Associating Natural Language Comment and Source Code Entities

Sheena Panthaplackel¹(spantha@cs.utexas.edu), Milos Gligoric², Raymond J. Mooney¹, Junyi Jessy Li³
Department of Computer Science¹, Department of Electrical and Computer Engineering², Department of Linguistics³
The University of Texas at Austin

Problem

Motivation: Learning to relate comment and code elements is critical to automated systems for generating comments and code and detecting inconsistent comments and code.

Task: Given a noun phrase (NP) in a comment, we classify the relationship between the NP and each candidate code token in the corresponding method as either associated or not associated.

Features + Model

- Independent classification per candidate code token
- Feature vector captures:
  - Code structure: grammar and API relevant to Java
  - Comment and code context: averaged pre-trained embeddings
  - Relationship between comment and code: cosine similarity between comment and code representations
- Lexical characteristics: overlap with NP and return statement tokens

Noisy Supervision

We propose obtaining noisy labels for this task by isolating parts of the comment and code that are edited at the same time, using GitHub’s commit history feature.

Primary Dataset

NP: Added NP in comment
Code token labels:
+ Added tokens
- Unchanged tokens (unlikely associated with NP that did not exist before)

Deletions Dataset

NP: Deleted NP in comment
Code token labels:
+ Deleted tokens
- Unchanged tokens (could be associated with NP that did exist before)

Conclusions:
- Learning from noisy supervision, our system outperforms multiple baselines
- “More noisy” data provides a valuable training signal, improving the scope for collecting data and training models

Results

Comparing to rule-based baselines

Augmenting training with varying numbers of "more noisy" examples from deletions