BiasChain: A Multi-Agent LLM Framework for Justified Peer Review Bias Detection

Supplementary Material

Anonymous ACL Submission

Table of Contents

---

1. Dataset Analysis

1.1 Dataset Overview

We utilized the publicly available `oaimli/PeerSum` dataset, specifically the train split containing over 14,000 academic papers with associated peer reviews. Due to computational constraints, we created a random subset of 200 papers, each with at least three reviews, yielding approximately 1,200+ reviews for analysis.

Extracted Fields: - `paper_id`: Unique identifier for each paper - `paper_title`: Title of the academic paper - `paper_abstract`: Abstract of the paper - `review_id`: Unique identifier for each review (filtered for 'official_reviewer' role only) - `review_contents`: Full text content of the peer review

1.2 Reviewer Disagreement Analysis

Reviewer Disagreements Distribution: The histogram analysis of rating variances among reviewers reveals a right-skewed distribution with a peak around variance of 10. This indicates that while most reviews exhibit moderate disagreement, some show extreme differences. A small peak at zero variance indicates cases of complete reviewer consensus.

Key Findings: - Most papers have moderate reviewer disagreement (variance ~10) - Small subset shows complete consensus (variance = 0) - Long tail of high-disagreement cases suggests systematic bias issues

1.3 Review Characteristics

Review Length vs. Confidence: Box plot analysis shows that lower confidence levels (-1, 1) exhibit higher variance in review lengths. Higher confidence levels (3-5) demonstrate more consistent review lengths, suggesting structured reviewing approaches.

Ratings vs. Confidence Correlation: Heatmap analysis reveals concentration around middle confidence levels, with higher ratings generally correlating with higher confidence. Lower confidence ratings appear sporadic across various review scores, indicating potential calibration inconsistencies.

---

2. Implementation Details

2.1 Model Configuration

Primary Model: `gemini-2.0-flash-thinking` **Key Parameters:** - Temperature: 0 (deterministic generation) - Model selection rationale: Robust reasoning capabilities and effectiveness in complex textual inference tasks

2.2 Computational Requirements

- **Dataset Size:** 200 papers, ~1,200 reviews
- **Processing Pipeline:** Sequential analysis through 4 specialized stages
- **Output Format:** Structured JSON schemas for each analysis stage

---

3. Prompt Templates

3.1 Sentiment and Tone Analysis

Schema Definition:

```
class JsonSchema(BaseModel):
    sentiment: Literal['Positive', 'Negative', 'Neutral'] =
        Field(description="The sentiment of the review")
    sentiment_reason: str = Field(description="A concise
        explanation justifying the detected sentiment based on
        the review's language, tone, and choice of words")
    tone: Literal['Formal', 'Informal', 'Neutral', 'Supportive', 'Critical
        = Field(description="The tone of the review")
    tone_reason: str = Field(description="An explanation of the
        detected tone using specific words, phrases, or stylistic
        features")
```

Prompt Template:

```
You are an expert in analyzing peer review text.
Analyze the following peer review and classify it according to
the following schema:
{json_schema}

The review starts here:
{text}
```

3.2 Internal Consistency Analysis

Schema Definition:

```python
class JsonSchema(BaseModel):
    consistency: Literal['Yes', 'No'] =
        Field(description="Is the review consistent in itself")
    consistency_reason: str = Field(description="The reason for
        the classified consistency")
```

Prompt Template:

```
You are an expert in analyzing peer review texts.
This time you have to check for the consistency of the review.
Also, check whether the review contradicts itself.
If no proper review, tell that.
Classify according to the following schema:
{json_schema}

The review starts here:
{review}
```

3.3 Inter-Review Comparison

Schema Definition:

```python
class JsonSchema(BaseModel):
    is_consistent_with_others: Literal['True', 'False'] =
        Field(description="Check whether the current review is
        consistent with other reviews")
    alignment_score: float = Field(ge=0, le=10,
        description="Alignment score of the current review with
        other reviews")
    contradictory_points: str = Field(description="List down
        contradictory points between the current review and other
        reviews, if any, else write 'No contradictory points'")
```

```
    possible_bias_flags: str = Field(description="List down the
        reasons for the current review being biased, if any, else
        write 'No possible bias flags'")
    summary_of_differences: str = Field(description="List down
        the summary of differences between current review and
        other reviews, if any, else write 'No differences
        found'")
```

Prompt Template:

```
You are an expert meta-reviewer.
You are provided with paper title, abstract and two sets of
reviews.
One set contains only one review and other set contains the list
of reviews.
Your main goal is to compare this one review with list of the
other reviews and give the output in following schema:
{json_schema}

Here is the data:
Paper Title: {paper_title}
Paper Abstract: {paper_abstract}
One Review: {main_review}
Other Reviews: {other_reviews}
```

3.4 Specialized Bias Detection Prompts

3.4.1 Novelty Bias Detection

```
You are an expert in **novelty bias detection** in peer review
texts.
Your task is to analyze the given peer review text and identify
signs of bias related to the **overemphasis on novelty** over
practical or theoretical contribution.
Your bias tool name is: Novelty Bias

Bias in this context may include:
- Excessive praise or criticism based solely on how "novel" the
approach is
- Devaluation of incremental research or well-established
methodologies
- Assumptions that novelty automatically equates to higher
quality

You must return your findings in **strict adherence** to the
following schema:
```

```
{child_json_schema}

The input starts here:
{input}
```

3.4.2 Methodology Bias Detection

```
You are an expert in **methodology preference bias detection** in
peer review texts.
Your task is to evaluate whether the reviewer shows unjustified
bias toward or against specific **methodologies or paradigms**.
Your bias tool name is: Methodology Bias

Bias in this context may include:
- Preference for specific frameworks, techniques, or tools
regardless of their objective fit
- Dismissal of qualitative or alternative approaches in favor of
dominant quantitative ones (or vice versa)
- Favoritism for trendy or mainstream methods without strong
justification

You must return your findings in **strict adherence** to the
following schema:
{child_json_schema}

The input starts here:
{input}
```

3.4.3 Confirmation Bias Detection

```
You are an expert in **confirmation bias detection** in peer
review texts.
Your task is to identify whether the review shows **preference
for findings or approaches that align with the reviewer's own
beliefs, work, or assumptions**.
Your bias tool name is: Confirmation Bias

Bias in this context may include:
- Favorable evaluation of papers that support the reviewer's
previous work or perspectives
- Dismissal or skepticism toward alternative frameworks without
objective critique
- Implicit reinforcement of the status quo or widely accepted
theories without open-minded evaluation
```

```
You must return your findings in **strict adherence** to the
following schema:
{child_json_schema}

The input starts here:
{input}
```

3.4.4 Positive Results Bias Detection

```
You are an expert in **publication bias detection**, particularly
focused on the **favoring of positive or significant results** in
peer review texts.
Your task is to identify whether the reviewer shows bias toward
outcome-based evaluation, especially overvaluing positive,
breakthrough, or state-of-the-art results.
Your bias tool name is: Positive Results Bias

Bias in this context may include:
- Disregard or undervaluation of null, negative, or replication
studies
- Language implying that positive results are inherently more
valuable or publishable
- Inflated praise for significant results without addressing
methodological soundness

You must return your findings in **strict adherence** to the
following schema:
{child_json_schema}

The input starts here:
{input}
```

3.4.5 Linguistic Bias Detection

```
You are an expert in detecting **linguistic proficiency bias** in
peer review texts.
Your task is to assess whether the reviewer penalizes the
author's language use in ways that unfairly affect scientific
evaluation.
Your bias tool name is: Linguistic Bias

Bias in this context may include:
- Overemphasis on grammar, fluency, or native-sounding English
```

```
- Equating writing quality with research quality, especially for
non-native authors
- Non-constructive feedback focused on language rather than
content clarity

You must return your findings in **strict adherence** to the
following schema:
{child_json_schema}

The input starts here:
{input}
```

3.4.6 Parent Agent Schema

```python
class ParentJsonSchema(BaseModel):
    bias_detected: Literal['True', 'False'] =
        Field(description="Is the bias detected?")
    bias_type: List[str] = Field(description="Types of biases
        detected from the provided tools strictly as returned by
        tools, if no bias from tools, write 'None'")
    confindence_score: float = Field(description="How sure are
        you about it overall?")
    evidence: List[str] = Field(description="In what statement
        did you find bias, if bias is detected else write 'None'
        if no bias found overall")
    suggestion_for_improvements: List[str] =
        Field(description="Give some improvement suggestions if
        bias is detected, else write 'None' if no bias found
        overall")
```

3.4.7 Child Agent Schema

```python
class ChildJsonSchema(BaseModel):
    bias_detected: Literal['True', 'False'] = Field("Is the bias
        detected?")
    bias_type: Literal['Novelty Bias', 'Confirmation Bias',
        'Methodology Bias', 'Positive Results Bias', 'Linguistic
        Bias', 'None'] =
        Field(description="Name of the bias only if it is
        detected, else None")
    confindence_score: float = Field(ge=0, le=10,
        description="How sure are you about it?")
```

```
    evidence: str= Field(description="In what statement did you
        find bias, if bias is detected else write 'None'")
    suggestion_for_improvements: str= Field(description="Give
        some improvement suggestions if bias is detected, else
        write 'None'")
```

PS: For a version of these prompts with syntax highlighting and improved formatting, please refer to the included prompts.html file.

---

4. Annotation Guidelines

4.1 Pipeline Overview

The BiasChain framework implements a four-stage sequential analysis pipeline:

1. **Tone and Sentiment Analysis:** Analyzes emotional and stylistic characteristics
2. **Internal Consistency Check:** Evaluates logical coherence within reviews
3. **Inter-Review Comparison:** Compares target review against peer reviews
4. **Bias Detection:** Synthesizes all prior analyses for bias classification

4.2 Module Specifications

4.2.1 Tone and Sentiment Analysis

- **Input:** Single review text
- **Output:**
  - Sentiment: [‘Positive’, ‘Negative’, ‘Neutral’]
  - Sentiment Reason: Justification for classification
  - Tone: [‘Formal’, ‘Informal’, ‘Neutral’, ‘Supportive’, ‘Critical’, ‘Balanced’]
  - Tone Reason: Explanation using specific textual features

4.2.2 Internal Consistency Check

- **Input:** Single review text
- **Output:**
  - Consistency: Yes/No binary classification
  - Consistency Reason: Explanation for classification

4.2.3 Inter-Review Comparison

- **Input:** Paper title, abstract, target review, other reviews for same paper
- **Output:**
  - Is consistent with others: Yes/No

    - Alignment Score: 0-10 scale
    - Contradictory Points: Specific disagreements identified
    - Possible Bias Flags: Potential bias indicators
    - Summary of Differences: Comprehensive comparison

4.2.4 Bias Detection

- **Input:** All previous module outputs plus paper metadata
- **Output:**
    - Bias Detected: True/False
    - Bias Types: From 5 predefined categories or None
    - Confidence Score: 0-10 scale
    - Evidence: Specific textual evidence
    - Suggestions for Improvements: Actionable recommendations

4.3 Bias Type Definitions

1. **Novelty Bias:** Overemphasis on novelty over practical/theoretical contribution
2. **Methodology Bias:** Unjustified preference for specific methodologies
3. **Confirmation Bias:** Preference for findings aligning with reviewer's beliefs
4. **Positive Results Bias:** Favoring positive/significant results over null findings
5. **Linguistic Bias:** Penalizing language quality affecting scientific evaluation

4.4 Example Annotation

Paper Title: "Efficient Deep Learning for Edge Devices"

Target Review: "This paper lacks significant novelty. While edge deployment is relevant, the techniques used are standard. The writing is clear, and the results are unimpressive. I don't see this making a meaningful contribution."

Other Reviews: [Two positive reviews acknowledging practical contribution and integration novelty]

Pipeline Results: - **Sentiment:** Negative (reason: "lacks significant novelty", "unimpressive results") - **Tone:** Critical (reason: dismissive language) - **Consistency:** Yes (maintains consistent critical viewpoint) - **Inter-Review Alignment:** No, Score: 3 (significantly harsher than peers) - **Bias Detection:** True, Type: Novelty Bias, Confidence: 7

Conclusion

This supplementary material provides comprehensive documentation of the BiasChain framework implementation, including detailed prompt templates, schema definitions, annotation guidelines, and additional analysis results. All

source code, data preprocessing scripts, and additional materials are available in the anonymous GitHub repository for reproducibility and future research.