithm 1 summarises the complete procedure.

A.2 Computational Complexity

The dominant cost of the proposed framework is pairwise similarity computation, which requires $\mathcal{O}(k^2)$ graph comparisons per sentence. Alignment-based metrics (Smatch, Smatch++, S^{2}Match) scale as $\mathcal{O}(|V|^2 r)$ per comparison, where $|V|$ is the number of graph nodes and r the number of alignment restarts; SEMA scales as $\mathcal{O}(|V| + |E|)$; and WLK scales as $\mathcal{O}(|V||E|h)$, where $|E|$ is the number of edges and h the number of Weisfeiler–Lehman iterations. The overall runtime across P parsers, B beam configurations, and N sentences is therefore $\mathcal{O}(P \cdot B \cdot N \cdot k^2 \cdot C_{\mathrm{sim}})$, where C_{sim} denotes the per-comparison cost of the chosen metric. Uncertainty estimation adds negligible overhead: DEGMAT and ECC operate in $\mathcal{O}(Nk^2)$, EIGV in $\mathcal{O}(Nk^3)$, and PRR evaluation in $\mathcal{O}(N \log N)$ per estimator–metric pair. To avoid redundant computation, all pairwise similarity matrices are cached as compressed `.npz` arrays after the first computation.

B Experiment Setup

This Appendix provides implementation details required to reproduce all experiments.

B.1 Dataset

Experiments are conducted on the AMR 3.0 (https://catalog.ldc.upenn.edu/LDC2020T02) test set containing 1,898 sentence–graph pairs. Each example consists of an input sentence and its corresponding gold AMR graph represented in PENMAN notation. All uncertainty estimation experiments are performed on parser outputs generated from this test split.

B.2 AMR Parsers

Six publicly available AMR parsers are evaluated in this study. All experiments use official implementations and publicly released pretrained checkpoints without modifying model architectures or retraining.

- SPRING (Bevilacqua et al., 2021) (https://github.com/SapienzaNLP/spring)

- IBM-Parser (Fernandez Astudillo et al., 2020) (https://github.com/IBM/transition-amr-parser)
- MBSE (Lee et al., 2022) using checkpoint: amr3joint_ontowiki2_g2g-structured-bart-large-seed44, loaded through the IBM implementation (https://github.com/IBM/transition-amr-parser).
- IBM-Neural (Drozdov et al., 2022) using checkpoint: amr3.0-structured-bart-large-neur-al-seed42, loaded through the IBM implementation (https://github.com/IBM/transition-amr-parser).
- HMG-AMR-T5 (Sataer et al., 2024) (https://github.com/YIKMAT/HMG-AMR/tree/main/HMG-AMR(T5))
- HMG-AMR-BART (Sataer et al., 2024) (https://github.com/YIKMAT/HMG-AMR/tree/main/HMG-AMR(Bart))

For reference, two instruction-tuned LLM baselines (Qwen3-32B and Ministral-14B) were additionally evaluated using direct AMR generation without task-specific training using the transformer library (Wolf et al., 2020). Due to substantially lower performance on AMR 3.0 as shown in Table 3 and the absence of dedicated AMR parsing capability, these models were excluded from uncertainty estimation experiments.

B.3 Similarity Construction and Evaluation

Pairwise candidate similarity matrices $\mathbf{S}_i \in \mathbb{R}^{k \times k}$ are constructed using four AMR-specific similarity metrics:

- Smatch (Cai and Knight, 2013): https://github.com/flipz357/amr-metric-suite
- S^2Match (Opitz et al., 2020): https://github.com/flipz357/amr-metric-suite
- SEMA (Anchiêta et al., 2019): https://github.com/flipz357/amr-metric-suite
- WLK (Opitz et al., 2021): https://github.com/flipz357/weisfeiler-leman-amr-metrics

Prediction quality for PRR evaluation is measured using three metrics:

- Smatch++ (Opitz, 2023): https://github.com/flipz357/smatchpp.git
- S^2Match (Opitz et al., 2020): https://github.com/flipz357/amr-metric-suite
- SemBLEU (Song and Gildea, 2019): https://github.com/freesunshine0316/sembleu

B.4 Uncertainty Estimation and Evaluation

Uncertainty is estimated using one lexical and three graph-based approaches:

- LexSim (Kuhn et al., 2023): estimates uncertainty from mean pairwise ROUGE-L similarity across beam candidates; lower lexical overlap indicates greater candidate diversity and higher uncertainty.
- EigV (Lin et al., 2024): constructs a similarity graph over candidates and measures uncertainty from the eigenvalues of the normalised Laplacian; larger eigenvalue deviations from one indicate weaker graph connectivity and greater candidate disagreement.
- DegMat (Lin et al., 2024): measures uncertainty as the complement of the normalised total pairwise similarity; a lower sum of edge weights in the similarity graph indicates higher uncertainty.
- Ecc (Lin et al., 2024): embeds each candidate in the Laplacian eigenspace and measures uncertainty as the spread of embeddings around their centroid; a larger Frobenius norm indicates greater candidate dispersion and higher uncertainty.

Prediction quality for PRR evaluation is measured using Smatch++ (Opitz, 2023), S^2Match (Opitz et al., 2020), and SemBLEU (Song and Gildea, 2019). Uncertainty quality is evaluated using the PRR (Malinin et al., 2017).

B.5 Hardware and Runtime

All uncertainty estimation and PRR calculations were performed in a CPU-only environment without GPU acceleration. Each job was allocated 4 CPU cores and 64 GB of RAM.

Configuration	Runtime
Per parser (all beams)	∼89,000–90,000 s (≈25 h)
Per parser per beam	∼22,000 s (≈6 h)
Full experiment (6 parsers)	>150 h wall-clock

Table 2: Hardware allocation and observed execution time were measured from job logs.

Parser	Smatch
SPRING (Bevilacqua et al., 2021)	0.8449
MBSE (Lee et al., 2022)	0.8374
HMG-AMR-BART (Sataer et al., 2024)	0.8358
IBM-Parser (Fernandez Astudillo et al., 2020)	0.8128
IBM-Neural (Drozdov et al., 2022)	0.8115
HMG-AMR-T5 (Sataer et al., 2024)	0.7693
Qwen3-32B (Yang et al., 2025)	0.1214
Ministral-14B (Liu et al., 2026)	0.0897

Table 3: Top-1 AMR parsing performance measured using Smatch.

Table 2 reports the observed runtime measured from execution logs. Generating beam candidates and computing uncertainty metrics required approximately 25 hours per parser across all beam sizes ($k \in \{3, 5, 8, 10\}$). Processing a single beam configuration required approximately 6 hours. Running the complete evaluation pipeline across all six parsers took approximately 150 hours of wall-clock time.

C Additional Results

This section presents additional analyses supporting the claims in the main paper, including parser-level performance comparisons, the correlation between AMR parsing quality and uncertainty estimation, and the stability of uncertainty estimators under varying beam sizes.

C.1 AMR Parsing Performance

Table 3 reports parser Smatch score performance on the gold dataset reproduced using the publicly released implementations and default hyperparameter settings provided in the original papers. To ensure consistent comparison across parsers, all results are evaluated before BLINKification. Under this setting, SPRING achieves the highest parsing accuracy (0.8449), followed by MBSE (0.8374) and HMG-AMR-BART (0.8358), indicating that recent encoder–decoder approaches remain highly competitive for AMR generation. IBM-Parser and IBM-Neural obtain comparable performance around 0.81 Smatch, whereas HMG-AMR-T5 shows lower accuracy relative to other specialised AMR parsers. For reference, two instruction-tuned LLM baselines (Qwen3-32B and Ministral-14B) are additionally evaluated using direct AMR generation without task-specific training and exhibit substantially lower performance, highlighting the difficulty of generating valid AMR structures through general-purpose language modelling alone. Due to their poor performance on the AMR 3.0 test set, these models are excluded from subsequent analysis.

C.2 Relationship Between Parsing Performance and Reliability

Figure 4 examines whether AMR parsing accuracy translates into more reliable uncertainty estimates. Each point represents a parser–beam combination (6 parsers × 4 beam sizes = 24 points per panel), where the x-axis shows the parser's fixed top-1 Smatch accuracy and the y-axis shows mean PRR averaged across all 13 uncertainty estimators for that parser and beam size. Because Smatch accuracy is fixed per parser, the four points per parser are stacked vertically, reflecting how PRR varies across beam sizes at the same accuracy level.

Across all evaluation metrics, a moderate positive relationship is observed ($\rho = 0.388$ for Smatch++, $\rho = 0.317$ for S^2Match, and $\rho = 0.360$ for SemBLEU), indicating that parsers with stronger top-1 accuracy tend to achieve higher mean PRR across beam configurations. However, the correlations confirm that accuracy alone does not fully determine the quality of uncertainty. Parsers with comparable Smatch scores often occupy different vertical regions of the PRR space, suggesting that beam decoding behaviour and architectural design also independently influence reliability, beyond accuracy.

The vertical spread of points within each parser further illustrates the effect of beam size on uncertainty quality: some parsers maintain stable PRR across beams, while others experience substantial degradation as beam size increases, reinforcing the findings from RQ2. Finally, the variation in ρ across evaluation metrics indicates that the strength of the performance–uncertainty relationship depends on how prediction quality is defined, consistent with the findings reported in RQ3.

Overall, parser accuracy provides only a partial explanation of uncertainty quality, and reliability emerges as a complementary evaluation dimension that is not captured by conventional AMR parsing metrics alone.

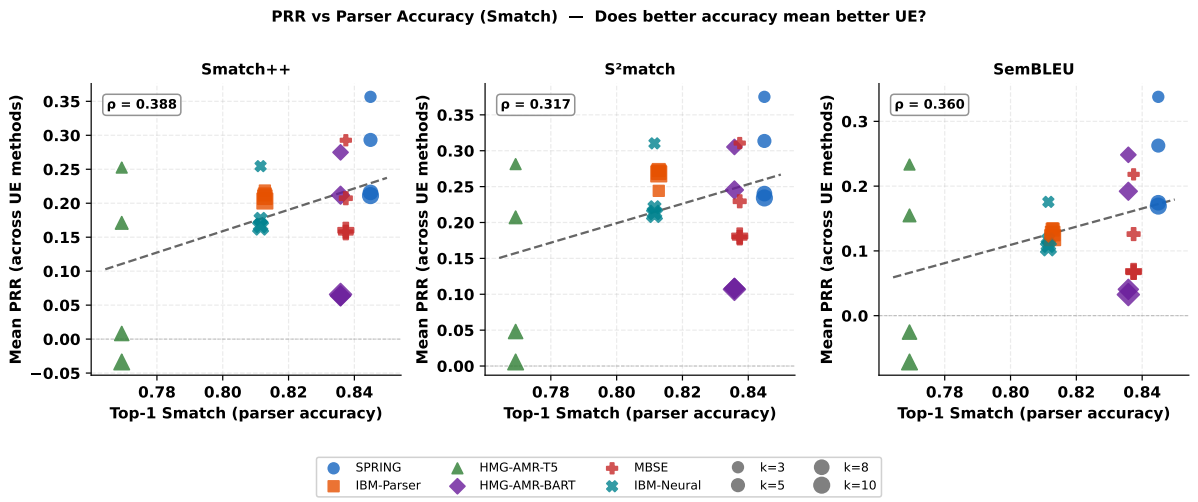

Figure 4: Relationship between parser accuracy (Smatch) and uncertainty quality (PRR) across beam sizes and evaluation metrics.

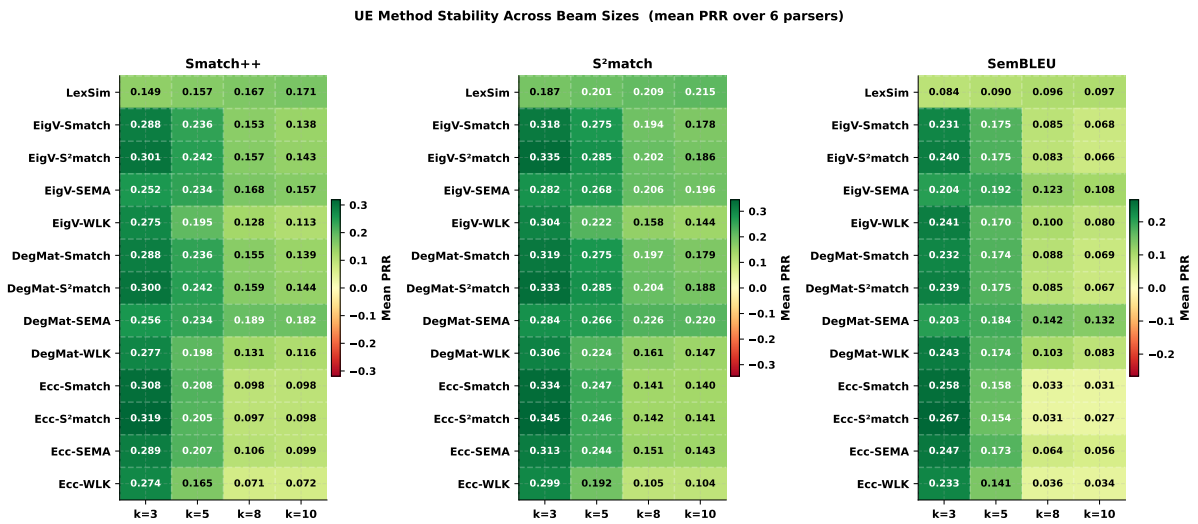

Figure 5: Mean PRR of uncertainty estimators across beam sizes aggregated over all parsers. Stable estimators exhibit less variation in performance across beam sizes.

C.3 Uncertainty Estimator Stability Across Beam Sizes

Figure 5 reports the mean PRR aggregated across all six parsers for each uncertainty estimator and beam size, directly supporting RQ2. The overall pattern is consistent with the main results: uncertainty-estimation reliability is highest at $k = 3$ and generally declines as the beam size increases, with the sharpest drops occurring between $k = 5$ and $k = 8$.

Degradation rates differ markedly across estimators. ECC-based methods start among the strongest at $k = 3$, with ECC-S^{2}MATCH reaching 0.345 under S^{2}Match, yet show the steepest decline, falling to 0.141 at $k = 10$. EIGV variants follow a similar trajectory, dropping from approximately 0.30–0.34 at $k = 3$ to 0.14 - 0.20 at $k = 10$, depending on the metric. Thus, both estimator families show substantial degradation in reliability as the beam size increases.

In contrast, LEXSIM is the only estimator whose mean PRR increases with beam size, rising from 0.149 to 0.171 under Smatch++ and from 0.187 to 0.215 under S^{2}Match, suggesting that its uncertainty estimates are less sensitive to increasing candidate diversity. Among graph-based methods, SEMA-based variants show greater robustness: DEGMAT-SEMA retains PRR values of 0.182 and 0.220 at $k = 10$ under Smatch++ and S^{2}Match, respectively, substantially outperforming other graph-

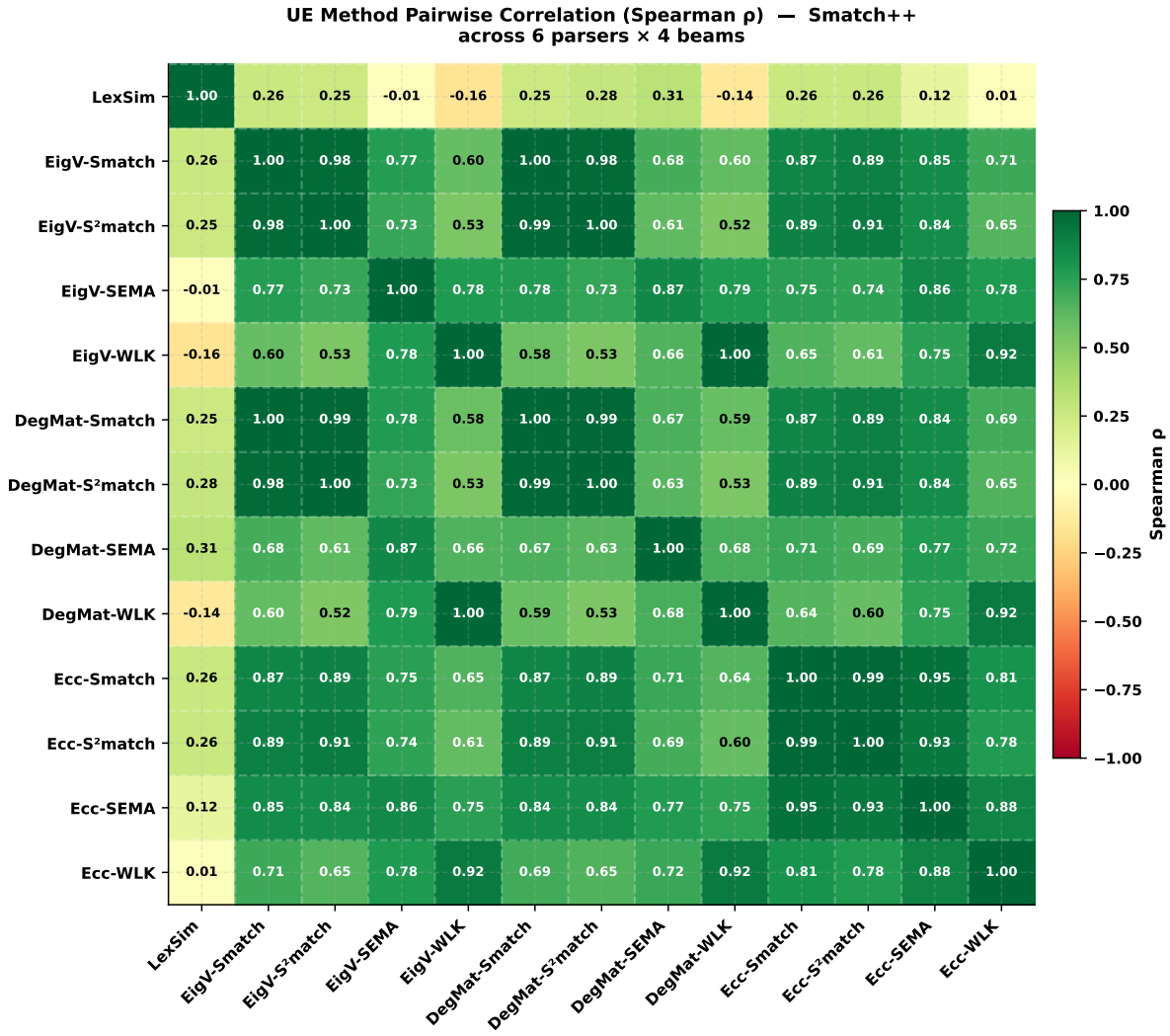

Figure 6: Pairwise Spearman correlation between uncertainty estimators evaluated using Smatch++. Graph-based estimators exhibit strong agreement, while LEXSIM captures a distinct uncertainty signal.

based methods at larger beam sizes.

Notably, S^2Match consistently yields higher mean PRR than Smatch++ across all estimators and beam sizes. For example, EIGV-S^2MATCH obtains 0.335 compared with 0.301 at $k = 3$. This indicates that the observed effect of increasing beam size also depends on how prediction quality is measured, with S^2Match generally producing stronger uncertainty-reliability scores in this analysis. Overall, moderate beam sizes ($k = 3$ to $k = 5$) provide the strongest trade-off between candidate diversity and uncertainty-estimation reliability.

C.4 Agreement Between UE

Figures 6, 8, and 11 report pairwise Spearman correlations between uncertainty estimators across all parser and beam configurations. These results provide additional evidence for RQ3 by examining whether uncertainty estimators produce consistent rankings under different evaluation metrics.

Across all metrics, EIGV, DEGMAT, and ECC exhibit consistently strong mutual agreement. Under Smatch++, most pairwise correlations exceed $\rho = 0.87$, with estimators using the same similarity function approaching near-perfect agreement (e.g. EIGV-SMATCH and DEGMAT-SMATCH: $\rho = 1.00$). Agreement is similarly high under S^2Match, where several pairs reach $\rho = 1.00$, and remains strong under SemBLEU ($\rho \geq 0.85$ for most graph-based pairs). This indicates that graph-based estimators generally preserve similar uncertainty rankings, although their agreement depends on the similarity function used.

In contrast, LEXSIM shows substantially weaker correlation with graph-based estimators across metrics. Under Smatch++, correlations range from

	SPRING		IBM -Parser		HMG-T5		HMG-BART		MBSE		IBM-Neural	
UE Method	**S++**	**SB**	**S++**	**SB**	**S++**	**SB**	**S++**	**SB**	**S++**	**SB**	**S++**	**SB**
LexSim	0.2209	0.1695	**0.2660**	**0.1615**	-0.0395	-0.0904	0.0858	0.0395	**0.2294**	0.1332	**0.2659**	**0.1685**
EigV-Smatch	0.1956	0.1421	0.2218	0.1251	-0.0838	-0.1199	0.0756	0.0359	0.1871	0.0802	0.2325	0.1431
EigV-S2match	0.1912	0.1322	0.2319	0.1232	-0.0794	-0.1176	0.0751	0.0330	0.1931	0.0752	0.2448	0.1473
EigV-SEMA	0.2504	0.2254	0.1919	0.1302	0.0334	-0.0060	0.1080	0.0811	0.1666	0.0859	0.1945	0.1328
EigV-WLK	0.2469	0.2231	0.1807	0.1243	0.0348	-0.0002	0.0605	0.0512	0.1555	0.0811	0.0000	0.0000
DegMat-Smatch	0.1971	0.1438	0.2220	0.1250	-0.0826	-0.1183	0.0776	0.0380	0.1878	0.0807	0.2328	0.1434
DegMat-S2match	0.1925	0.1336	0.2321	0.1234	-0.0755	-0.1141	0.0772	0.0351	0.1942	0.0763	0.2452	0.1477
DegMat-SEMA	**0.2709**	**0.2465**	0.1984	0.1353	**0.0685**	**0.0232**	**0.1369**	**0.1091**	0.2061	**0.1334**	0.2104	0.1472
DegMat-WLK	0.2510	0.2275	0.1814	0.1248	0.0419	0.0074	0.0643	0.0550	0.1580	0.0832	0.0000	0.0000
Ecc-Smatch	0.1853	0.1311	0.2051	0.1023	-0.1136	-0.1452	0.0268	-0.0176	0.1101	0.0073	0.1768	0.1075
Ecc-S2match	0.1817	0.1222	0.2034	0.1011	-0.1117	-0.1477	0.0231	-0.0251	0.1085	0.0035	0.1850	0.1102
Ecc-SEMA	0.1800	0.1506	0.1471	0.0911	-0.0207	-0.0547	0.0271	-0.0015	0.1041	0.0385	0.1542	0.1117
Ecc-WLK	0.1821	0.1513	0.1601	0.0981	-0.0089	-0.0449	0.0168	-0.0104	0.0824	0.0104	0.0000	0.0000

Table 4: PRR at beam size $k = 10$ for all uncertainty estimation methods across parsers and quality metrics (Smatch++ (S++) and SemBLEU (SB)). Higher PRR indicates better alignment between uncertainty estimates and prediction quality. Green cells denote the best-performing uncertainty estimator within each parser–metric pair. Consistent with the main findings, SPRING achieves the strongest overall reliability, while DEGMAT-SEMA frequently emerges as the most effective graph-based uncertainty estimator at larger beam sizes.

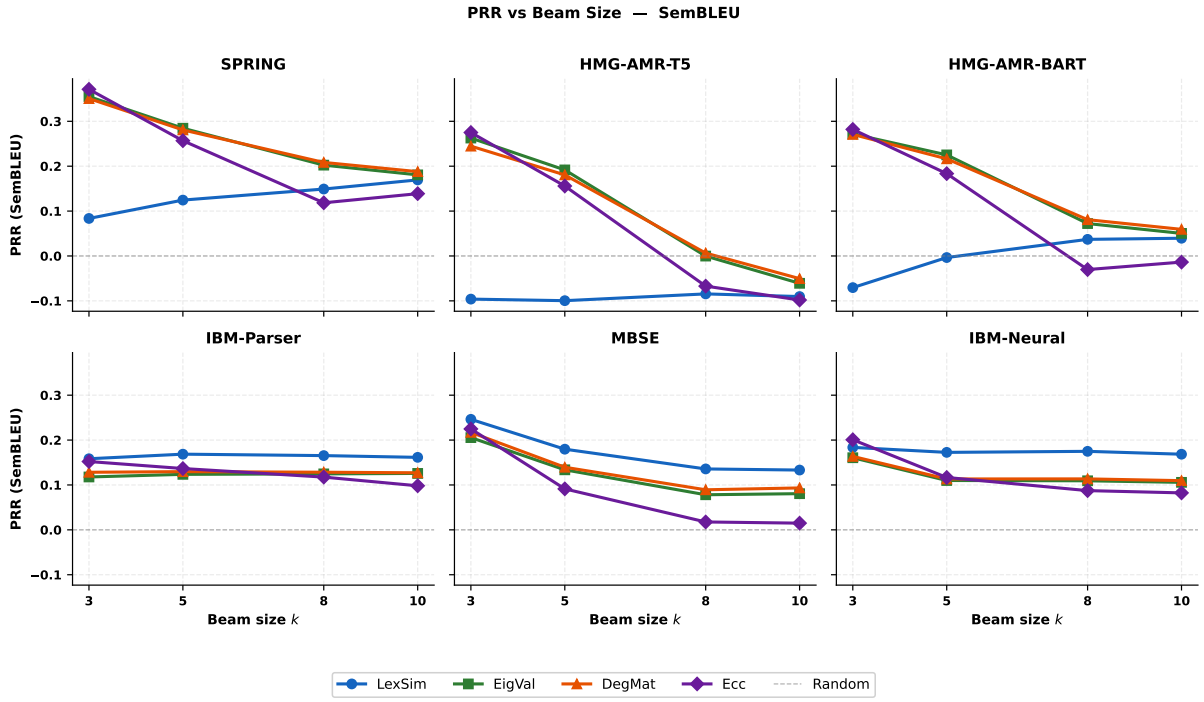

Figure 7: Parser-level PRR as a function of beam size evaluated using SemBLEU. Each subplot corresponds to one parser and compares uncertainty estimators across beam sizes.

$\rho = -0.16$ (EIGV-WLK) to $\rho = 0.31$ (DEGMAT-SEMA), while under SemBLEU they decrease to as low as $\rho = -0.14$ (DEGMAT-WLK). This indicates that lexical overlap captures an uncertainty signal that differs substantially from structural graph agreement, and that the degree of this difference varies with the evaluation metric.

Metric-dependent differences are also observed in the PRR results. Figure 5 shows consistently higher mean PRR under S^2Match than Smatch++ across estimators and beam sizes; for example, EIGV-S^2MATCH obtains 0.335 compared with 0.301 at $k = 3$. Under SemBLEU, WLK-based estimators show stronger agreement with other graph-based methods ($\rho \geq 0.85$), reducing the separation between estimator families observed under Smatch++ and S^2Match. Overall, these results show that changing the evaluation metric can alter both uncertainty reliability and agreement between estimators, indicating that no single metric provides a complete view of uncertainty reliability.

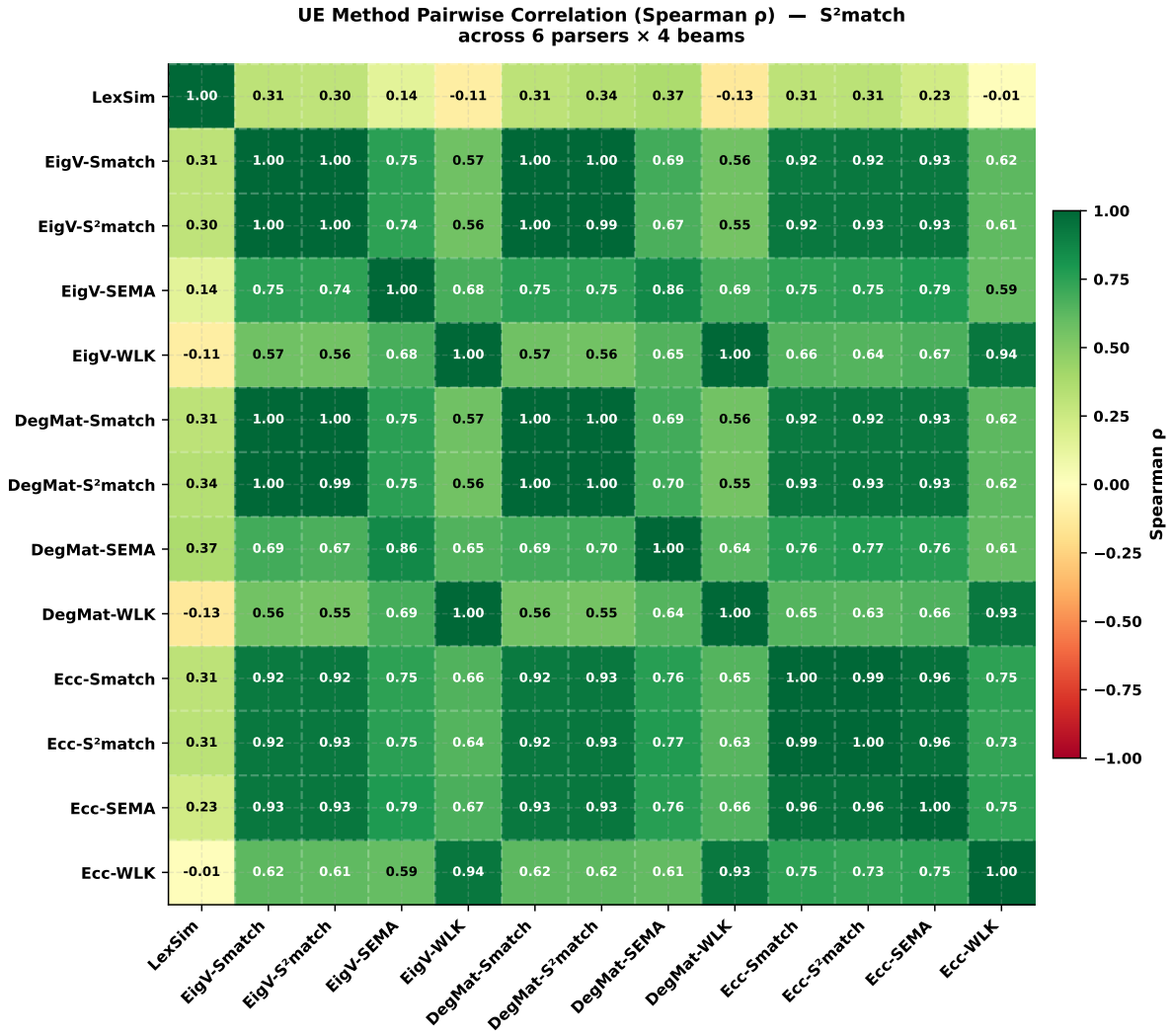

Figure 8: Pairwise Spearman correlation between uncertainty estimators evaluated using S^2Match.

C.5 Parser Reliability Across Beam Sizes

Figures 7, 9, and 10 visualise parser-level PRR across beam sizes, with each curve reporting the mean PRR for a given uncertainty estimator.

SPRING maintains the strongest and most consistent reliability profile across all metrics. Under Smatch++, PRR remains above 0.18 at $k = 10$ for all estimators except ECC, which converges with others at larger beams. Under S^2Match and SemBLEU, SPRING similarly retains positive PRR across all estimators and beam sizes, confirming its robustness to candidate expansion.

IBM-Parser is the most stable parser across beam sizes. Under all three metrics, PRR remains nearly flat from $k = 3$ to $k = 10$, with LEXSIM consistently outperforming graph-based methods and remaining the strongest estimator throughout. IBM-Neural shows a similar stability pattern, with LEXSIM again achieving the highest PRR at larger beams.

HMG-AMR-T5 exhibits the steepest degradation across all metrics. Under Smatch++, graph-based estimators drop below zero by $k = 8$, and LEXSIM also approaches zero by $k = 10$. Under SemBLEU, all estimators reach negative PRR at $k = 8$ and $k = 10$, indicating that beam candidates become actively misleading. HMG-AMR-BART shows similar sensitivity, particularly under SemBLEU, where ECC drops below zero at $k = 10$.

MBSE achieves a competitive PRR at $k = 3$ under all metrics. However, it degrades noticeably at larger beams, particularly under S^2Match where ECC falls to ≈ 0.12 at $k = 10$, substantially below LEXSIM and DEGMAT. Overall, these results reinforce RQ4: the most reliable parser is the one that maintains reliable uncertainty estimates as candidate diversity increases not the one with the highest parsing accuracy.

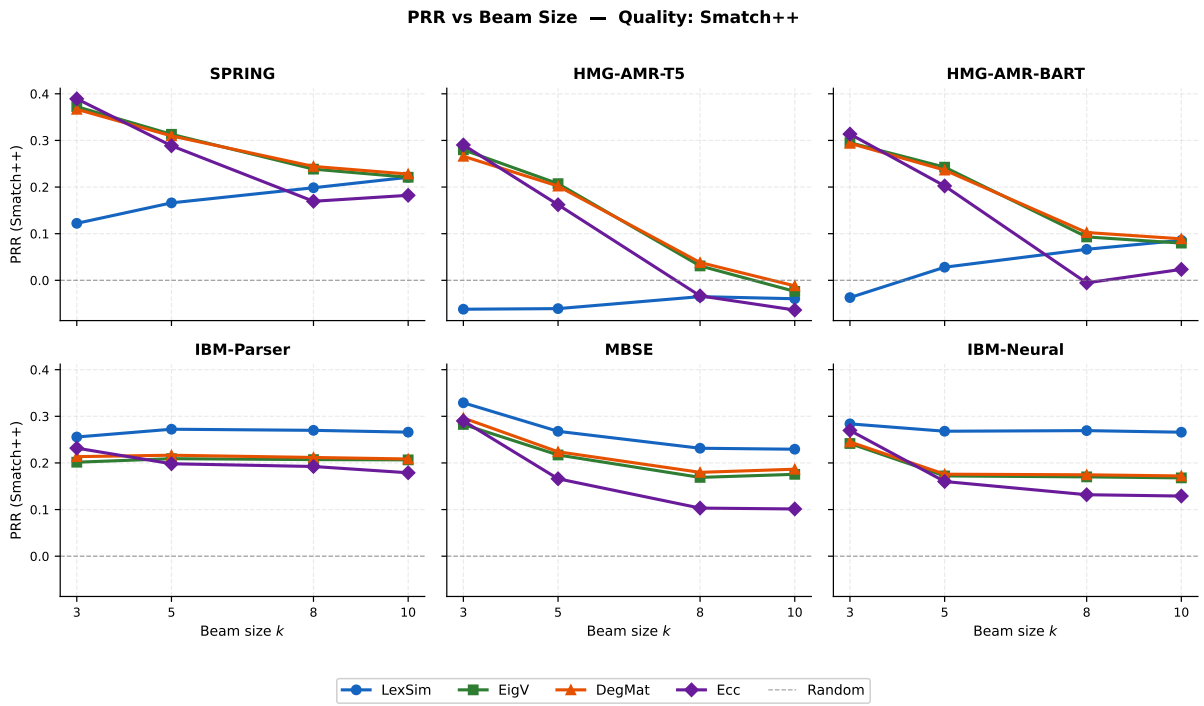

Figure 9: Parser-level PRR as a function of beam size evaluated using Smatch++. Each subplot corresponds to one parser and compares uncertainty estimators across beam sizes.

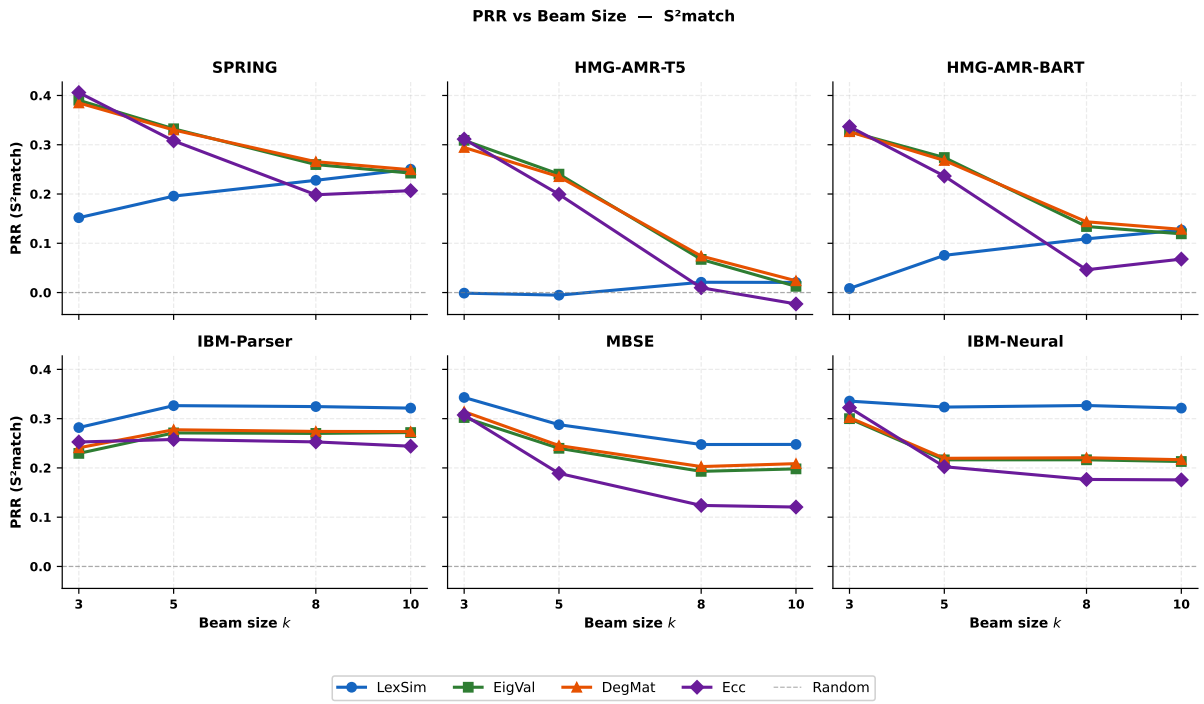

Figure 10: Parser-level PRR as a function of beam size evaluated using S^2Match. Each subplot corresponds to one parser and compares uncertainty estimators across beam sizes.

C.6 Detailed Uncertainty Estimation Results

Table 4 reports complete PRR results at $k = 10$ across all parsers, uncertainty estimators, and evaluation metrics. These results provide a more detailed view of RQ4 by examining whether the reliability of a parser is consistent across different uncertainty estimators at a large beam size. DEGMAT-SEMA achieves the strongest PRR for SPRING under both Smatch++ (0.271) and SemBLEU (0.247), and

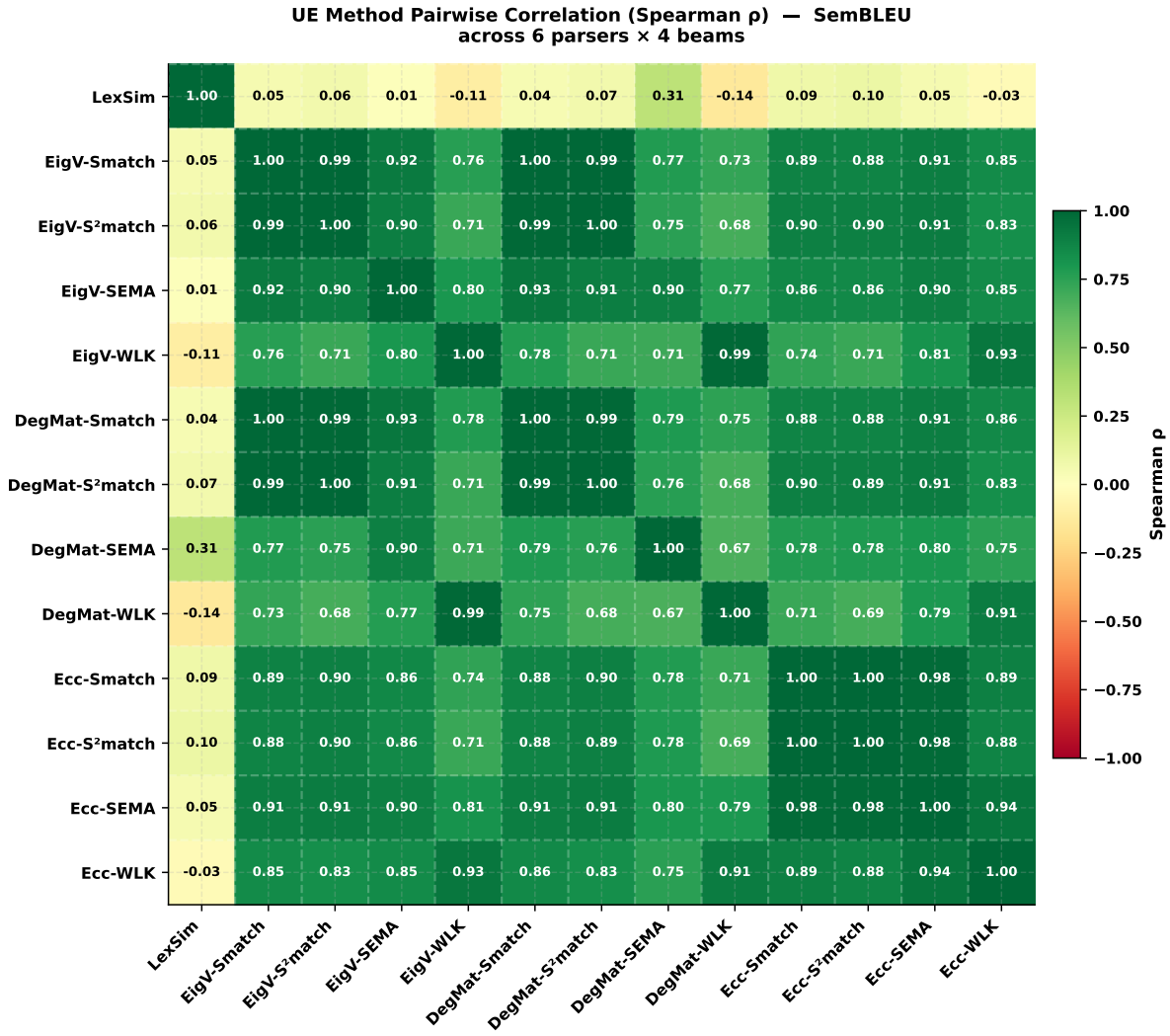

Figure 11: Pairwise Spearman correlation between uncertainty estimators evaluated using SemBLEU.

also leads for HMG-AMR-BART (0.137, 0.109) and HMG-AMR-T5 (0.069, 0.023). In contrast, LexSim achieves the strongest PRR for IBM-Parser (0.266, 0.162), IBM-Neural (0.266, 0.169), and MBSE (0.229, 0.133). Thus, the estimator providing the most reliable uncertainty signal at $k = 10$ varies across parsers, indicating that parser comparison based on reliability is also influenced by the interaction between the parser and the uncertainty estimator.

Ecc-based methods perform weakest overall at $k = 10$, with several parsers recording negative PRR under SemBLEU, such as HMG-AMR-BART (-0.025) and HMG-AMR-T5 (-0.148). This suggests that eigenspace-based candidate spread becomes less informative as beam size increases for these parsers. Overall, the detailed results show that although SPRING provides the strongest overall reliability, its advantage does not imply that a single uncertainty estimator is optimal across all parsers.