Multi-Label Classification for Requirement Smell Detection in Natural-Language Requirements

Karolis Trinkūnas, Jolanta Miliauskaitė

Vilnius Gediminas Technical University, Saulėtekio al. 11, Vilnius
karolis.trinkunas@stud.viniustech.lt, jolanta.miliauskaite@vilniustech.lt

Abstract. Natural-language software requirements often contain quality defects such as ambiguity, vagueness, subjectivity, and nonverifiability. This paper presents a multi-label approach for detecting five requirement smell categories: Subjective, Ambiguous, Nonverifiable, Negative, and Vague. The method adapts SetFit with a SentenceTransformer encoder, a weighted binary-relevance logistic-regression head, hint-aware augmentation, and label-specific threshold tuning. Experiments on harmonized datasets show that the approach supports automated requirement quality analysis and interpretable smell detection.

Keywords: requirements engineering, requirement smells, requirement quality, multi-label classification, natural language processing, SetFit.

1 Introduction

Software requirements define expected system behavior and guide subsequent development activities. When written imprecisely, they can introduce defects that propagate into design, implementation, and testing, increasing cost and reducing system quality [1], [2]. This is especially relevant for natural-language requirements, which remain widely used in practice but are prone to ambiguity, vagueness, and other linguistic quality issues [1].

Requirement smells have been proposed as linguistic indicators of potential defects in requirement statements [2]. More recent work has applied NLP and machine learning to automate smell detection, including multi-label formulations that account for multiple co-occurring issues in a single requirement [3], [4], [5].

Existing work has focused mainly on rule-based NLP and heavier deep-learning approaches, whereas this study explores a lightweight SetFit-based approach to multi-label requirement smell detection with harmonized multi-source data preparation, hint-aware augmentation, and label-specific threshold tuning [6].

DOI: https://doi.org/10.15388/LMITT.2026.32

2 Background and related work

Natural-language requirements are widely used but remain vulnerable to defects such as ambiguity, vagueness, subjectivity, and nonverifiability [1], [2]. Prior work introduced requirement smells as linguistic indicators of potential quality problems in requirement statements [2].

Early automated approaches relied mainly on dictionaries, part-of-speech tagging, and other lightweight NLP techniques to detect recurring classes of problematic language [2]. Related studies also examined ambiguity detection more specifically using dictionary- and knowledge-based methods [7], [8]. However, such approaches are often sensitive to wording variation, domain context, and predefined rules, which limits generalization across heterogeneous requirements datasets [4].

More recent work has explored machine learning and deep learning for automated requirement smell detection [3], [4]. Because a single requirement may exhibit more than one smell at the same time, the task is naturally formulated as a multi-label classification problem [3], [4]. Although prior studies show measurable progress, differences in smell taxonomies, datasets, and evaluation protocols limit direct comparison with the present work. Table 1 summarizes representative requirement smell detection studies and their reported evaluation metrics. In addition, the present work is informed by broader methodological literature on data harmonization, text augmentation, and imbalanced classification [9], [10], [11].

Table 1. Representative requirement smell detection studies with reported evaluation metrics.

Author(s), Year	Work	Method / Focus	Reported metrics	Difference from this work
Femmer et al. [2], 2017	*Rapid quality assurance with Requirements Smells*	Rule-based / NLP-assisted smell detection with the Smella prototype	P/R: 59% / 82%	Rule-based; not multi-label or transformer-based
Habib et al. [4], 2021	*Detecting Requirements Smells With Deep Learning: Experiences, Challenges and Future Work*	Exploratory multi-class multi-label smell detection with ML baselines	MLP P/R/ F1: 0.87 / 0.79 / 0.83; SVM P/R/ F1: 0.89 / 0.81 / 0.84	Early multi-label study; overfitting and dataset limitations

Author(s), Year	Work	Method / Focus	Reported metrics	Difference from this work
Veizaga, Shin, and Briand [3], 2024	*Automated Smell Detection and Recommendation in Natural Language Requirements*	Smell detection with recommendation support	P/R: 89% / 89%	Strong industrial baseline; not centered on lightweight SetFit-based multi-label classification
Alem et al. [5], 2025	*Multi-label software requirement smells classification using deep learning*	Multi-label deep learning with LSTM, Bi-LSTM, and GRU using ELMo and Word2Vec	Best Macro-F1: 90.3%	Closest task match, but uses heavier recurrent models

3 Proposed method

Requirement smell detection is formulated as a five-label multi-label text classification task because a single requirement may contain multiple quality issues [3], [4], [5]. SetFit was selected because it is intended for efficient text classification under limited labeled data conditions [6]. Prior work on requirement smell detection has also reported dataset limitations and overfitting challenges in learned approaches [4]. In the present study, these challenges were addressed through data harmonization and imbalance-aware preparation. Compared with purely rule-based methods [2], learned approaches offer greater flexibility for capturing contextual and semantic information [3], [4], and SetFit provides a lightweight and practical choice for the present setting. Figure 1 shows the overall workflow of the proposed SetFit-based multi-label requirement smell detection pipeline, from text preprocessing and semantic encoding to hint extraction, augmentation, multi-label prediction, and label-specific thresholding.

Fig. 1. Process model for SetFit-based multi-label requirement smell detection.

The method relies on harmonized data preparation, label-consistent dictionaries, and lexical hints to support robust multi-label training. The classifier adapts SetFit [6] for multi-label requirement smell detection by replacing the standard single-label setup with a binary-relevance formulation,

in which one weighted logistic-regression classifier is trained for each smell category. Requirements are encoded with a SentenceTransformer model, and label-specific decision thresholds are tuned on the validation split. In addition, hint-aware augmentation is used to strengthen category-relevant lexical cues during training. This design was informed by prior work on requirement smell detection and lexical quality indicators [4], [12]. Dictionary refinement was used during harmonization, whereas final comparisons focused on dataset variants and augmentation settings.

4 Data preparation and training decisions

The dataset was prepared for multi-label requirement smell classification over five categories: Subjective, Ambiguous, Nonverifiable, Negative, and Vague. Because DS1–DS4 differ in labels, annotation conventions, lexical coverage, and structure, dataset preparation was treated as a harmonization task rather than simple concatenation, enabling a unified multi-label corpus for training. This use of harmonization follows broader methodological guidance on integrating heterogeneous data sources [9].

As shown in Fig. 2, dataset preparation began with DS1–DS4 and the original ARTA SmellyWordsDictionary [12]. Column names were aligned, low-frequency quality-type columns were removed, requirement texts were cleaned, and dictionary-based label alignment was applied. Inputs were limited to 32 words as a pragmatic, validation-guided preprocessing heuristic to reduce contextual variation while remaining close to the average requirement length in the merged datasets. Because the retained categories were imbalanced, imbalance-aware learning was used. A balanced dataset variant with a 60/40 smell-to-no-smell ratio was also evaluated as an exploratory sensitivity analysis of reduced class skew. Because balancing can affect the interpretation of imbalanced-class evaluation [11], the balanced variant should therefore be interpreted as an exploratory analysis rather than as the primary unbiased benchmark.

Dictionary refinement resolved label inconsistencies and expanded category coverage. Across the four datasets, overlapping words were reviewed, existing assignments were retained where possible, missing categories were inferred, and each word was assigned to a single canonical category. This harmonization improved lexical comparability across partially inconsistent source annotations [9]. The resulting changes are shown in Table 2 and the final distribution of defect categories across the three splits is shown in Table 3.

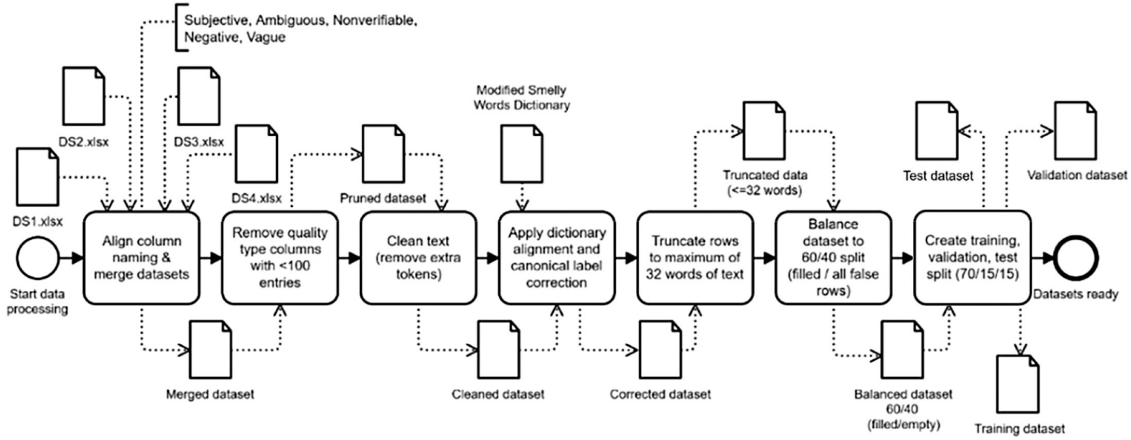

Fig. 2. Data preparation workflow for multi-label requirement smell classification.

Table 2. Smells dictionary label modifications.

New category	Words moved
Subjective	Likely, sufficient, typical, appropriate, reasonable
Ambiguous	Clear, possible, consistent, relevant, practical
Nonverifiable	Accurate, approximate, efficient, normal, reliable
Vague	Considerable, adequate
Negative	-

Table 3. Defect category distribution across training, validation, and test splits.

Defect Category	Train	Validation	Test	Total
Subjective	264 (80 unique)	51 (36 unique)	53 (41 unique)	368 (157 unique)
Ambiguous	345 (140 unique)	76 (58 unique)	82 (59 unique)	503 (257 unique)
Nonverifiable	54 (21 unique)	17 (10 unique)	11 (7 unique)	82 (38 unique)
Negative	97 (4 unique)	15 (4 unique)	21 (2 unique)	133 (10 unique)
Vague	113 (18 unique)	22 (7 unique)	35 (9 unique)	170 (34 unique)

As shown in Table 3, the merged dataset is imbalanced across the five defect categories. Ambiguous and Subjective occur much more frequently than Nonverifiable, while Negative has very few unique instances despite a moderate number of total occurrences. This imbalance motivated the use of weighted learning during training. A balanced dataset variant was also evaluated to adjust the smell/no-smell ratio, although label-level imbalance across the five smell categories remained.

To improve training quality, the training split used linguistic hints derived from requirement text. These hints consisted of words or short phrases associated with potential smells and were used to strengthen the relationship between wording patterns and target labels. Validation and test splits were left unchanged for evaluation.

The training dataset was further extended through label-specific hint-aware augmentation using counterfactual and hint-only variants. In the reported comparison runs, augmentation was enabled only for Subjective, Ambiguous, and Nonverifiable, because these labels offered the clearest opportunity for label-preserving lexical transformations under limited-data conditions. This choice follows broader findings from text-classification research that augmentation is most useful in low-data regimes when transformations preserve label semantics [10]. Here, augmentation was

restricted to lexical operations closely tied to the smell categories so that generated samples would reinforce category-relevant cues without introducing semantically incompatible variation. The training process followed the SetFit paradigm, where sentence embeddings are learned and then used by a lightweight classifier [6]. Compared with larger end-to-end transformer pipelines, this setup offers a practical trade-off between representational quality and computational complexity [6]. After model fitting, decision thresholds were tuned separately for each label on the validation split rather than using a single shared threshold.

In the original source dataset CSV files (DS1–DS4), each row contained the requirement text and separate category-specific columns. During preprocessing, these columns were parsed into token lists. A binary multi-label vector was then constructed by assigning a value of 1 to a category when at least one hint token was present and 0 otherwise. The parsed tokens were also stored as label-specific hint lists for later dictionary enrichment and augmentation.

The classifier used the sentence-transformers/all-MiniLM-L6-v2 encoder and a custom weighted binary-relevance logistic regression head. To determine the most effective training setup, several combinations of hyper-parameters were evaluated, with model selection based on the highest validation F1 scores. The explored settings included different values for SetFit iterations, training epochs, batch size, learning rate, zero-row weight, and label-specific augmentation sizes. Augmentation settings were varied through counterfactual-removal and hint-only transformations for selected smell categories. The final reported configuration, selected on the validation split and then assessed on the held-out test split, used one epoch, 100 iterations, batch size 32, learning rate 2×10^(-5), balanced class weights, and a zero-row weight of 0.5. Hint-aware augmentation was enabled for Subjective, Ambiguous, and Nonverifiable, with both counterfactual-removal and hint-only multipliers set to 1. Label-specific decision thresholds were then tuned on the validation split by maximizing per-label F1. Training was performed on an ASUS TUF 5090 OC GPU.

5 Evaluation results and linguistic interpretation

The proposed classifier was evaluated on the merged and balanced dataset variants, each with and without hint-only augmentation. Here, "val" denotes the validation split used for model selection and comparison of

configurations, whereas “test” denotes the held-out test split used only for the final evaluation of the best-performing configuration. Validation results are reported for both dataset variants, while the final test result is reported for the best-performing balanced configuration. Evaluation uses subset accuracy, micro-F1, macro-F1, hamming accuracy, and smell/no-smell F1.

Table 4. Comparative evaluation results across dataset variants and hint-only augmentation settings.

Dataset/split	Tuned	Hint-only aug	Subset Acc	Micro-F1	Macro-F1	Hamming Acc	Smell F1
Merged (val)	No	No	0.6984	0.5145	0.5495	0.9238	0.6541
Merged (val)	No	Yes	0.7483	0.5723	0.6106	0.9383	0.6973
Balanced (val)	No	No	0.6204	0.6291	0.6508	0.8980	0.8339
Balanced (val)	No	Yes	0.6939	0.6965	0.7065	0.9135	0.8436
Balanced (test)	Yes	Yes	0.6286	0.6577	0.6276	0.9004	0.8392

As shown in Table 4, hint-only augmentation improved results on both dataset variants, with the strongest gains on the merged dataset. On the merged validation split, enabling hint-only augmentation improved all reported aggregate metrics. On the balanced validation split, hint-only augmentation again improved overall results, particularly for micro-F1, macro-F1, and smell/no-smell F1, although the gains were smaller than on the merged dataset. Overall, the balanced dataset yielded stronger smell-detection and aggregate F1 performance, whereas the merged dataset preserved stronger exact-match performance, indicating a trade-off between class balance and preservation of the natural data distribution.

Per-label performance also varied across dataset variants and augmentation settings. In the merged dataset, Ambiguous and Subjective tended to be weaker than Negative and Vague, which is linguistically plausible because these categories often depend more on context and interpretation than on stable lexical cues. In the balanced setting, overall multi-label and smell/no-smell performance improved, but Nonverifiable showed less stable test performance than several other categories. Across conditions, Negative and Vague remained among the most consistently detectable labels, suggesting stronger textual regularities. Prior research on requirement smells has similarly shown that smell types differ in detectability depending on how directly they are expressed in language [2].

Comparisons with earlier requirement smell detection studies should be interpreted cautiously because dataset preparation, label granularity, class balance, and evaluation settings differ. The present results are therefore best understood as comparative evidence within the current experimental setup rather than as a strict benchmark ranking against prior work.

A qualitative example further illustrates the model's behavior. For the requirement "If any part of the call fails, an audible and visual indication shall be provided in the appropriate cab.", the classifier assigned the highest probability to Subjective, which is linguistically reasonable because the phrase "appropriate cab" introduces an evaluative and context-dependent formulation rather than a precisely measurable criterion. Such cases suggest that the model responds not only to isolated keywords, but also to wording patterns associated with requirement quality concerns.

Overall, the comparative results suggest that the proposed approach is effective for multi-label requirement smell detection, while also showing that dataset composition and hint-only augmentation materially influence performance. Hint-only augmentation improved results on both dataset variants, whereas the balanced dataset produced stronger smell/no-smell and aggregate F1 scores and the merged dataset preserved stronger exact-match performance. The final reported test results were obtained from the best-performing balanced configuration with tuning and hint-only augmentation, evaluated on the held-out balanced test split. Together, these findings suggest that the classifier captures linguistically relevant distinctions useful for automated requirement quality analysis.

6 Threats to validity

This study has several threats to validity. Harmonization of multiple source datasets may introduce residual inconsistencies, while heuristic preprocessing decisions, including the 32-word limit, dictionary-based label alignment, and the exploratory balanced dataset variant, may influence the reported results. Hint-aware augmentation may also favor more explicit smell expressions over context-dependent ones, although it was restricted to label-preserving lexical transformations. In addition, the limited size and composition of the available datasets may reduce the stability and generalizability of label-specific results. Therefore, the findings should be interpreted within the current experimental setup rather than as a strict benchmark comparison with prior work.

7 Conclusion

This study has several threats to validity. Harmonization of multiple source datasets may introduce residual inconsistencies, while heuristic preprocessing decisions, including the 32-word limit, dictionary-based label alignment, and the exploratory balanced dataset variant, may influence results. Hint-aware augmentation may also favor more explicit smell expressions over context-dependent ones, although it was restricted to label-preserving lexical transformations. In addition, the limited size and composition of the available datasets may reduce the stability and generalizability of label-specific results. Therefore, the findings should be interpreted within the current experimental setup rather than as a strict benchmark comparison with prior work.

References

[1] N. Kiyavitskaya, N. Zeni, L. Mich, and D. M. Berry, "Requirements for tools for ambiguity identification and measurement in natural language requirements specifications," *Requir. Eng.*, vol. 13, no. 3, pp. 207–239, 2008, doi: 10.1007/s00766-008-0063-7.

[2] H. Femmer, D. Méndez Fernández, S. Wagner, and S. Eder, "Rapid quality assurance with Requirements Smells," *Journal of Systems and Software*, vol. 123, pp. 190–213, Jan. 2017, doi: 10.1016/j.jss.2016.02.047.

[3] A. Veizaga, S. Y. Shin, and L. C. Briand, "Automated Smell Detection and Recommendation in Natural Language Requirements," *IEEE Transactions on Software Engineering*, vol. 50, no. 4, pp. 695–720, Apr. 2024, doi: 10.1109/TSE.2024.3361033.

[4] M. K. Habib, S. Wagner, and D. Graziotin, "Detecting Requirements Smells With Deep Learning: Experiences, Challenges and Future Work," Aug. 2021, doi: 10.1109/REW53955.2021.00027.

[5] A. L. Alem, K. K. Gebretsadik, S. A. Mengistie, and M. F. Admas, "Multi-label software requirement smells classification using deep learning," *Sci. Rep.*, vol. 15, no. 1, Dec. 2025, doi: 10.1038/s41598-025-86673-w.

[6] L. Tunstall *et al.*, "Efficient Few-Shot Learning Without Prompts," Sep. 2022, [Online]. Available: http://arxiv.org/abs/2209.11055

[7] H. Femmer, D. M. Fernández, E. Juergens, M. Klose, I. Zimmer, and J. Zimmer, "Rapid requirements checks with requirements smells: Two case studies," in *1st International Workshop on Rapid Continuous Software Engineering, RCoSE 2014 - Proceedings*, Association for Computing Machinery, Jun. 2014, pp. 10–19. doi: 10.1145/2593812.2593817.

[8] T. Kato and K. Tsuda, "A Method of Ambiguity Detection in Requirement Specifications by Using a Knowledge Dictionary," in *Procedia Computer Science*, Elsevier B.V., 2022, pp. 1482–1489. doi: 10.1016/j.procs.2022.09.205.

[9] C. Cheng *et al.*, "A General Primer for Data Harmonization," Dec. 01, 2024, *Nature Research*. doi: 10.1038/s41597-024-02956-3.

[10] M. Bayer, M.-A. Kaufhold, and C. Reuter, "A Survey on Data Augmentation for Text Classification"
[11] J. S. Aguilar-Ruiz and M. Michalak, "Classification performance assessment for imbalanced multiclass data," *Sci. Rep.*, vol. 14, no. 1, Dec. 2024, doi: 10.1038/s41598-024-61365-z.
[12] M. Zakeri-Nasrabadi and S. Parsa, "Natural Language Requirements Testability Measurement Based on Requirement Smells," *Neural Comput. Appl.*, vol. 36, no. 21, pp. 13051–13085, Mar. 2024, doi: 10.1007/s00521-024-09730-x.