UCSC NLP at SemEval-2026 Task 10: Boundary-Aware Span Extraction and RoBERTa Classification for Conspiracy Detection

Dom Marhoefer
UC Santa Cruz
dmarhoef@ucsc.edu

Milos Suvakovic
UC Santa Cruz
msuvakov@ucsc.edu

Glenn Grant-Richards
UC Santa Cruz
ggranri@ucsc.edu

Aidan Pinero
UC Santa Cruz
apinero@ucsc.edu

Ryan King
UC Santa Cruz
rytking@ucsc.edu

Abstract

We present our systems for SemEval-2026 Task 10 (PsyCoMark), addressing conspiracy marker extraction (Subtask 1) and document-level conspiracy detection (Subtask 2). For marker extraction, we formulate the task as multi-label span classification over enumerated candidate spans, using IoU$\geq$0.95 positive labeling, hard-negative sampling, and containment-based non-maximum suppression (NMS) with boundary-aware span representations. Document classification is modeled independently using a sequence classifier with label smoothing and a stratified train–validation split. Analysis shows that entity-like roles (Actor, Victim) are detected robustly, while abstract roles (Action, Effect, Evidence) remain sensitive to boundary criteria. On the official test set, our systems rank 7th in Subtask 1 (0.2251 macro F1) and 12th in Subtask 2 (0.7694 weighted F1).

1 Introduction

Conspiracy narratives are often described in terms of recurring narrative roles: a perceived *Actor* performs an *Action* against a *Victim*, justified by *Evidence* and producing harmful *Effects*. SemEval-2026 Task 10 (PsyCoMark) operationalizes this perspective through two subtasks: extracting conspiratorial roles from text and classifying whether a document expresses a conspiracy narrative (Samory et al., 2026).

- **Subtask 1: Conspiracy-marker extraction.** Given a document, systems predict labeled spans corresponding to five semantic roles (*Actor*, *Action*, *Effect*, *Evidence*, *Victim*). The output is a set of markers of the form `{type, start, end}` per document.

- **Subtask 2: Document-level conspiracy classification.** Each document is assigned one of three labels (*Yes*, *No*, *Can't tell*).

A central feature of the PsyCoMark subtasks is that the previously defined role structures occur in both conspiratorial and non-conspiratorial discourse (critical narrative). Documents labeled as non-conspiratorial may contain complete *Actor–Action–Victim* structures without conspiratorial intent, while conspiratorial documents differ primarily in framing and epistemic stance. As a result, marker extraction and conspiracy detection are intrinsically related but not equivalent: success requires the precise boundary identification of abstract semantic roles while also modeling document-level stance.

For Subtask 1, we use multi-label span classification over enumerated candidate spans with IoU$\geq$0.95 labeling, hard-negative mining, and containment-based NMS; for Subtask 2, we use a document classifier with label smoothing and a stratified train–validation split. On the SemEval evaluation server, our best submissions ranked 7th in Subtask 1 (macro F1 0.2251) and 12th in Subtask 2 (weighted F1 0.7694). Both models achieved competitive performance relative to other shared-task submissions.

Analysis indicates that boundary sensitivity remains a primary failure mode for abstract roles (*Action*, *Effect*, *Evidence*), particularly under stricter token-level overlap criteria. Entity-like roles (*Actor*, *Victim*) were comparatively robust, suggesting that semantic abstraction and variable span length dominate marker difficulty rather than topical content.

2 Background and Task Setup

2.1 PsyCoMark Task and Data

Dataset characteristics. The PsyCoMark corpus contains over 4,100 English Reddit submission statements drawn from more than 190 subreddits (Samory et al., 2026). The document-level conspiracy labels are moderately imbalanced, with 35.3%

Proceedings of the The 20th International Workshop on Semantic Evaluation (2026), pages 1495–1500

Yes, 46.6% No, and 18.1% Can't tell instances. Annotations are dense: the majority of documents contain at least one psycholinguistic marker, and many contain multiple role types.

A **non-conspiratorial** example contains multiple markers without endorsing a conspiracy:

> "[Germany]$_{Actor}$ has [upset]$_{Action}$ [other]$_{Victim}$ EU member states by securing a [disproportionately]$_{Effect}$ large share of the bloc's common pool of vaccines, according to [a]$_{Evidence}$ report..."

In contrast, a **conspiratorial** document expresses conspiratorial framing:

> "So [they]$_{Actor}$ want us to believe it was a [suicide]$_{Effect}$, when [It's]$_{Evidence}$ so blatantly obvious it's not...just because [Jeffrey]$_{Victim}$ is dead doesn't mean it ends here...he was [under suicide watch]$_{Action}$..."

This overlap of role structure across labels makes marker extraction and conspiracy classification intrinsically coupled yet non-identical tasks.

2.2 Related Work and Modeling Motivation

Psycholinguistic structure. Conspiracy thinking is associated with pattern seeking, perceived threat, and intentional plot framing (Wood et al., 2012; van Prooijen and Douglas, 2017). Linguistically, this often manifests as oppositional discourse and in-group vs. out-group framing (Korenčić et al., 2024), which computationally distinguishes conspiratorial from critical narratives. These findings suggest that conspiratorial narratives exhibit recurrent semantic role structure rather than purely topical content, motivating approaches that explicitly model roles such as Actor, Victim, and Evidence.

Computational approaches. Prior work has incorporated psycholinguistic features such as LIWC-derived lexical categories (Tausczik and Pennebaker, 2010) and broader psycholinguistic profiles to identify conspiracy propagators (Giachanou et al., 2023), or fine-tuned neural models for document-level conspiracy classification (Liu et al., 2025). Hybrid systems combining emotional or moral framing features have also been explored (George et al., 2024). While such features provide indirect cues about epistemic stance, they do not explicitly represent narrative role relations.

To analyze these relations, span-based classification has been widely adopted in information extraction tasks such as coreference resolution and semantic role labeling, where enumerating candidate spans enables modeling of overlapping and variable-length structures (Lee et al., 2017).

3 System Overview

We address PsyCoMark with two independent RoBERTa-large (Liu et al., 2019) models, one for each subtask. Rather than incorporating external psycholinguistic features, our model learns role representations directly from the span-level supervision provided by PsyCoMark annotations. Subtask 1 is formulated as multi-label classification over enumerated candidate spans, with post-processing to resolve overlapping and nested predictions. Subtask 2 is formulated as standard sequence classification over the full document text. We train the two models separately and perform no parameter sharing or cross-task feature transfer; outputs are combined only to match the required submission format. Experiments were conducted on a single NVIDIA A100 GPU and a single RTX-4070 GPU using PyTorch 2.10 and Hugging Face Transformers 5.1.

3.1 Subtask 1: Conspiracy Marker Extraction

3.1.1 Architecture

A key challenge in PsyCoMark span extraction is precise boundary localization for semantically abstract, multi-word roles whose lengths vary substantially across contexts. We formulate conspiracy marker extraction as span classification rather than token-level tagging. Given a tokenized document, we enumerate all candidate spans up to a maximum length of $L = 32$ tokens (chosen for efficiency). Each candidate span (i, j) is independently classified into one or more semantic roles.

For each span (i, j), we construct a contextual representation v_{span} by concatenating six components:

$$v_{span} = [h_i; h_j; \bar{h}_{i:j}; w_{emb}; h_{i-1}; h_{j+1}]$$

where:

- $h_i, h_j \in R^H$: contextualized embeddings of the span start and end tokens.
- $\bar{h}_{i:j} \in R^H$: mean-pooled contextualized embedding over tokens within the span.

- $w_{emb} \in R^H$: learned embedding encoding span width (length). The embedding table size is $33 \times H$.
- $h_{i-1}, h_{j+1} \in R^H$: contextualized embeddings of the immediate left and right context tokens (zero-padded at boundaries).

With RoBERTa-large hidden size $H = 1024$, the concatenated representation has dimension $6H = 6144$. The span vector is passed through a two-layer MLP, producing logits for the five conspiracy roles:

$$6H \xrightarrow{\text{Linear}} H \xrightarrow{\text{ReLU}} \xrightarrow{\text{Dropout}(0.1)} \xrightarrow{\text{Linear}} 5$$

3.1.2 Training

Loss Function. We train the span classifier using binary cross-entropy with logits (BCE) for multi-label prediction. To address severe class imbalance (most candidate spans are negative), we apply per-role positive weighting:

$$\text{pos\_weight}_r = \frac{N_{neg,r}}{N_{pos,r}}$$

where $N_{neg,r}$ and $N_{pos,r}$ are the counts of negative and positive spans for role r, computed from the first 200 training batches to approximate class frequency without a full pass over the training set. We clip weights at 20.0 for stability.

Labeling Strategy. To enforce precise boundary learning, a candidate span is labeled positive for a role only if its Intersection-over-Union (IoU) with a gold span is ≥ 0.95. This high-overlap criterion encourages exact span boundaries rather than approximate matches.

Sampling. Enumerating all spans yields an overwhelming number of negative examples. We therefore use a three-tier sampling strategy to construct each training batch (up to 160 spans per document):

1. **Positives:** All gold-aligned spans.
2. **Hard negatives:** Spans with partial overlap (IoU $\in [0.50, 0.75)$) with gold spans.
3. **Random negatives:** Additional non-overlapping spans sampled uniformly.

Hard negatives introduce near-boundary confusions that improve span boundary discrimination.

3.1.3 Decoding

Span predictions are converted into final markers through three post-processing steps:

1. **Thresholding:** We apply per-role probability thresholds to sigmoid scores. For the submitted run, thresholds were tuned on the validation set via per-role grid search to maximize Macro F1 under *token*-level IoU ≥ 0.5 (matching the official evaluation).
2. **Containment-based NMS:** Overlapping spans are pruned using non-maximum suppression (NMS). A lower-scoring span B is suppressed by a higher-scoring span A if their containment ratio $\frac{|A \cap B|}{\min(|A|,|B|)}$ exceeds contain_thr, or if IoU$(A, B) \geq$ iou_thr. This removes redundant nested or highly overlapping predictions. We use role-specific thresholds (contain_thr 0.65–0.75, iou_thr 0.35–0.45) tuned on the validation set, with more permissive values for low-recall roles (Action, Effect, Evidence) to improve recall.
3. **Span merging:** After NMS, adjacent spans of the same role separated by a small character gap (≤ 3) are merged to form contiguous markers.

3.1.4 Experimental Details

Hyperparameters. We fine-tune RoBERTa-large with AdamW using a learning rate of 2×10^{-5}, weight decay 0.01, linear warmup over 10% of training steps, and cosine decay thereafter. The maximum sequence length is 512 tokens and the maximum span length is 32. Training uses batch size 2 for up to 14 epochs with early stopping (patience 5) based on validation decoded micro F1 (character IoU ≥ 0.3). We use random seed 42.

3.1.5 Results

On the validation set, our best span model achieved a decoded micro F1 of **0.6501** after thresholding and containment-based NMS, where a predicted span is counted correct if it matches a gold span with character-level IoU ≥ 0.3. Per-role validation performance is shown in Table 1.

Role	Precision	Recall	F1
Actor	0.745	0.798	0.771
Action	0.656	0.527	0.584
Effect	0.603	0.491	0.541
Evidence	0.652	0.504	0.569
Victim	0.744	0.644	0.691
Micro Avg	**0.692**	**0.613**	**0.650**

Table 1: Subtask 1 validation results (decoded spans; character IoU ≥ 0.3).

On the official test set, the shared task reports token-based overlap F1 with IoU ≥ 0.5. Under this evaluation, our submitted run achieved 0.2251 macro F1 and 0.2408 micro F1.

The difference between validation and official scores reflects both the stricter token-level IoU threshold and the greater sensitivity of macro F1 to low-frequency roles. Validation used IoU ≥ 0.3 during development to stabilize span boundary learning, while submission thresholds were tuned under IoU ≥ 0.5 to match the shared-task metric.

Role	Precision	Recall	F1
Actor	0.252	0.684	0.368
Action	0.106	0.331	0.160
Effect	0.098	0.256	0.142
Evidence	0.128	0.293	0.178
Victim	0.191	0.508	0.277
Macro F1		0.225	

Table 2: Subtask 1 official test results (token IoU ≥ 0.5).

3.2 Subtask 2: Conspiracy Detection

3.2.1 Architecture

We formulate Subtask 2 as document-level sequence classification. Each input document is encoded with RoBERTa-large, and the contextual representation of the `[CLS]` token is used as a fixed-length document embedding.

This representation is passed through a classification head consisting of a linear projection with Tanh activation and dropout, followed by a final linear layer producing logits for the three labels: *Yes*, *No*, and *Can't tell*. The RoBERTa encoder and classification head are jointly fine-tuned.

3.2.2 Training

Loss Function. We train the document classifier using cross-entropy loss over the three labels (*Yes*, *No*, *Can't tell*). To reduce overconfidence and improve generalization under class imbalance, we apply label smoothing with $\alpha = 0.05$.

Data Split. We use a stratified 90/10 train–validation split to preserve class distribution.

Optimization. The model is fine-tuned with AdamW using a linear learning-rate schedule with warmup. Training proceeds for up to 8 epochs with early stopping (patience 3) based on validation Weighted F1.

3.2.3 Experimental Details

Hyperparameters. We fine-tune RoBERTa-large with AdamW using a learning rate of 2×10^{-5}, weight decay 0.01, and linear warmup over 10% of training steps. The maximum sequence length is 512 tokens and label smoothing is 0.05. Training uses batch size 4 for up to 8 epochs with early stopping (patience 3).

3.2.4 Results

On the validation split, our best classifier achieved Weighted F1 0.6739 and Accuracy 0.6744. On the official test set, our submitted run achieved Weighted F1 0.7694 and Accuracy 0.7730. The official evaluation script excludes instances labeled "Can't tell," resulting in metrics computed over 608 matched samples. These scores are therefore not directly comparable to our three-class validation results. The higher test performance likely reflects both this label exclusion and differences in split difficulty.

4 Discussion

4.1 What Worked

Hard Negative Mining. Hard negatives (spans with 50–75% overlap with gold spans) proved critical for boundary learning. Without these near-miss examples, the model frequently predicted overly broad spans that contained the correct marker but included extraneous context. Hard negatives encouraged sharper span localization.

Subtask Independence. The document-level classifier in Subtask 2 receives only the raw input text and does not use predicted spans or role features from Subtask 1. Outputs from the two subtasks are combined only at submission time to match the shared-task output format. No information is exchanged between models during

training or inference. This design isolates span extraction and document classification performance and avoids cross-task error propagation.

Evaluation Alignment. Performance was highly sensitive to how closely development metrics matched the official evaluation protocol. For Subtask 1, early model selection used a relaxed character-level IoU criterion to stabilize boundary learning, whereas the shared-task evaluation uses token-level overlap with a stricter IoU threshold. This mismatch explains the validation–test gap and highlights the importance of calibrating span learning directly to the target metric. For Subtask 2, the official scoring excluded "Can't tell" instances, changing the effective evaluation distribution relative to three-class validation. Aligning validation procedures with the exact scoring setup proved as important as architectural choices.

4.2 Limitations

Error Analysis. Under the strict token IoU ≥ 0.5 evaluation metric, our model achieves 0.65 micro F1 on the validation split. The decrease to 0.225 macro F1 on the official test set indicates substantial distribution shift in the unseen data and the outsized impact of low-frequency abstract roles.

Qualitative error analysis reveals two dominant boundary failure modes. False positives frequently manifest as boundary drift, where the model captures extraneous syntactic modifiers (e.g., predicting `[April 5 statement]` instead of the precise gold *Evidence* annotation `[statement]`). Conversely, false negatives stem from substantial under-segmentation of compositional phrases (e.g., predicting just `["nuclear demolition".]` instead of the full *Action* clause `[spilled the beans on the Mossad operation involving "nuclear demolition".]`).

Abstract Role Boundaries. Roles such as *Action*, *Effect*, and *Evidence* exhibit greater semantic abstraction and variable span length than entity-like roles (*Actor*, *Victim*). This led to lower recall and boundary inconsistency, particularly for longer or compositional spans. The large drop from relaxed validation IoU (0.3) to official IoU (0.5) indicates substantial boundary sensitivity; future work should evaluate with stricter metrics during training to better align with shared-task evaluation.

Span Enumeration Constraints. Limiting candidate spans to length $L = 32$ improved computational efficiency but may exclude valid long markers, especially for multi-clause *Effect* or *Evidence* spans. The fixed maximum span length therefore imposes a recall ceiling.

4.3 Future Work

Joint Modeling of Subtasks. While independence avoided error propagation, well-implemented joint modeling could allow document classification to leverage predicted role structure. For example, aggregated *Actor* and *Victim* spans could provide explicit narrative signals for Subtask 2.

Structured Span Decoding. Current decoding relies on thresholding and NMS heuristics. Structured prediction approaches (e.g., span graphs or conditional decoding) could better enforce role consistency and reduce overlapping errors.

Role-Aware Encoding. Incorporating role-conditioned representations or span-type embeddings during encoding may improve discrimination of abstract roles and reduce boundary ambiguity.

Psycholinguistic and Lexical Features. While our current architecture relies exclusively on learned contextual embeddings, incorporating psycholinguistic markers, sentiment analysis, and topic modeling could help better disentangle conspiratorial framing from critical narrative.

5 Conclusion

Official evaluation summary. On the SemEval evaluation server, our system achieved 0.2251 macro F1 for span extraction and 0.7694 weighted F1 for document classification. For Subtask 1, span classification with hard-negative training and containment-based decoding achieved 0.6501 decoded micro F1 on validation (character IoU ≥ 0.3, internal development metric) and 0.2251 macro F1 on the official test. For Subtask 2, a standard RoBERTa-large classifier achieved 0.6739 weighted F1 on validation and 0.7694 on test without explicit role features. These results ranked 7th in Subtask 1 and 12th in Subtask 2 among participating systems, indicating that document-level conspiracy signals can be learned directly from raw text. Future work includes joint modeling of span roles and document classification to better capture narrative structure.

References

A George, M Ahrens, J Pierrehumbert, and M McMahon. 2024. Conspiracy detection beyond text: exploring the feasibility of adding psycho-linguistic features to enhance conspiracy detection models. In *Disinformation in Open Online Media*, Lecture Notes in Computer Science, pages 32–45. Springer.

Anastasia Giachanou, Bilal Ghanem, and Paolo Rosso. 2023. Detection of conspiracy propagators using psycho-linguistic characteristics. *Journal of Information Science*, 49(1):3–17.

Damir Korenčić, Berta Chulvi, Xavier Bonet Casals, Alejandro Toselli, Mariona Taulé, and Paolo Rosso. 2024. What distinguishes conspiracy from critical narratives? a computational analysis of oppositional discourse. *Computing Research Repository*.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 188–197, Copenhagen, Denmark. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Zhiwei Liu, Paul Thompson, Jiaqi Rong, and Sophia Ananiadou. 2025. Conspemollm-v2: A robust and stable model to detect sentiment-transformed conspiracy theories. In *ECAI 2025*. IOS Press.

Mattia Samory, Felix Soldner, and Veronika Batzdorfer. 2026. SemEval-2026 task 10: PsyCoMark – psycholinguistic conspiracy marker extraction and detection. In *Proceedings of the 20th International Workshop on Semantic Evaluation (SemEval-2026)*, San Diego, United States. Association for Computational Linguistics.

Yla R. Tausczik and James W. Pennebaker. 2010. The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of Language and Social Psychology*, 29(1):24–54.

Jan-Willem van Prooijen and Karen M. Douglas. 2017. Conspiracy theories as part of history: The role of societal crisis situations. *Memory Studies*, 10(3):323–333.

Michael J. Wood, Karen M. Douglas, and Robbie M. Sutton. 2012. Dead and alive: Beliefs in contradictory conspiracy theories. *Social Psychological and Personality Science*, 3(6):767–773.