

RQ1 Results

RQ1 — How do community smells evolve over software releases?

This research question aims to explore the evolution of community smells over time, across different releases of a software system. To answer this question, we divided to three subqutions as follows:

RQ1-A: To what extent are community smells subject to change?

To answer this research question, we employ quantitative analysis and apply statistical metrics to illustrate how community smells change across software releases.

RQ1-B: Which community smells lead to a higher frequency of changes across the releases?

This sub-question is addressed through a quantitative investigation focused on identifying which types of community smells are most frequently associated with changes. We compute evolution metrics for community smells, including their presence over time (in terms of both the number of releases and number of days). Additionally, we calculate a smell presence metric to quantify how persistently each smell type appears.

RQ1-C: Why Might Short Releases Have More Community Smells Than Long Releases?

To explore this sub-question, we conduct a comparative statistical analysis between short and long release cycles. We examine whether short cycles exhibit a higher concentration of community smells and explore potential reasons for this phenomenon. In particular, we analyze developer sentiment, considering that shorter cycles often involve tight deadlines and prioritization of speed and delivery over collaboration, coordination, and overall process quality.

Results:

RQ1-A: RQ1-A: To what extent are community smells subject to change?

Project name	Total number of releases	Total number of CS	Mean of CS per release	Median of CS per release	Stdev of CS per release
Transformers	134	406	3.03	3	0.9
Pytorch	50	112	2.42	2	0.92
Shardingspher	29	57	1.97	2	1.15
Netdata	76	185	2.43	3	1.27

The analysis reveals that community smells (CS) are present across all observed software projects and exhibit varying degrees of fluctuation over time.

RQ1-B: Which community smells lead to a higher frequency of changes across the releases?

A quantitative investigation: We compute the evolution metric of community smells in terms of number of releases and number of days. We compute the smell presence metric precisely as follows:

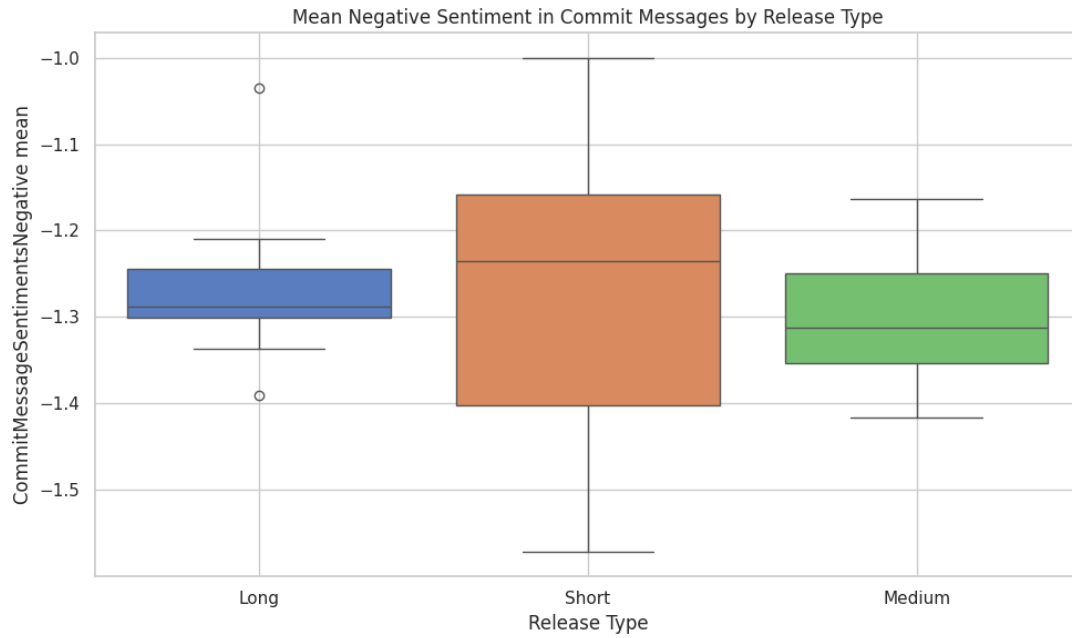
$$\text{Community_smell_presence metric} = \frac{\# \text{ Total CS Release Duration (days)}}{\text{Project Cycle Duration (days)}} + \frac{\# \text{ Total CS releases}}{\text{total number of releases}}$$

Project Name	Smell	Total Smell Duration	Total Project Duration	Releases with Smell	Total Releases	Smell Evolution Metric
Transformers	OSE	309	1848	72	134	0.704521225
	BCE	1387	1848	107	134	1.549048588
	PDE	1448	1848	106	134	1.57459456
	RS	1273	1848	121	134	1.591837888
Pytorch	OSE	2	2768	1	51	0.0203
	BCE	2067	2768	41	51	1.55067
	PDE	2409	2768	45	51	1.752656
	RS	1231	2768	28	51	0.993745
Shardingsphere	OSE	33	2282	3	29	0.117909
	BCE	1081	2282	17	29	1.059914
	PDE	2021	2282	27	29	1.816661
	RS	225	2282	10	29	0.443425
Netdata	OSE	210	3072	26	76	0.410465
	BCE	2191	3072	61	76	1.515848
	PDE	1591	3072	51	76	1.188956
	RS	731	3072	47	76	0.856377

RQ1-C: Why Might Short Releases Have More Community Smells Than Long Releases?

Netdata project as example:

Metrics	Short releases	Long releases
Total number of releases	23	10
Total number of CS	59	12
Mean of CS per release	2.57	1.2
Median of CS per release	3	1
Stdev of CS per release	0.79	0.42



The findings support the hypothesis that shorter release cycles may be linked to higher emotional stress, possibly due to tight deadlines and limited time for coordination. This emotional volatility could contribute to the higher occurrence of community smells in short releases.