

# Topic Modeling Based Code Review Hotspot Prediction

10/29/25

Tongwei Zhang  
Zhaoyi Ge  
Zhiao Wei



## Intro: Predicting Review Hotspots in Agentic Pull Requests

- PRs increasingly include **AI coding agents** (Copilot, Cursor, Claude Code).
- Reviewers face a simple, costly question: **“Where should I look first?”**
- Our focus: given a PR’s title, description, and diff, predict the files most likely to attract review comments (the “hotspots”).
- Empirical foundation: **AIDev** (agent-authored PRs with diffs, comments, follow-up edits; we’ll report the exact snapshot/DOI).

Problem Statement and Motivation

PAGE 2



## Why this matters and what’s missing?

- **Practical need:**
  - Code review is time-boxed;
  - Triage is hard, especially for agent-produced, multi-module patches with uneven rationale (Bacchelli & Bird, ICSE’13).
- **Missing pieces:** No reproducible **file-level hotspot** construct; few transparent baselines beyond churn; agent patches shift distributions.
- **Goal:**
  - **Efficiency:** Reduce time-to-first-useful-comment in large PRs (Bacchelli & Bird, ICSE’13).
  - **Safety net:** Lower risk of missing consequential files in multi-module edits.
  - **Human-in-the-loop:** Provide an **attention map**, not an automated decision.

Problem Statement and Motivation

PAGE 3



## Methodology: Baseline

- What kind of evidence suggest a file is a hotspot?
  - Review comments (in-line comments, file comments)
  - Follow-up commits
- Evaluation Metric
  - Hit@k, Precision@k, Recall@k, nDCG@k
    - measure how well a system ranks relevant items among its top-k results

Methodology and Schedule

PAGE 4



## Methodology: Pipeline

- A per-repository hotspot prediction pipeline
- For all historic PRs:
  - Create a unified "document" that represents the complete discussion surrounding a PR.
  - Run topic modelling on the document.
  - In the meantime, extract the modified files of the PR
  - Link the topic distribution back to the files.
- Result: An **attention map** which can be used to predict review hotspots for each file.
- Then use the attention map to predict new PR hotspots.

## Datasets and Tools

- AIDev Dataset:
  - Has everything we need: PR's title, description, reviews, and changed file
  - 7122 Agentic PRs and 6628 human PRs from popular repositories
- Topic Modelling
  - **BERTopic**: a modern **topic modeling** technique that uses **transformer-based embeddings** to find and describe topics in a collection of documents.

## Research Questions

### 1. How can we define review hotspots in agentic PRs?

Operationalize using review comments + following commits

### 2. What are the characteristics of review hotspots in agentic PRs?

Analyze patterns across files, code complexity and PR structure.

### 3. How accurately can we predict review hotspots from PR descriptions and diffs?

Evaluate with Hit@k, Precision@k, Recall@k, nDCG@k.

### 4. What topics drive hotspot predictions?

Identify key themes (auth, testing, etc.)

## Threats to Validity and Mitigation Strategies

### 1. Internal Validity: Definition of Review Hotspots.

#### Problem:

Comments ≠ always important  
Critical files may have few comments

#### Mitigation:

✓ Inter-rater agreement check  
✓ Case studies

### 2. External Validity: Dataset Representativeness.

#### Problem:

AIDev may not represent all workflows

#### Mitigation:

✓ Detailed dataset statistics  
✓ Stratified evaluation  
✓ Explicit scope of conclusions

## Conclusion & Any ideas?

- **Problem:** Reviewers struggle to triage multi-file Agentic-PRs. We ask: which changed files will attract review comments first?
- **Contributions**
  - A **reusable baseline** with clear **ground truth** and metrics (Hit@k, Precision@k, Recall@k, nDCG@k) plus open scripts.
  - A **pipeline** that applies topic modeling to predict attention-needed files for each Agentic-PR.
- **Alignment**
  - Fits MSR: topic-driven prediction, large-sample empirical study on AIDev.
- **Any Questions or advice?**

UNIVERSITY OF  
WATERLOO



Thanks!