Beyond the Exact Match

Investigating the Relationship Between Syntactic and Semantic Equivalence in Human and LLM Test Assertions.

Ruben van der Giessen. | Supervisors: Mitchell Olsthoorn, Annibale Panichella.

Problem

Writing test assertions is expensive and takes time. While LLMs can be used to reduce this burden, current evaluation methods are lacking (exact match or bugs).

No research currently investigates the relationship between the syntactic and semantic equivalence between LLM-generated and human-written reference assertions.

Background

- Molinelli et al. (2025): LLM-generated assertions rival those of humans in mutation scores.
- Primbs et al. (2025): Made a fine-tuned model that achieved a 59.5% exact match rate, but caught few real-word bugs.

Research Gap: No clear idea of when syntactic differences lead to “worse” assertions, instead of being equivalent.

Research Questions

To what extent do LLM-generated assertions differ syntactically but remain semantically equivalent to reference assertions?

Subquestions:

1. What is the correlation between syntactic and semantic equivalence?
2. What types of syntactic transformations does the LLM often apply, while preserving semantic equivalence?
3. At which "thresholds" for syntactic equivalence do the LLM-generated assertions generally fail to be semantically equivalent?

Results

1. Based on filtered dataset with 177 datapoints.
2. Moderately strong correlation (ρ = 0.685) between syntactic and semantic similarity. While meaningfully connected, one cannot consistently predict the other.
3. LLM prefers to omit assertion message and use equality assertions over boolean checks. Might indicate a preference for simple and informative assertions.
4. A threshold of 0.41 for syntactic similarity leads to a median semantic similarity of 0.5712 below and 1.000 above the threshold. Standard deviation above the threshold is half as high as below the threshold.

Methodology

1. Create dataset
2. Generate LLM assertions
3. Calculate syntactic and semantic similarity. Approximate semantic similarity through similarity of killed mutants in mutation testing.
4. Analyse datapoints to find patterns
   a. RQ1: Calculate correlation coefficient.
   b. RQ2: Open coding on syntactically different yet semantically equivalent entries.
   c. RQ3: Decision tree with depth of 1 to determine split.