POLITECNICO DI TORINO
Repository ISTITUZIONALE

Classifying Gas Pipe Damage Descriptions in Low-Diversity Corpora

Original
Classifying Gas Pipe Damage Descriptions in Low-Diversity Corpora / Catalano, L., D'Asaro, F., Pantaleo, M., Jamshed, M., Acharjee, P., Giulietti, N., Fossat, E., Rizzo, G.. - 4112:(2025). (CLiC-it 2025 – Eleventh Italian Conference on Computational Linguistics Cagliari (ITA) September 24-26, 2025).

Availability:
This version is available at: 11583/3002060 since: 2025-07-24T08:44:49Z

Publisher:
CEUR

Published
DOI:

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

26 November 2026

Classifying Gas Pipe Damage Descriptions in Low-Diversity Corpora

Luca Catalano[1,*], Federico D'Asaro[1,2], Michele Pantaleo[2], Minal Jamshed[2], Prima Acharjee[2], Nicola Giulietti[3], Eugenio Fossat[3] and Giuseppe Rizzo[1]

[1]*LINKS Foundation – Torino, Italy*
[2]*Politecnico di Torino – Torino, Italy*
[3]*Composite Research – Torino, Italy*

Abstract

This paper introduces a retrieval-based text classification framework tailored for language corpora in the domain of gas pipe damage description analysis, with a specific focus on determining patch applicability. Due to the scarcity of free-text damage descriptions in this domain, we construct a synthetic binary classification dataset, referred to as *CoRe-S*. This dataset consists of 11,904 damage descriptions generated from structured attributes, where each instance is labeled as either *Patchable* (True) or *Unpatchable* (False). The *CoRe-S* dataset presents two primary challenges: (i) a class imbalance, where positive cases are the minority, and (ii) frequent use of domain-specific terminology, which results in low lexical diversity across descriptions. To quantify this lack of variation, we introduce the *Corpus Pairwise Diversity* statistic, which measures the degree of lexical dissimilarity between documents in a corpus.

We adopt a training-free, retrieval-based text classification approach and demonstrate that *Sentence-BERT-NLI* is the most effective encoder under low-diversity conditions, as it excels at capturing subtle lexical and semantic differences between otherwise similar documents. To address the class imbalance, we apply random undersampling, which outperforms other under-sampling strategies in our experiments. Our results show that the proposed retrieval-based classifier significantly outperforms other training-free text classification methods—whether zero-shot, few-shot, or similarity-based—achieving an improvement of approximately 35.2% in macro F1-score over the second-best method.

Our code is publicly available at: https://github.com/links-ads/core-unimodal-retrieval-for-classification.

Keywords

Gas pipe damage description analysis, Training-free text classification, Low lexical diversity, Low lexical diversity

1. Introduction

Text classification is the task of assigning predefined labels to a given text and has been applied to a wide range of domains, including sentiment analysis [1], emotion recognition [2], news classification [3], and spam detection [4]. Early approaches typically decomposed the task into two stages: feature extraction using neural models such as Recurrent Neural Networks (RNNs) [5, 6] or Convolutional Neural Networks (CNNs) [7], followed by feeding the extracted features into a classifier [8] to predict labels. With the emergence of transformer architectures [9], Large Pretrained Models (LPMs) such as BERT [10] and GPT [11] have become the foundation for modern NLP systems. Trained on massive textual corpora, these models demonstrate strong generalization capabilities across various downstream tasks, often without requiring additional task-specific training data. In this work, we address the task of text classification over gas pipe damage descriptions, with the objective of determining whether a patch is applicable (*True*) or not (*False*). Due to the limited availability of free-text damage reports in this domain, we construct a synthetic

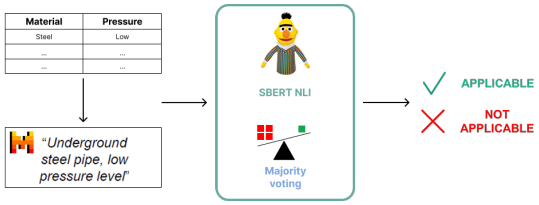


Figure 1: Overview of our classification framework. Starting with a tabular dataset, we use the Mistral-7B language model to generate textual descriptions. For a given input query, we retrieve the most similar labeled descriptions using embedding similarity. The final label is determined through a majority voting mechanism.

CLiC-it 2025: Eleventh Italian Conference on Computational Linguistics, September 24 — 26, 2025, Cagliari, Italy
*Corresponding author.
†These authors contributed equally.
luca.catalano@linksfoundation.com (L. Catalano); federico.dasaro@polito.it (F. D'Asaro); michele.pantaleo@studenti.polito.it (M. Pantaleo); s329091@studenti.polito.it (M. Jamshed); prima.acharjee@studenti.polito.it (P. Acharjee); n.giulietti@composite-research.com (N. Giulietti); e.fossat@composite-research.com (E. Fossat); giuseppe.rizzo@linksfoundation.com (G. Rizzo)
0009-0007-9306-6883 (L. Catalano); 0009-0003-8727-3393 (F. D'Asaro); 0009-0000-0485-1310 (M. Pantaleo); 0009-0007-3082-8633 (M. Jamshed); 0009-0007-8678-8765 (P. Acharjee); 0000-0003-0083-813X (G. Rizzo)
© 2025 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR-WS.org/Vol-4112/21_main_long.pdf

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073
published 2025-11-30

binary classification dataset, referred to as **CoRe-S**. This dataset comprises 11,904 damage descriptions generated from structured attributes such as pipe material, lesion type, and pipe exposure. This setting poses two main challenges: (i) a class imbalance, where positive cases are the minority; and (ii) low lexical diversity, as descriptions tend to be highly similar across classes, relying heavily on domain-specific terminology and recurring linguistic patterns. Consequently, texts from different categories may be lexically indistinguishable, complicating classification based on surface-level features.

To quantify this lexical variability, we introduce a novel statistic, **Corpus Pairwise Diversity**, which measures the degree of lexical dissimilarity between documents within a corpus. When applied to our dataset, this statistic produces significantly lower values compared to generalist corpora such as *20NewsGroups* [12], which are characterized by a broader vocabulary and greater topical diversity.

For the classification task, we employ a training-free, retrieval-based framework, depicted in Figure 1, that leverages PLMs, consisting of a document encoder and a similarity-based classifier. Given the low corpus diversity and frequent repetition of domain-specific terms—regardless of class—conventional semantic search models may underperform in this setting, as they often fail to capture fine-grained linguistic distinctions. For instance, two descriptions may differ only in a subtle feature such as pressure level, which can determine whether a leak is patchable.

This observation motivates the hypothesis that encoders focusing on logical inference, rather than relying solely on surface-level semantic similarity, are better suited for classification in such contexts. Accordingly, we employ the Sentence-BERT model pre-trained on Natural Language Inference (NLI), a task that requires determining whether a hypothesis can be logically inferred, contradicted, or is neutral with respect to a given premise. We adopt *SBERT-NLI* [13], which effectively captures subtle lexical and semantic differences between near-identical documents. To mitigate the effects of class imbalance, we apply random undersampling to the retrieval corpus, which achieves superior performance compared to alternative imbalance-handling strategies in our experiments. Experimental results demonstrate that our text classification model consistently outperforms state-of-the-art training-free approaches, including zero-shot, few-shot, and similarity-based methods.

The main contributions of this work are as follows:

- We introduce **CoRe-S**, a novel dataset in the domain of gas pipe damage descriptions, which, to the best of our knowledge, is the first dataset developed in this domain.
- We introduce a novel statistic, **Corpus Pairwise Diversity**, to quantify the lexical dissimilarity between documents within a corpus.
- We demonstrate that in low-diversity settings, a Natural Language Inference–pretrained encoder, specifically *SBERT-NLI*, outperforms standard semantic similarity models by effectively capturing subtle distinctions between documents belonging to different classes.

2. Background on Training-Free Text Classification

With the advent of transformer architectures equipped with attention mechanisms [9], a new wave of Large-scale Pretrained Models (LPMs) has emerged. These models are trained on vast textual corpora such as BooksCorpus (800M words) [14] and Common Crawl [15]. Modern PLMs are predominantly based on either the BERT [10] or GPT [11] architectures. BERT utilizes a transformer encoder to produce dense contextual representations of input text, making it well-suited for language understanding tasks. In contrast, GPT adopts a decoder-only architecture originally designed for generative applications, though it has also shown strong performance in classification tasks [16, 17]. Both architectural families exhibit strong transfer learning capabilities, enabling effective adaptation to a variety of downstream tasks, and paving the way for training-free approaches to text classification.

BERT-based approaches leverage embeddings to compare semantic similarity between pieces of text. Depending on the nature of the task, these methods can be broadly categorized into: (i) *zero-shot methods*, which compare the input text directly with class labels or their representative keywords [18, 19, 13]; and (ii) *retrieval-based methods*, which perform semantic search over a database containing auxiliary knowledge [20, 21].

Schopf et al. [22] presented, for the first group of methods (zero-shot), two different approaches. The first one consists of representing each document as the average of its paragraph embeddings. Similarly, each label is represented as the average embedding of a set of predefined keywords associated with that label. Classification is then performed by computing the similarity between the document and label embeddings, assigning the label with the highest similarity score. The second approach, instead, implements a zero-shot entailment technique. Each input document is paired with a hypothesis representing a candidate label, and the model predicts whether the hypothesis is entailed by the input.

GPT-based approaches, on the other hand, leverage the full potential of natural language processing and the generative capabilities embedded in the models. These methods are typically applied in either: (i) a *zero-shot* fashion, where predictions are made without any labeled

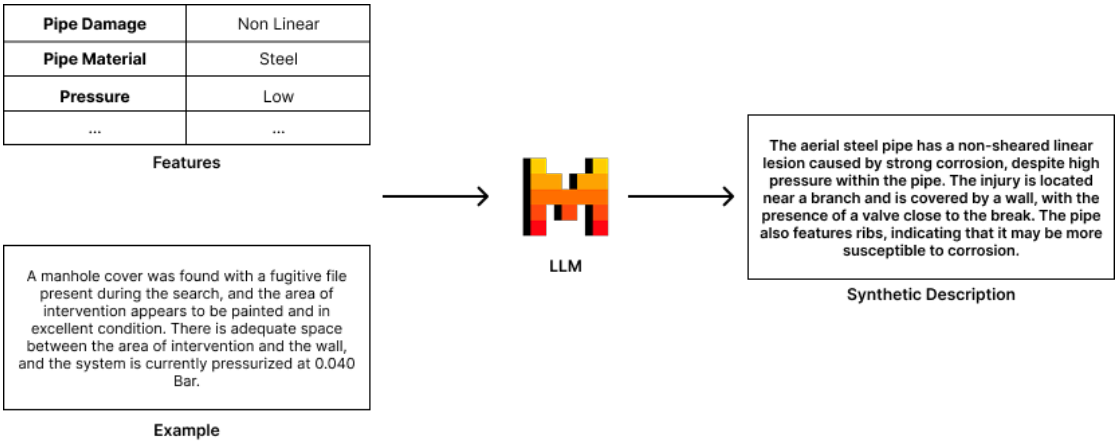


Figure 2: Illustration of the text generation process using the Mistral-7B model. Tabular data are transformed into natural language descriptions, based on the characteristics represented by the features and the style provided by the example description.

Table 1
Description of pipe-related variables and their possible values.

Variable Name	Description	Possible Values
`pipe_damage`	Type of damage affecting the pipe.	Galvanised fittings, Steel, Bitumen-coated steel, Polyethylene-coated steel, Cast iron, Polyethylene
`pipe_material`	Components or structural elements of the pipe.	Non-sheared linear lesion, Hole, Cluster of holes, Sheared linear lesion, Visible axial deformation, Thread, Elbow, Sleeve, Tee, Nipples, Ball valve
`corrosion`	Indicates whether the pipe is affected by corrosion.	True, False
`mecc_charac`	Indicates whether the pipe retains mechanical integrity despite corrosion.	True, False
`pressure`	Pressure level inside the pipe.	High, Low
`wall`	Whether there is a gap larger than 1 cm between the pipe and the wall.	True, False
`wall_more`	If a wall is present, indicates whether it can be broken or removed.	True, False
`valve`	Presence of a valve near the damaged area.	True, False
`ribs`	Presence of structural ribs on the pipe.	True, False
`coated_tube`	Indicates whether the pipe is coated.	True, False
`welds`	Presence of welds on the pipe.	True, False

examples [16]; or (ii) *in-context learning*, where the model generates textual outputs (e.g., label words) conditioned on a prompt that usually includes a few annotated examples for downstream tasks [23, 24].

In this paper, we present a BERT-based, retrieval-enhanced approach to tackle two central challenges in the classification of gas pipe damage descriptions: low lexical diversity and class imbalance.

3. CoRe-S Dataset

In current pipe repair operations, data is typically collected through structured questionnaires completed after the intervention. These forms use categorical and boolean fields to document the conditions surrounding the fault and the type of repair performed.

In this work, we propose a simplified data collection approach based on free-text fault descriptions. We also introduce a novel use case for these descriptions: supporting technicians in determining whether a fault is patchable or requires replacement of the damaged pipe segment.

To explore this idea and assess its feasibility, we construct a synthetic dataset by transforming existing structured tabular data—originally collected in the field—into natural language descriptions.

The original tabular dataset comprises 11,904 pipe repair interventions. Each intervention is described using 11 categorical or boolean features—listed in Table 1—which capture the condition of the pipe at the time of the damage. Additionally, each record is labeled as Patchable (`True`) or Not Patchable (`False`), depending on whether the intervention involved a successful patch or required replacement of the pipe segment. Among all interventions, only 126 examples (1.06%) are labeled as successful patches, while the remaining 11,778 (98.94%) represent replacements.

We generate the textual descriptions using the large language model (LLM) *Mistral-7B Instruct v0.3*[1].

Figure 2 illustrates through an example the pipeline used to generate the dataset, where a prompt—shown in Figure 3—combines (i) a randomly selected example from a curated set of 36 real technician-written descriptions and (ii) a structured template filled with the most informative features extracted from the tabular dataset, enabling the LLM to produce realistic and domain-specific textual representations of pipe failures.

Specifically, for each entry in the original tabular dataset $\mathbf{x}_i \in \mathbb{R}^F$, we extract the relevant feature values and insert them into the template prompt, together with the example used to guide the writing style.

The label $y_i \in True, False$ indicates whether the intervention was resolved via patching ($y_i = True$) or required pipe replacement ($y_i = False$), and is directly inherited from the original dataset.

The resulting **CoRe-S** dataset consists of pairs (t_i, y_i), where t_i is the synthetic textual description generated from the structured features of intervention i, and y_i is the corresponding repair label.

To ensure the quality and reliability of the generated descriptions, we perform a human review process to: (i) verify stylistic consistency with real examples written by technicians, and (ii) randomly assess the semantic alignment between each description t_i and the original feature vector $\mathbf{x}_i$.

Prompt

You are required to write a descriptive paragraph about a pipe fault using the following details:

- Pipe damage: {pipe_damage}

- Pipe Material: {pipe_material}

- Strong Corrosion: {corrosion}, but maintains its mechanical characteristics: {mec_charact}

- The pipe is spaced from the wall by at least 1 cm: {wall} (in case of False is possible to create space between pipe and wall: {wall_more})

- Pressure less than 0.040 in the Pipe: {pressure}

- Presence of escape joint connection or valve near the Break: {valve}

- Presence of ribs: {ribs}

- Coated tube: {coated_tube}

- presence of welds: {welds}

Here is an example of textual description you can use as reference {example}

Please avoid providing explanations regarding the causes or consequences of the fault.

Figure 3: Prompt template used for converting tabular data representing pipe damage into textual descriptions. The prompt is composed of: (1) the features relevant for generating the content, and (2) an example description written by a specialist to guide the style.

4. Corpus Pairwise Diversity Statistic

This section introduces the formal definition of the **Corpus Pairwise Diversity** statistic, which serves as a foundational element for both the design and evaluation of our retrieval-based classifier. By measuring the average dissimilarity between the vocabularies of document pairs, this statistic informs downstream components that rely on accurate estimations of inter-document similarity.

4.1. Definition

Let $D = \{d_1, \ldots, d_N\}$ be a corpus of N documents, where each document d_i is represented as the set of its unique terms. The *Jaccard distance* between two documents d_i and d_j is

$$\delta_J(d_i, d_j) = 1 - \frac{|d_i \cap d_j|}{|d_i \cup d_j|} \in [0, 1].$$

We then define the *Corpus Pairwise Diversity* statistic as

$$CPD(D) = \frac{1}{\binom{N}{2}} \sum_{1 \le i < j \le N} \delta_J(d_i, d_j).$$

By construction, $CPD(D) \in [0, 1]$; low values indicate high overall similarity, and high values indicate high overall dissimilarity among documents. It is *non-negative*, symmetric, and unaffected by the order of the set D.

[1] https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3

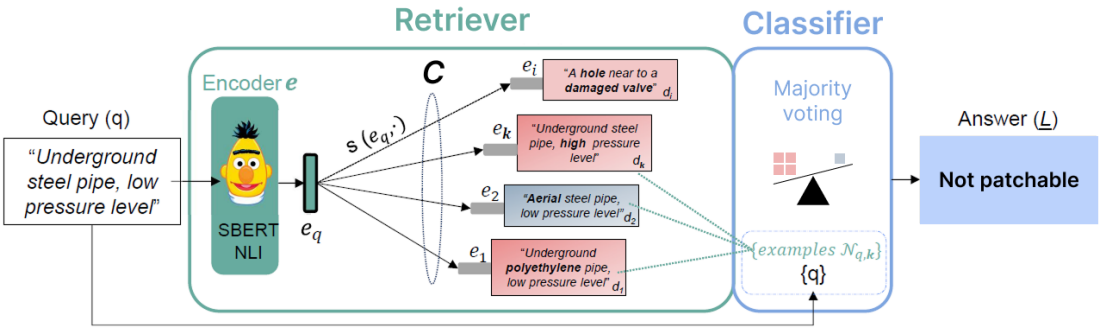


Figure 4: Overview of our classification pipeline. Given an input query Q, we retrieve the top-k most similar labeled descriptions based on embedding similarity. The final label is assigned using a majority voting mechanism over the retrieved top-k documents.

Table 2
Pairwise lexical diversity (CPD) measured across various text classification datasets. Here, $|D|$ indicates the number of documents, and $|V|$ represents the vocabulary size.

Dataset	$\|D\|$	$\|V\|$	$CPD(D)$
20NewsGroups	10,998	85,551	0.99
Yahoo! Answers	1,375,428	739,655	0.99
CoRe-S	11,903	2283	0.69

Moreover, it is also *invariant to document length and term frequency*, even when vocabulary sizes differ substantially.

4.2. Empirical Analysis

To better understand the behaviour of the CPD statistic, we compute it across multiple corpora. Table 2 shows that datasets like *20NewsGroups* and *Yahoo! Answers* generally obtain higher diversity scores $CPD(D)$, indicating increased textual heterogeneity and more extensive vocabularies. In contrast, the *CoRe-S* dataset exhibits lower diversity, which can be attributed to its specialized terminology and repetitive textual patterns. This is likely a consequence of the constrained set of attributes used during the generation process (see Section 3), which restricts variability in term usage. As a result, it becomes challenging to distinguish between damage descriptions across different categories.

5. Retrieval-based Classifier

We adopt a zero-shot learning approach, depicted in Figure 4 built around a retrieval-based pipeline. The strategy involves retrieving the top-k most similar labeled textual descriptions based on embedding similarity and, using a majority voting mechanism across these top-k retrieved instances, determine the final label assigned to the input.

5.1. Formal Description

Let $D \subseteq X^*$ be the set of all documents, where X is a finite alphabet of symbols. The dataset D is partitioned into two subsets: the *query set* $Q \subseteq D$ and the *corpus* $C \subseteq D$. For each query $q \in Q$, the system retrieves relevant documents from the *corpus* C, which contains descriptions of past pipe failures, each labeled as *patchable* (true) or *not patchable* (false). Let $e : X^* \to \mathbb{R}^n$ be an *encoding function* that maps a document into an n-dimensional embedding space using a pre-trained model and let $s : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ be a *similarity function* that measures the closeness between two embedded documents $q \in Q$ and $c \in C$, the *retrieval process* is defined as:

$$\mathrm{Re}_{s,k,C}(q) = \underset{N \subseteq C : |N| = k}{\arg\max} \sum_{c \in N} s(e(q), e(c)) \quad (1)$$

where C is the corpus, k is the number of top retrieved documents, $s(e(q), e(d))$ is the similarity score between the query document$q$ and the corpus document c. We denote the resulting top-k retrieved documents for a given query q as:

$$N^*_{q,k} = \mathrm{Re}_{s,k,M}(q) \quad (2)$$

Finally, the system produces its final prediction by applying *majority voting* over the labels of the documents in $N^*_{q,k}$:

$$\hat{y}_q = \mathrm{MajorityVote}\left(\{\mathrm{label}(d) \mid d \in N^*_{q,k}\}\right) \quad (3)$$

5.2. Encoder and Similarity Metrics Selection

For our training-free classification pipeline, we explore several pre-trained encoders to generate high-

quality semantic embeddings for both queries and corpus documents. All selected encoders are transformer-based models chosen for their zero-shot capabilities, strong performance on general-purpose semantic similarity benchmarks, and availability through the `sentence-transformers` library, which facilitates seamless integration into our pipeline. Specifically, we test `all-mpnet-base-v2`[2], a sentence-transformer model based on MPNet [19], fine-tuned on over 1 billion sentence pairs for semantic similarity tasks. We also include `multi-qa-mpnet-base`[3], a variant of MPNet fine-tuned on multiple question-answering datasets—including Natural Questions, TriviaQA, and SQuAD—to better handle question-style inputs [25]. Finally, we use `bert-base-nli-mean-tokens`[4], a BERT-based encoder trained on the SNLI and MultiNLI datasets for natural language inference (NLI) [13].

We evaluate two popular similarity metrics for comparing document embeddings: the *dot product*, which captures the directional similarity between embeddings and the *Euclidean distance* (ℓ_2), which measures the straight-line distance between vectors in the embedding space.

5.3. Corpus Under-sampling Techniques

To address class imbalance in our dataset, we use several under-sampling strategies that reduce the number of documents in the corpus set of the majority class. We test different algorithms: Random Under-sampling, Near Miss with its 3 different versions and the Edited Nearest Neighborhood.

5.3.1. Random Under-sampling

It is a simple technique that randomly removes examples from the majority classes until the desired class distribution is reached.

5.3.2. NearMiss

The algorithm consists of preserving samples from the majority class that are most relevant for the classification task, based on the evaluations of distances between samples from the majority and minority classes. There are different versions of the same algorithm:

- **NearMiss-1** selects majority class samples with the smallest average distance to the closest samples of the minority class.
- **NearMiss-2** selects majority class samples with the smallest average distance to the farthest samples of the minority class.
- **NearMiss-3** first selects a subset of minority samples and retains their nearest neighbors among the majority. Then, it keeps the majority class samples with the largest average distance to their selected neighbors.

5.3.3. Edited Nearest Neighbors (ENN)

The `EditedNearestNeighbors` (ENN) technique uses a K-Nearest Neighbors (KNN) approach to filter out noisy or ambiguous samples from the majority class. The procedure involves training a KNN classifier on the entire corpus, then for each instance in the majority class, identifying its k nearest neighbors and remove the instance if any or most of its neighbors belong to a different class.

6. Experiments

6.1. Experimental Details

Experiments are conducted using an NVIDIA GeForce RTX 2080 Ti GPU. Model performance is primarily evaluated using the F1-Macro score to ensure a balanced assessment across classes. Additionally, all results are obtained through 5-fold cross-validation, which involves changing the split of the corpus and query set in each fold to ensure robust evaluation. For the main results, we also report the Recall-Macro and Precision-Macro scores.

6.2. Results

6.2.1. Comparison with Zero-Shot Classification Methods

We compare our zero-training retrieval-based classification approach with several zero-shot and few-shot classification baselines.

The **Baseline approach**[22] represents each document as the average of its paragraph embeddings. Similarly, each label is represented as the average embedding of a set of predefined keywords associated with that label. Classification is performed by computing the similarity between the document and label embeddings, assigning the label with the highest similarity score. We evaluate this method using two different encoders: `all-MiniLM-L6-v2` and `all-mpnet-base-v2`.

We also implement a **zero-shot entailment technique**[22], using pre-trained models such as `DistilBERT`, `BART-large`, and `DeBERTa`. Each input document is paired with a hypothesis representing a candidate label, and the model predicts whether the hypothesis is entailed by the input.

[2] https://huggingface.co/sentence-transformers/all-mpnet-base-v2
[3] https://huggingface.co/sentence-transformers/multi-qa-mpnet-base-dot-v1
[4] https://huggingface.co/sentence-transformers/bert-base-nli-mean-tokens

Table 3
Main results across CoRe Synthetic dataset.

Method	Precision *(macro)*	Recall *(macro)*	F1 *(macro)*
Baseline-mean	0.517	0.506	0.508
0SHOT-NLI	0.497	0.500	0.497
LLM – 0SHOT	0.514	0.736	0.466
LLM – FEWSHOT	0.517	**0.756**	0.501
OURS	0.704	0.672	**0.687**

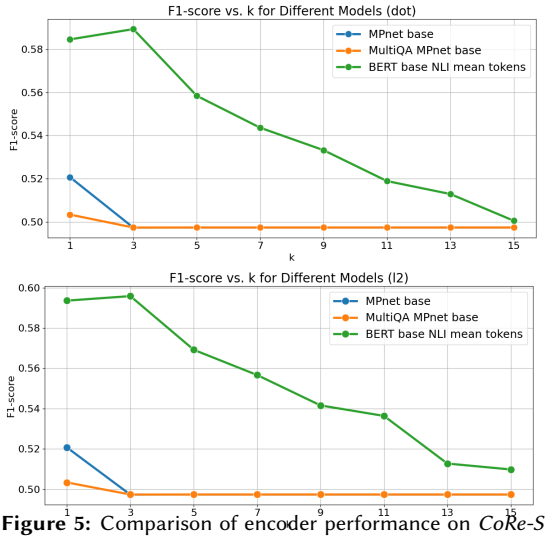


Figure 5: Comparison of encoder performance on *CoRe-S* datasets using different similarity metrics for various values of k.

Additionally, we evaluate **large language models (LLMs)** through zero-shot and few-shot prompting. In the zero-shot setting, the prompt includes only a description of the task and candidate labels. For the few-shot setting, we extend the prompt by adding two randomly selected labeled examples (one per class) drawn from the training set. These examples are excluded from the test set.

Table 3 presents the best results obtained by each method. Our retrieval-based approach achieves the highest performance overall, reaching a macro-F1 of 0.687.

6.2.2. Ablation Study

Similarity Metric Selection In our zero-shot pipeline, we evaluate two most used similarity metrics: the **dot product** and the **Euclidean distance** (ℓ_2). Figure 5 illustrates the performance of our retrieval-classification strategy under both similarity functions. Both metrics are tested across all selected encoders to determine which yields better retrieval quality for classification. Our experiments reveal that the ℓ_2 distance consistently outperforms the dot product: it better captures fine-grained semantic dissimilarity by measuring absolute geometric distance in embedding space. Consequently, all the other results in this work are reported using ℓ_2 similarity.

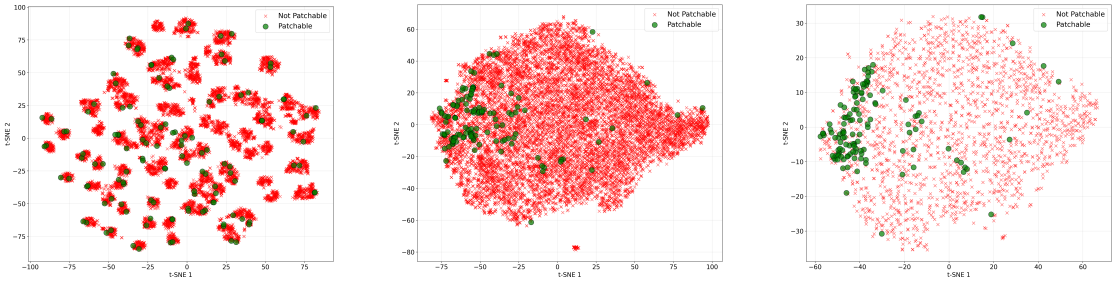
Encoder Selection Figure 5 shows macro-F1 performance for values of k ranging from 1 to 15, using both dot-product and Euclidean (ℓ_2) similarity on the *CoRe-S* dataset. Across all values of k, SBERT consistently outperforms MPNet and QA-MPNet, achieving peak performance at $k = 3$ with Euclidean similarity. This indicates that NLI-pretraining enables SBERT to better capture logical relations relevant to the task.

This performance advantage becomes more evident when considering the linguistic challenges posed by the *CoRe-S* dataset: it contains a high degree of lexical overlap between sentence pairs, where distinctions often hinge on subtle cues such as *adjectives and negation* (e.g., "low vs. high pressure" or "no corrosion"). These subtle differences are critical in determining whether a leak is patchable or non-patchable. General-purpose encoders can be misled by shared technical terms like *pressure* or *pipe*, which can dominate similarity computations without truly capturing the underlying logical relationship.

To further illustrate this, Figure 6(a) and Figure 6(b) show t-SNE visualizations of the document embeddings produced by MPNet and SBERT-NLI, respectively. In Figure 6(b), embeddings corresponding to the same label—especially the green points representing the positive class—tend to be more tightly clustered. In contrast, Figure 6(a) reveals that MPNet's embeddings are more dispersed, with red points (False label) forming scattered clusters across the space. The SBERT-NLI plot also exhibits a prominent macro-cluster containing both true and false labels, but with a clearer organization and denser neighborhood structures, particularly among positively labeled instances. This spatial coherence further supports the claim that entailment-based encoders are better equipped to model subtle semantic nuances crucial for this task.

Corpus Under-sampling Figure 7 shows how different sampling strategies impact performance on the *CoRe-S* dataset across values of k from 1 to 15. The results reported in the figure represent the best outcomes obtained across the tested hyperparameter configurations.

When no under-sampling is applied, macro-F1 peaks at $k = 7$ (0.609), but then declines as if additional neighbors introduce semantic noise. In contrast, applying under-sampling leads to higher macro-F1 scores across all values of k. Notably, random under-sampling achieves the best overall performance, improving from 0.601 at $k = 1$ to



(a) MPNet (b) SBERT NLI (c) SBERT NLI + Undersampling

Figure 6: t-SNE visualizations of embeddings produced by: (a) the MPNet encoder, (b) SBERT NLI, and (c) SBERT NLI with random undersampling. Clearer cluster separation is observed between the labels `true` (patchable) and `false` (not patchable).

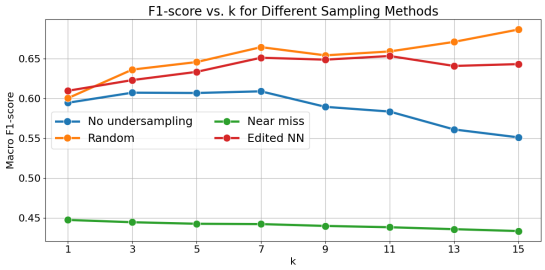


Figure 7: Comparison of our retrieval-for-classification approach with different undersampling strategies on *CoRe-S* datasets using ℓ_2 similarity for various values of k.

a peak of 0.687 at $k = 15$. This suggests that random under-sampling effectively balances the class distribution in the corpus, enabling the model to achieve stronger generalization and more robust performance.

The use of near-miss under-sampling, on the other hand, significantly degrades performance. Although the edited nearest neighbor (edited nn) strategy performs better than using no under-sampling at all, it still falls short of the results achieved with random under-sampling. This may be because these strategies remove fewer training examples and may not sufficiently rebalance the corpus. In fact, the high similarity in textual descriptions with label patchable or non-patchable can lead to very close embeddings and as a result, these strategies might remove fewer examples. Random under-sampling instead operates solely based on a class ratio threshold, resulting in a more pronounced reduction and a more effective rebalancing of the corpus. The best performance is achieved with a reduced corpus of 962 training samples and the full set of 5,952 query instances.

Figure 6(c) illustrates a t-SNE representation of the document embeddings produced by the best encoder, SBERT-NLI, after applying random under-sampling to the corpus. As previously shown, SBERT-NLI naturally clusters green points (true labels) and red points (false labels) near each other, reflecting its ability to capture fine-grained semantic distinctions. After under-sampling, however, the red points are pushed further away from the green points, creating clearer separations between classes. This enhanced separation corresponds to improved macro-F1 performance, demonstrating how under-sampling helps the model better distinguish between patchable and non-patchable instances by reducing class imbalance and mitigating semantic noise.

6.2.3. Cross-Corpus Encoder Selection with Varying Lexical Diversity

To further explore the influence of corpus lexical diversity on model performance, we expand our evaluation beyond *CoRe-S* to include two additional text classification datasets: *20NewsGroups* and *Yahoo Answers*, both of which demonstrate higher lexical variability, as shown in Section 4 using our proposed *Corpus Pairwise Diversity* statistic.

We compare the performance of three document encoders within the same retrieval-based classification framework: SBERT-NLI, MPNet and QA-MPNet. For evaluation, each dataset's test set is evenly split into two subsets: one half is used as the retrieval corpus and the other half as the query set, where classification performance is measured.

Table 4 reports the best F1 scores achieved by each encoder on the respective datasets. The results reveal a clear interaction between corpus lexical diversity and encoder effectiveness. On the low-diversity *CoRe-S* dataset, SBERT-NLI achieves the highest F1 score, supporting our hypothesis that NLI-pretrained models are better suited for distinguishing fine-grained linguistic nuances between similar documents. In contrast, on the higher-diversity datasets *20NewsGroups* and *Yahoo Answers*, MPNet consistently outperforms the other encoders. In these settings, MPNet's enhanced ability to capture broad semantic content makes it more effective at handling lexical variation.

Table 4
Best F1 scores obtained with each encoder across datasets, using the same retrieval-based classification framework.

Dataset	Model	F1 Score
20NewsGroup	MPNet	**0.752**
	SBERT-NLI	0.555
	QA-MPNet	0.744
Yahoo Answers	MPNet	**0.638**
	SBERT-NLI	0.505
	QA-MPNet	0.618
CoRe-S	MPNet	0.521
	SBERT-NLI	**0.609**
	QA-MPNet	0.503

7. Limitations

A key limitation of this study is the reliance on synthetic data. While synthetic fault descriptions are necessary due to the lack of large-scale real-world technician-written reports, they may not fully capture the noise, variation, and contextual complexity present in actual field documentation. This may affect the generalizability of the findings when applied to real-world scenarios. Future work should explore the collection and use of authentic, technician-authored data to validate and refine the proposed method.

8. Conclusion

In this paper, we address the task of classifying gas pipe damage descriptions. Starting from a set of damage features and real examples, we generate a new dataset called *CoRe-S*, the first of its kind in this domain. This dataset exhibits low lexical diversity, characterized by a restricted and repetitive vocabulary, along with severe class imbalance. To quantify lexical diversity within a corpus, we propose the *Corpus Pairwise Diversity* statistic.

To overcome these challenges, we design a training-free retrieval-based text classifier that leverages SBERT-NLI to handle low lexical diversity, combined with undersampling techniques to mitigate class imbalance. Experimental results demonstrate that our method outperforms other training-free approaches, including zero-shot, few-shot, and similarity-based methods. Additional experiments suggest that natural language inference pretrained text encoders are particularly effective in low-diversity scenarios where subtle differences between texts of different labels must be captured.

Future work may involve a more extensive comparison of text encoder effectiveness across various text classification datasets exhibiting different levels of lexical diversity.

Acknowledgments

The authors acknowledge that this work has been partially funded by the European Union and by the Italian Ministry of Enterprises and Made in Italy (MIMIT), through the EXPAND project, Grant Agreement No. 101083443.

References

[1] B. Liu, Sentiment analysis and opinion mining, Springer Nature, 2022.
[2] F. D'Asaro, J. J. M. Villacís, G. Rizzo, Transfer learning of large speech models for italian speech emotion recognition, in: 2024 IEEE 18th International Conference on Application of Information and Communication Technologies (AICT), IEEE, 2024, pp. 1–6.
[3] N. Rai, D. Kumar, N. Kaushik, C. Raj, A. Ali, Fake news classification using transformer based enhanced lstm and bert, International Journal of Cognitive Computing in Engineering 3 (2022) 98–105.
[4] T. Liu, S. Li, Y. Dong, Y. Mo, S. He, Spam detection and classification based on distilbert deep learning algorithm, Applied Science and Engineering Journal for Advanced Research 3 (2024) 6–10.
[5] G. Wang, C. Li, W. Wang, Y. Zhang, D. Shen, X. Zhang, R. Henao, L. Carin, Joint embedding of words and labels for text classification, arXiv preprint arXiv:1805.04174 (2018).
[6] Q. Xie, Z. Dai, E. Hovy, T. Luong, Q. Le, Unsupervised data augmentation for consistency training, Advances in neural information processing systems 33 (2020) 6256–6268.
[7] J. Wei, K. Zou, Eda: Easy data augmentation techniques for boosting performance on text classification tasks, arXiv preprint arXiv:1901.11196 (2019).
[8] A. Jacovi, O. S. Shalom, Y. Goldberg, Understanding convolutional neural networks for text classification, arXiv preprint arXiv:1809.08037 (2018).
[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in neural information processing systems 30 (2017).
[10] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 conference of the North American chapter of

the association for computational linguistics: human language technologies, volume 1 (long and short papers), 2019, pp. 4171–4186.
[11] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, et al., Improving language understanding by generative pre-training.(2018), 2018.
[12] K. Lang, Newsweeder: Learning to filter netnews, in: Machine learning proceedings 1995, Elsevier, 1995, pp. 331–339.
[13] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, arXiv preprint arXiv:1908.10084 (2019).
[14] Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, S. Fidler, Aligning books and movies: Towards story-like visual explanations by watching movies and reading books, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 19–27.
[15] J. Ye, X. Chen, N. Xu, C. Zu, Z. Shao, S. Liu, Y. Cui, Z. Zhou, C. Gong, Y. Shen, et al., A comprehensive capability analysis of gpt-3 and gpt-3.5 series models, arXiv preprint arXiv:2303.10420 (2023).
[16] X. Sun, X. Li, J. Li, F. Wu, S. Guo, T. Zhang, G. Wang, Text classification via large language models, arXiv preprint arXiv:2305.08377 (2023).
[17] Z. Wang, Y. Pang, Y. Lin, Large language models are zero-shot text classifiers, arXiv preprint arXiv:2312.01044 (2023).
[18] T. Schopf, D. Braun, F. Matthes, Lbl2vec: An embedding-based approach for unsupervised document retrieval on predefined topics, arXiv preprint arXiv:2210.06023 (2022).
[19] K. Song, X. Tan, T. Qin, J. Lu, T.-Y. Liu, Mpnet: Masked and permuted pre-training for language understanding, Advances in neural information processing systems 33 (2020) 16857–16867.
[20] S. Ahmadi, A. Shah, E. Fox, Retrieval-based text selection for addressing class-imbalanced data in classification, arXiv preprint arXiv:2307.14899 (2023).
[21] T. Abdullahi, R. Singh, C. Eickhoff, Retrieval augmented zero-shot text classification, in: Proceedings of the 2024 ACM SIGIR international conference on theory of information retrieval, 2024, pp. 195–203.
[22] T. Schopf, D. Braun, F. Matthes, Evaluating unsupervised text classification: zero-shot and similarity-based approaches, in: Proceedings of the 2022 6th International Conference on Natural Language Processing and Information Retrieval, 2022, pp. 6–15.
[23] O. Rubin, J. Herzig, J. Berant, Learning to retrieve prompts for in-context learning, arXiv preprint arXiv:2112.08633 (2021).
[24] H. Su, J. Kasai, C. H. Wu, W. Shi, T. Wang, J. Xin, R. Zhang, M. Ostendorf, L. Zettlemoyer, N. A. Smith, et al., Selective annotation makes language models better few-shot learners, arXiv preprint arXiv:2209.01975 (2022).
[25] N. Thakur, N. Reimers, J. Daxenberger, I. Gurevych, A. Anand, Beir: A heterogeneous benchmark for zero-shot evaluation of information retrieval models, Proceedings of the 30th ACM International Conference on Information and Knowledge Management (CIKM) (2021). URL: https://arxiv.org/abs/2104.08663.

Declaration on Generative AI

During the preparation of this work, the author(s) used ChatGPT (OpenAI) in order to: Paraphrase and reword and Improve writing style. After using these tool(s)/service(s), the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.