Team GPLSI at QClef 2025: Quantum-Inspired Instance Selection and Clustering

Notebook for the QuantumCLEF Lab at CLEF 2025

Juan Pablo Consuegra-Ayala[1,2,*,†], Aitana Morote-Martínez[1,2,†], Francisco Valero-Abellón[1,2,†], Elena Lloret[1,3,4,†], Paloma Moreda[1,3,4,†] and Manuel Palomar[1,3,4,†]

[1]*Natural Language Processing and Information Systems Group (GPLSI), University of Alicante, Alicante, Spain, 03690.*
[2]*Digital Intelligence Centre (CENID), University of Alicante, Alicante, Spain, 03690.*
[3]*University Institute for Computing Research (IUII), University of Alicante, Alicante, Spain, 03690.*
[4]*Department of Language and Computing Systems, University of Alicante, Alicante, Spain, 03690.*

Abstract

This paper presents the participation of the GPLSI team in the *Quantum CLEF (QClef) Lab* at *CLEF 2025*, focusing on *Task 2 - Instance Selection* and *Task 3 - Clustering*. The QClef Lab explores the applicability of quantum and quantum-inspired techniques to core AI tasks, emphasizing optimization efficiency and data reduction. In Task 2, we propose three multi-paradigm approaches for selecting representative training instances for sentiment classification, leveraging sentiment-aware pairing, local set-based criteria, and classical heuristics. In Task 3, we introduce a single quantum-inspired clustering framework that integrates four distinct pivot selection strategies for document grouping in embedding space. Our methods achieved competitive performance across both tasks. In particular, our *LocalSets* method achieved the highest effectiveness in Task 2 while substantially reducing the training set, and our *FPS-Medoids* approach delivered the best results for Task 3 in terms of *nDCG@10*. Overall, our findings support the potential of annealing-based techniques to deliver effective trade-offs between performance and computational efficiency in realistic machine learning pipelines.

Keywords

Quantum Computing, Quantum Annealing, Quantum NLP, CLEF, Instance Selection, Clustering

1. Introduction

The *Quantum CLEF (QClef) Lab* at *CLEF 2025* [1, 2] explores the integration of quantum computing technologies into real-world information retrieval and machine learning workflows. The lab serves as a benchmark initiative to assess how quantum-inspired and quantum-native methods can enhance computational efficiency and effectiveness across core AI tasks. By leveraging both quantum and simulated annealers, QClef aims to investigate the suitability of these technologies for structured and unstructured data analysis.

QClef 2025 is organized into three primary tasks: *Task 1 - Feature Selection*, *Task 2 - Instance Selection*, and *Task 3 - Clustering*. Each task presents a different machine learning challenge, offering opportunities to apply quantum approaches in data preprocessing, optimization, and representation learning.

Our team, *GPLSI*, participated in *Task 2* and *Task 3*. The main hypothesis guiding our work is that quantum-inspired optimization methods can lead to more representative and compact data subsets, thereby improving downstream learning and analysis tasks such as classification and clustering. In particular, we explore the utility of Quadratic Unconstrained Binary Optimization (QUBO) formulations and hybrid clustering strategies to address instance selection and structure discovery in sentiment analysis datasets.

CLEF 2025 Working Notes, 9 – 12 September 2025, Madrid, Spain
*Corresponding author.
†These authors contributed equally.
juan.consuegra@ua.es (J. P. Consuegra-Ayala); aitana.morote@ua.es (A. Morote-Martínez); francisco.valero@ua.es (F. Valero-Abellón); elloret@dlsi.ua.es (E. Lloret); moreda@dlsi.ua.es (P. Moreda); mpalomar@dlsi.ua.es (M. Palomar)
0000-0003-2009-393X (J. P. Consuegra-Ayala); 0009-0002-1694-8288 (A. Morote-Martínez); 0009-0002-4739-2171 (F. Valero-Abellón); 0000-0002-2926-294X (E. Lloret); 0000-0002-7193-1561 (P. Moreda); 0000-0002-1441-7865 (M. Palomar)

Our specific objectives are:

- To develop and compare three instance selection approaches for Task 2.
- To design a quantum-inspired clustering method for Task 3.
- To evaluate all methods using the official datasets and protocols provided by QClef 2025.

The remainder of this report is structured as follows. Sections 2 and 3 detail our methodology for instance selection and clustering, respectively. Section 4 presents the experimental setup and discusses results and observations. Finally, Section 5 concludes with a summary and future directions.

2. Methodology —Task 2: Instance Selection

This section outlines the methodology employed for Task 2, which focuses on instance selection (IS) aimed at improving the efficiency and performance of downstream sentiment analysis models. We explore three distinct approaches, each designed to identify representative and informative subsets of data. Section 2.1 introduces a pairing strategy based on sentiment polarity and semantic similarity to capture nuanced contrasts. Section 2.2 details a technique that leverages local neighborhood bounded by the local set-concept for instance selection. Finally, Section 2.3 presents a hybrid approach combining heuristic pre-filtering with statistical optimization. Together, these methodologies offer complementary perspectives on selecting high-quality training instances.

2.1. Approach I —Sentiment Pairs

This approach focuses on selecting a balanced and representative subset of sentiment-labeled instances by leveraging clustering and quantum-inspired optimization. The goal is to construct a reduced dataset that maintains semantic diversity and class balance. The method consists of three main stages: (I) balancing through clustering, (II) pair construction and selection, and (III) subset optimization via QUBO.

2.1.1. Balancing via K-Medoids Clustering

To ensure a balanced set of instances from both sentiment classes, we first independently apply *K-Medoids clustering* [3] to the positive and negative subsets of the dataset. K-Medoids is chosen for its robustness in identifying representative instances (medoids), which are actual data points that minimize within-cluster dissimilarity.

Let $\mathcal{D}^+$ and $\mathcal{D}^-$ be the sets of positive and negative instances, respectively. We apply K-Medoids to each set to obtain:

$$\mathcal{M}^+ = \text{KMedoids}(\mathcal{D}^+, k), \quad \mathcal{M}^- = \text{KMedoids}(\mathcal{D}^-, k)$$

where k is the number of desired medoids per class. The value of k controls the number of selected instances from each class and ensures that $|\mathcal{M}^+| = |\mathcal{M}^-|$, enforcing class balance.

2.1.2. Pair Construction and Selection

After balancing the dataset through medoid extraction (Section 2.1.1), we construct cross-class pairs by combining positive and negative medoids in a greedy manner. Each positive medoid is paired with its closest unpaired negative medoid based on cosine similarity in the feature space. This process ensures that each negative medoid is used at most once, and the resulting sequence of pairs reflects a decreasing order of similarity —from the most semantically aligned pairs to less similar ones.

Formally, let:

- $\mathcal{M}^+ = \{m_1^+, \ldots, m_k^+\}$ be the set of positive medoids,
- $\mathcal{M}^- = \{m_1^-, \ldots, m_k^-\}$ be the set of negative medoids,
- $\text{sim}(m_i^+, m_j^-)$ denote the cosine similarity between any two medoids.

We initialize an empty set $\mathcal{P} = \emptyset$. For each $m_i^+ \in \mathcal{M}^+$, we select:

$$m_j^- = \arg \max_{m^- \in \mathcal{M}^- \setminus U} \text{sim}(m_i^+, m^-)$$

where U is the set of already paired negative medoids. We then add the pair (m_i^+, m_j^-) to $\mathcal{P}$ and update $U \leftarrow U \cup \{m_j^-\}$.

This greedy construction results in k semantically aligned positive-negative pairs, each with an associated similarity score.

Pair Selection Strategies To reduce the number of candidate pairs while maintaining semantic and similarity diversity, we apply an agglomerative clustering approach that integrates two complementary views of pair dissimilarity: (1) the difference in intra-pair similarity scores and (2) the semantic distance between mean document vectors of the pairs.

Let each pair $p_i = (m_i^+, m_i^-) \in \mathcal{P}$ consist of a positive and negative medoid, along with their cosine similarity score s_i. We define a combined distance measure between any two pairs p_i and p_j as follows:

1. **Similarity Score Distance.** The first component quantifies the absolute difference in intra-pair similarity:
$$d_{\text{sim}}(p_i, p_j) = |s_i - s_j|$$
2. **Semantic Mean Vector Distance** Each pair is represented by the mean of its two constituent document vectors:
$$\mu_i = \frac{1}{2}(m_i^+ + m_i^-)$$
The semantic distance between two pairs is then computed using cosine distance:
$$d_{\text{vec}}(p_i, p_j) = 1 - \cos(\mu_i, \mu_j)$$
3. **Combined Distance Metric** The final distance between pairs is given by the average of the two components:
$$d(p_i, p_j) = \frac{1}{2}\left(d_{\text{sim}}(p_i, p_j) + d_{\text{vec}}(p_i, p_j)\right)$$

Using this combined pairwise distance matrix, we apply Agglomerative Clustering [4, 5] with average linkage and a precomputed distance metric. This clustering groups pairs that are both semantically similar and close in intra-pair similarity score. From each resulting cluster, a single representative pair is selected. The representative is defined as the most central pair within its cluster, i.e., the one with the minimum total distance to all other members in the cluster:

$$p_c^* = \arg \min_{p_i \in \text{cluster}_c} \sum_{p_j \in \text{cluster}_c} d(p_i, p_j)$$

This yields a final set $\mathcal{P}' \subseteq \mathcal{P}$ of K representative and diverse pairs that form the candidate pool for the subsequent QUBO-based optimization (Section 2.1.3). By integrating both intra-pair and inter-pair semantics, this selection process ensures that the reduced dataset maintains coverage over the underlying structure of the sentiment space.

2.1.3. QUBO Formulation for Subset Selection

Once a reduced set of candidate cross-class pairs $\mathcal{P}'$ is obtained (Section 2.1.2), we aim to select a representative subset that is both informative and non-redundant. While auxiliary procedures such as K-Medoids clustering for balancing, greedy algorithms for pair construction, and agglomerative clustering for pair selection operate within polynomial time complexity, the overall problem of instance selection often involves solving a combinatorial core that is inherently hard. In our case, this core is formulated as a Quadratic Unconstrained Binary Optimization (QUBO) problem [6, 7], which is NP-hard.

Therefore, despite relying on efficient polynomial-time heuristics to pre-process or structure the data, the final optimization step still represents a significant computational challenge. This justifies the use of quantum or quantum-inspired techniques, such as quantum annealing, to solve the QUBO formulation more effectively than classical brute-force or metaheuristic approaches, especially when scalability and solution quality are critical. The QUBO formulation allows us to encode the trade-off between the relevance of individual pairs and the redundancy among them, and solve the resulting combinatorial problem using classical or quantum annealing techniques.

Relevance Score The relevance (or importance) of each pair is quantified based on how far its similarity score deviates from the average of the minimum and maximum similarity scores among all selected pairs. This discourages the selection of semantically central pairs (i.e., neither too similar nor too dissimilar):

Let s_i denote the cosine similarity score of pair $p_i \in \mathcal{P}'$, and let:

$$c = \frac{\min_{p_i \in \mathcal{P}'} s_i + \max_{p_i \in \mathcal{P}'} s_i}{2}$$

Then, the relevance score for p_i is defined as:

$$\text{relevance}_i = |s_i - c|$$

This assigns higher importance to pairs that diverge from the central similarity mass, promoting diversity in semantic strength.

Redundancy Score To penalize semantic overlap among selected pairs, we define a redundancy score that measures similarity between every pair of pairs. For two candidate pairs p_i and p_j, redundancy is computed as the sum of cosine similarities across three views:

1. Between their positive document vectors (m_i^+).
2. Between their negative document vectors (m_i^-).
3. Between their mean vectors (μ_i).

The total redundancy between p_i and p_j is:

$$\text{redundancy}_{i,j} = \cos(m_i^+, m_j^+) + \cos(m_i^-, m_j^-) + \cos(\mu_i, \mu_j)$$

This encourages diversity by penalizing the inclusion of multiple pairs that represent similar semantic regions in the embedding space.

QUBO Objective Finally, we define a QUBO objective function that combines relevance and redundancy:

$$\min_{\mathbf{x} \in \{0,1\}^K} \sum_{i=1}^{K} -\text{relevance}_i \cdot x_i + \sum_{i=1}^{K} \sum_{j=1}^{K} \text{redundancy}_{i,j} \cdot x_i x_j + \text{penalty}$$

where $x_i \in \{0, 1\}$ indicates whether pair p_i is selected, K is the total number of candidate pairs, the first term rewards the inclusion of relevant pairs, and the second term penalizes redundant selections. To restrict the number of selected pairs to k', we add a soft constraint: $\gamma \left(\sum_i x_i - k'\right)^2$. This term is incorporated into the QUBO as an additional quadratic penalty.

2.1.4. Submission Configuration Summary

All submissions—labeled as `SentimentPairs`—are based on variations of key parameters used in the instance selection and QUBO optimization pipeline. The two main configurations differ in how output pairs are compiled—either from the final QUBO solution only (`just-final`) or from QUBO-selected

pairs enriched with related instances (`pair-related`). In the latter case, related instances refer to pairs that belong to the same cluster as the QUBO-selected pairs, based on the clustering step used to reduce the candidate set from k to K (Section 2.1.2). Each submission uses a fixed number of annealing reads (2000), enforces a balance limit k of 90% of the minority class size, sets the number of candidate cross-class pairs $K = 150$, and restricts the QUBO-selected final subset to $k' = 0.5 \times K = 75$ pairs. Table 1 summarizes the explored configurations.

Table 1
Parameter settings for the experimental submission variants of the *SentimentPairs* approach.

Dataset	Output Compilation	# Reads	k (Balance Limit)	K (Pair Limit)	k' (QUBO Limit)
Vader	just-final	2000	0.9 × 1763 = 1586	150	150 / 2 = 75
	pair-related	2000	0.9 × 1763 = 1586	150	150 / 2 = 75
Yelp	just-final	2000	0.9 × 2000 = 1800	150	150 / 2 = 75
	pair-related	2000	0.9 × 2000 = 1800	150	150 / 2 = 75

2.2. Approach II —Local Sets

The main idea of this method is to exploit the concept of local set [8], a geometrical construct describing the instance neighborhood, to select an optimal set of instances for training a classification model. Our approach extends the *Local Set Border Selector (LSBo)* [9], a hybrid method composed by a noise filtering step and the condensation of the noise free database. It also benefits of a modified version of a supervised clustering algorithm based on local sets, *LS-clustering* [10], to prepare the dataset for the selection using the annealing technique. Throughout our method, distances between instances are computed as euclidean distances.

2.2.1. Instance Taxonomy

Our approach relies on a categorization of instances based on their contribution to classification accuracy, where each category can be quantified using a measure based on local sets.

- **Noise Instances** Noise instances disagree in classification with their neighborhoods, thereby obscuring the relationship between the features of an instance and its label [11]. Datasets like Yelp Reviews [12] and Vader NYT [13], where labels originate from subjective human ratings, are prone to such noise. Within the *Local Set-Based Smoother (LSSm)* framework [9], noise instances are identified with a measure of *harmfulness*, tied to how many instances identify them as their nearest enemy.
- **Redundant/Useless Instances** By definition, these instances do not significantly influence classification. Redundancy arises when excessive same-class neighbors exist. LSSm characterizes redundancy with the *usefulness* measure, based on the number of local sets an instance belongs to.
- **Border instances** Instances near class borders can be identified within the LSBo framework as instances having the lowest *Local Set Cardinality (LSC)* among the members of their local sets.
- **Central/typical instances** Typical instances, also called prototypes, are instances central enough to represent a local region. They are determined within the scope of the *Local Set-based Centroids selector (LSCo)* [9], which selects a set of centroids of clusters obtained with LS-clustering.

In the instance selection literature, methods classified as *edition* methods are typically directed at removing noise, while *condensation* methods aim at removing redundancy. Methods combining both, such as our method, are called *hybrid* [14]. Generally, the selection criterion also fluctuates between focusing on border instances or central instances, as both contribute positively to classification accuracy. Finding an optimal criterion is subject to the dataset characteristics [10]. Our method is a combination of both main criteria, as we retain both a set of border instances and a set of central instances.

2.2.2. Method Objective

Our method aims to enhance the LSBo, which already achieves a good tradeoff between accuracy and reduction, supported with empirical evidence [9], by selecting an additional set of instances through a high complexity step executed with quantum annealing.

To maximize accuracy with this additional set, rather than prioritizing centrality, our strategy will be to select the less-redundant set among the non-border instances. Our heuristic to detect redundancy is based on the distribution of same-class instances in local regions delimited by borders. More concretely, a pair of same-class instances (or clusters, as will be explained hereafter) will be awarded or penalized for selection by comparing their inner distance to the sum of the distances to their relative nearest border.

The complete algorithm pseudocode can be found at Algorithm 1.

Algorithm 1 Method for Instance Selection based on Local Sets for a Quantum Annealer

Require: Set of instances T
Require: Number of decision variables k

Set of selected instances $S \leftarrow \emptyset$
Set of preselected clusters $C' \leftarrow \emptyset$
Set of selected clusters $C^* \leftarrow \emptyset$

1. Noise Removal
$N \leftarrow \text{LSSm}(T)$
$T \leftarrow T \setminus N$ *Removing noise instances*

2. Border Detection
$B \leftarrow \text{LSBo}(T)$
$S \leftarrow B$ *Selecting border instances*

3. Clustering
$T \leftarrow T \setminus B$
Set of clusters $C \leftarrow$ Modified LS-Clustering(T)

4. Cluster Preselection
Compute nearest same-class border for each cluster
Sort clusters in ascending order of distance to their nearest border
for i = 1 to k **do**
$C' \leftarrow C' \cup \{C[i]\}$ *Keeping only the k clusters closest to borders*
end for

5. Quantum Selection
Build QUBO matrix $Q(C')$
$C^* \leftarrow \text{QA}(Q)$ *Removing redundant clusters with quantum annealing (QA)*
for each $c^* \in C^*$ **do**
$S \leftarrow S \cup \{c^*\}$ *Adding selected cluster members*
end for

return S

In the following sections, the algorithm steps 3, 4 and 5 will be explained in detail. A more complete presentation of the algorithms LSSm and LSBo of steps 1 and 2 can be found in the paper [9]. Section 2.2.3 details the method used for clustering, Section 2.2.4 presents our heuristic for preselecting clusters and Section 2.2.5 explains how we obtain the weights of the QUBO matrix determining the final selection.

2.2.3. Modified LS-Clustering

Our strategy to reduce the problem size is to associate each decision variable to a cluster containing same-class local regions of instances. To that end, non-border, non-noisy instances are clustered using a modified version of the local set-based clustering [10]. The original algorithm selects local sets with a high LSC, resulting in clusters containing wide same-class regions. We modify the clustering algorithm by processing small local sets first, obtaining numerous clusters containing smaller patches of instances of the same region. This will allow us to select the less redundant patch of each region.

The algorithm pseudocode can be found at Algorithm 2.

Algorithm 2 Modified Local Set-based Clustering

Require: Set of instances T
Require: Set of noise instances N and border instances B

```
Set of clusters C ← ∅
T ← T \ (B ∪ N)                                   Filter noise and border instances
Compute the LSs (Local Sets) for each instance in T
Sort instances in ascending order of LSC (Local Set Cardinality)   Smaller local sets are processed first
T_added ← ∅                                       Instances already added to a cluster
for each t ∈ T do
    if t ∉ T_added then
        T_added ← T_added ∪ {t}
        Create cluster c with medoid T and members LS(t) \ T_added
        T_added ← T_added ∪ LS(t)
        Add c to C
    end if
end for
return C
```

2.2.4. Cluster Preselection

Due to the hardware constraints, the number of problem decision variables is restricted to 150, meaning that the quantum annealer can handle at most 150 candidate clusters at once. To meet this constraint, we adopt a heuristic that preselects the 150 clusters whose centroids are closest to their nearest same-class border instance. The process is as follows:

1. For each cluster c_i, compute its centroid μ_i as the mean of its member instances.

$$\mu_i = \frac{1}{|c_i|} \sum_{x \in c_i} x$$

2. Let B be the set of border instances obtained using the LSBo method, and $y_i \in \{0, 1\}$ the binary class of each cluster. For each cluster c_i, find the distance d_i^* to the nearest same-class border instance $b_i \in B$.

$$d_i^* = \min_{b_i \in B,\, y_{b_i} = y_i} \|\mu_i - b_i\|^2$$

3. Select the 150 clusters with the smallest distances d_i^* to form the preselection set.

2.2.5. Cluster Selection using Quantum Annealing

The time complexities of the different algorithms described so far —LSBo, modified LS-clustering and algorithm for computing nearest borders— are $\mathcal{O}(T^2)$, $\mathcal{O}(T)$ and $\mathcal{O}(CBd)$ respectively, where T represents the set of initial instances, C represents the total number of clusters, B the number of border instances and d the feature dimensionality. Therefore, running these algorithms before the step that will be executed in the quantum annealer does not compromise efficiency.

The problem of selecting the less-redundant set of clusters based on their location around border instances is formulated using a QUBO model. Specifically, a pair of same-class clusters is rewarded if the distance between their centroids is greater than the sum of their respective distances to the nearest border instance, as such cases are likely to correspond to clusters belonging to distinct local regions separated by borders. The QUBO formulation is described below:

Let each preselected cluster be associated with:

- Centroid μ_i,
- Nearest border distance d_i^*,
- Class label $y_i \in \{0, 1\}$.

We define the inter-cluster distance and the separation margin δ_{ij} as:

$$\text{inter\_dist}_{ij} = \|\mu_i - \mu_j\|, \quad \delta_{ij} = \text{inter\_dist}_{ij} - (d_i^* + d_j^* + T)$$

where $T \geq 0$ is a tolerance parameter that increases the required separation between same-class clusters to avoid being considered redundant. Larger values of T reflect stricter redundancy criteria. For both datasets (Vader NYT and Yelp Reviews), we use a value of 0.2 for the parameter T.

We define the QUBO matrix $Q \in \mathbb{R}^{n \times n}$ as:

$$Q[i,j] = \begin{cases} \text{self\_bias} & \text{if } i = j \\ \alpha \cdot \dfrac{1}{\text{inter\_dist}_{ij} + \varepsilon} & \text{if } \delta_{ij} < 0 \text{ and } y_i = y_j \\ -\beta \cdot \dfrac{1}{\text{inter\_dist}_{ij} + \varepsilon} & \text{if } \delta_{ij} \geq 0 \text{ and } y_i = y_j \\ 0 & \text{if } y_i \neq y_j \end{cases}$$

with the following parameters (we use the same parameter values for both datasets):

- α = 1.0: penalty scaling factor for redundant same-class cluster pairs,
- β = 1.0: reward scaling factor for well-separated same-class cluster pairs,
- `self_bias` = -1.0: negative diagonal term that promotes sparsity in the selection.
- ε = 10^{-6}: small constant added to avoid division by zero.

To enforce class balance in the selected set of clusters, we add a penalty to the QUBO formulation:

$$Q \leftarrow Q + \gamma \cdot \mathbf{w}\mathbf{w}^\top \quad \text{where } w_i = 1 - 2y_i$$

where γ regulates the strength of the penalty term. We use a value of $\gamma = 5.0$ for both datasets. The sum of the terms $1 - 2y_i$ for all clusters c_i will be minimized when there is the same number of clusters of each class in the final solution. As the optimal solution is the solution that minimizes the expression Q, adding this term with a high enough γ allows us to promote class balance. After obtaining the final cluster selection, all the selected cluster member instances are added to the submission set containing the border instances.

2.3. Approach III —Pre-Selection and Statistical Tuning

We propose *PREST*, *Pre-Selection and Statistical Tuning*, for QUBO Annealing a three-step pipeline that begins with classical heuristics and ends with the exact same optimization task running on both CPUs and quantum hardware. First, we build candidate panels greedily and wrap them into a single QUBO [15]. Because the objective and hyper-parameters stay identical, the classical-versus-quantum comparison is strictly like-for-like. The workflow comprises three stages:

1. **Stage I – Representation & Pre-filtering**: Sentence-BERT embeddings are enriched with scalar features. Nine classical heuristics generate candidate subsets for a grid of target sizes k. A metric bundle $\{\text{silhouette}, \text{Davies–Bouldin}, \text{balance}, \text{diversity}, \text{coverage}\}$ [16] ranks the outputs.
2. **Stage II – Combinatorial Optimization**: each of the four best heuristics selected in Stage I is cast as a QUBO whose coefficients encode *diversity, class-frequency fairness, emotional contrast* g_i*, and sentiment confidence* s_i. A SA sweep over k and QUBO weights keeps the two top-scoring panels per heuristic.
3. **Stage III – Quantum-Ready Aggregation**: To comply with the ≤ 150-variable limit of current quantum annealers, the retained subsets are clustered into *super-nodes*. We solve two QUBO variants—unconstrained and cardinality-constrained—with both SA and QA in a D-Wave QPU, yielding the final instance sets.

2.3.1. Stage I — Representation and Pre-filtering

Augmented sentence embeddings Each sentence is embedded with the `all-mpnet-base-v2` Sentence-BERT model [17], producing a 768-dimensional vector. To capture additional information we concatenate two task-specific scalars: **emotional contrast** (g_i) defined as the gap between the two highest emotion logits, and **sentiment confidence** (s_i), given by the maximum soft-max probability produced by the sentiment classifier.

Subset generators We generate nine deterministic subsets for each target size $k \in \{200, 400, \ldots, 2000\}$, leveraging a range of complementary heuristics grounded in core-set theory, graph analysis, and uncertainty modeling. Classical core-set methods—*Max–Min*, *K-Means Centroids*, and *Farthest-First*—are commonly used for feature selection in Natural Language Processing (NLP) [18]. To exploit structural cues, we include *Community selection* via modularity maximization [19] and *Closeness Centrality* [20], two graph-based sampling heuristics. The *Density-Weighted* heuristic prioritizes low-density regions in the embedding space, enhancing representational diversity by sampling from semantically sparse areas [21]. *Emotional Conflict*, a custom heuristic introduced in this work, identifies semantically similar sentence pairs with opposing affective signals—highlighting emotionally ambiguous cases relevant to affective representation [22]. *Diversity Sampling* maximizes coverage by iteratively picking points farthest from the current subset, thereby enhancing generalization and reducing feature redundancy [23]. Finally, *Uncertainty-Based* selection focuses on items near sentiment boundaries, following Bayesian active-learning and margin-sampling principles [24].

Metric-Driven Filtering Each subset is evaluated with five criteria [16] —silhouette ($\mathcal{S}$), Davies–Bouldin ($\mathcal{D}$), class balance ($\mathcal{B}$), diversity ($\mathcal{V}$), and coverage ($\mathcal{C}$). For every dataset we gather the raw scores of all nine heuristics and all target sizes k, compute the global mean μ_m and standard deviation σ_m of each metric m, and obtain z-scores $z_m = (m - \mu_m)/\sigma_m$. The normalized scores are finally averaged into $\Psi = \frac{1}{5}\big(\mathcal{S} - \mathcal{D} + \mathcal{B} + \mathcal{V} + \mathcal{C}\big)$, and the four highest-ranked heuristics progress to Stage II.

2.3.2. Stage II — Combinatorial Optimization

After the Stage I filtering we retain *Max–Min*, *K-Means Centroids*, *Emotional Conflict* and *Uncertainty-Based* . For each pool $\mathcal{P} = \{p_1, \ldots, p_n\}$ and target size $k \in \{1000, 1500, 2000, 2500\}$ (a reduced grid chosen for runtime reasons) we flip a binary switch $\mathbf{x} \in \{0, 1\}^n$ ($x_i = 1$ selects p_i). The QUBO combines four signals:

- **Diversity** d_{ij}, amplified when the two sentences come from different sentiment classes and inverted when they coincide.
- **Class-frequency fairness**: diagonal bias n_{y_i}/n, where n_{y_i} is the number of instances in class y_i and n is the total number of instances, that favours minority classes.
- **Emotional contrast** $\lambda_{emo}\, g_i$ on the diagonal;
- **Sentiment confidence** $\lambda_{sent}\big(1 - |s_i - 0.5|\big)$ on the diagonal (prefer items whose softmax probability is close to 0.5).

The QUBO matrix Q is

$$Q_{ij}^{\text{unc}} = \begin{cases} \dfrac{n_{y_i}}{n} + \lambda_{emo}\, g_i + \lambda_{sent}\big(1 - |s_i - 0.5|\big), & i = j, \\ d_{ij}, & i \neq j,\ y_i \neq y_j, \\ -d_{ij}, & i \neq j,\ y_i = y_j. \end{cases}$$

Hyper-parameter sweep We explore $\lambda_{emo} \in \{0.5, 1.0, 2.0, 2.5\}$ and $\lambda_{sent} \in \{0.25, 0.5, 1.0\}$, crossed with the four k values, i.e. $4 \times 3 \times 4 = 48$ QUBOs per heuristic.

Optimization Each pair $(k, \boldsymbol{\lambda})$ defines a QUBO instance, solved with Neal's SA sampler (100 reads, default settings). Panels are re-scored with the same metric bundle and the two best configurations per heuristic are retained for Stage III.

2.3.3. Stage III — Quantum-ready Aggregation

Only the subsets generated by the *K-Means Centroids* and *Emotional Conflict* heuristics outperformed all others in Stage II across all datasets, so we focus on those two for the quantum run. Both methods generate datasets that exceed the variable budget of current QPUs, so each is compressed into $C = 150$ *super-nodes* via k-means and reformulate the problem as a second QUBO optimized with both classical Simulated Annealing (SA) and Quantum Annealing (QA). A super-node $\mathcal{S}_c = \langle \bar{\mathbf{z}}_c, \hat{y}_c, \bar{g}_c, \bar{s}_c \rangle$ stores centroid, majority class, mean emotional-contrast and mean sentiment-confidence.
To keep the search flexible, we run the QUBO in two versions: one gently caps the solution at roughly half of the available super-nodes (cardinality-constrained), while the other drops that limit altogether and lets the annealer decide for itself how many super-nodes are worth keeping.

QUBO without cardinality constraint Let $C = |\{\mathcal{S}_c\}|$ and $\mathbf{x} \in \{0,1\}^C$. For super-nodes i, j define the class indicator $\delta_{ij}^{(y)} = 1$ if $\hat{y}_i = \hat{y}_j$ and 0 otherwise, and the Euclidean distance $d_{ij} = \|\bar{\mathbf{z}}_i - \bar{\mathbf{z}}_j\|_2$. Following our implementation `build_qubo_qclef_no_cardinality()`, the unconstrained QUBO reads

$$Q_{ij}^{\text{unc}} = \begin{cases} \dfrac{n_{\hat{y}_i}}{C} + \lambda_{emo}\,\bar{g}_i + \lambda_{sent}\big(1 - |\bar{s}_i - 0.5|\big), & i = j, \\ d_{ij}, & i \neq j,\ \delta_{ij}^{(y)} = 0, \\ -d_{ij}, & i \neq j,\ \delta_{ij}^{(y)} = 1. \end{cases}$$

Here $n_{\hat{y}_i}$ is the frequency of class $\hat{y}_i$ in the panel; λ_{emo} and $\lambda_{sent} \geq 0$ weight emotional contrast and sentiment confidence, respectively.

QUBO with cardinality constraint [25] To steer the solution towards $k = \lceil C/2 \rceil$ selected super-nodes we build a Binary Quadratic Model (BQM) from Q^{unc} and add a combination penalty generated with `dimod.generators.combinations`. Its strength is set automatically to the maximum energy spread of the unconstrained BQM:

$$Q_{ij}^{\text{card}} = Q_{ij}^{\text{unc}} \;+\; \alpha\big(k - \sum_{\ell} x_\ell\big)^2, \qquad \alpha = \texttt{bqm.maximum\_energy\_delta()}.$$

Optimization and evaluation Both QUBOs are solved independently with **SA** (same schedule as Stage II) and **QA** (2 000 reads, default settings). The binary solutions are projected back to their original sentences and rescored with the five-metric bundle.

3. Methodology —Task 3: Clustering

This section outlines the methodology employed for Task 3, which focuses on centroid selection aimed at accelerating document retrieval over the ANTIQUE sentence-embedding corpus. We investigate four complementary clustering pipelines, each designed to generate a small set of representative centroids that preserve cluster quality and maximize downstream retrieval effectiveness.

Our hybrid, QA-driven *k*-medoids pipeline unfolds in three steps:

1. **Pivot selection**: we start from the 6 513 sentence embeddings provided with the ANTIQUE corpus; this full set of $N \approx 6500$ vectors is filtered down to $P = 150$ well-spread candidate pivots.
2. **Annealing optimization** : cast a QUBO on those P pivots to identify the final k medoids.
3. **Global assignment**: assign every document to its nearest medoids, producing the final k clusters.

3.1. QUBO for the Reduced *k*-Medoids (Bauckhage's Formulation)

Following the standard QUBO framework [15] and Bauckhage's quadratic reduction [26], we embed the P pivots as rows of $\mathbf{X} \in \mathbb{R}^{P\times d}$ (ℓ_2-normalized). Let $\delta_{ij} = 1 - \cos(\mathbf{x}_i, \mathbf{x}_j)$ and introduce binary $y_j \in \{0, 1\}$, with $y_j = 1$ if pivot j is selected. The k-medoids objective becomes the pseudo-Boolean function

$$\gamma\Big(\sum_j y_j - k\Big)^2 + \frac{1}{k}\sum_{i<j} \delta_{ij}\, y_i\, y_j,$$

which expands to the QUBO

$$\sum_{i\le j} Q_{ij}\, y_i\, y_j \;+\; \gamma k^2, \quad Q_{ij} = \begin{cases} \gamma - \frac{\alpha}{2}\,\delta_{ij}, & i \neq j, \\ \gamma - \frac{\alpha}{2}\,\delta_{ii} + \beta\sum_{u=1}^{P}\delta_{iu} - 2\gamma k, & i = j, \end{cases}$$

where we fix $\alpha = 1/k$, $\beta = 1/P$, and $\gamma = 2k$, balancing dispersion, linear penalties, and enforcing $\sum_i y_i = k$. The parameter γ acts as a Lagrange multiplier, which is crucial in balancing the minimization of the objective function against the constraint that the number of selected medoids is equal to k. This same parameter set is kept across all four methods (A–D) for both the exact-P reduction and the final exact-k optimization. These values are chosen according to the guidelines provided by Bauckhage et al. [26] to ensure a balanced contribution of the different terms in the objective function. The resulting `BinaryQuadraticModel` is submitted to a D-Wave annealer (2 000 reads); the returned vector $\mathbf{y}^*$ flags the selected medoids.

This Bauckhage QUBO offers several practical advantages: by specifying only its upper-triangular entries, the matrix is symmetric ($Q_{ij} = Q_{ji}$), simplifying minor embedding on quantum hardware; the diagonal shift $-2\gamma k$ cancels the linear-term bias from the squared cardinality penalty, avoiding skew toward too many or too few pivots; the constant offset E_0 maintains equivalence with the original objective yet can be dropped after solving without affecting cluster assignments; and, since $\delta_{ii} = 0$ and most off-diagonal δ_{ij} are small, Q is naturally sparse, reducing the solver's memory footprint.

3.2. Pivot-Selection Strategies

With the QUBO core fixed, we isolate the effect of classical preprocessing by testing four distinct pivot-selection strategies that each deliver $P = 150$ candidates.

Method A: *qIIMAS strategy (SubMedoids).* Following the technique introduced by team qIIMAS in last year's competition [27], each document is assigned a binary variable $s_i \in \{0, 1\}$ indicating whether it remains a pivot. An "exact-P" QUBO with cost matrix $Q'_{ij} = \delta_{ij}/P$ forces the cardinality constraint $\sum_i s_i = P$ while minimizing pairwise similarity within the chosen subset. Solving this model on the same annealing backend used downstream yields a maximally diverse pivot set that captures even the rarest semantic regions of the corpus. Any surplus indices are deterministically trimmed, whereas deficits are filled with previously unused, randomly sampled documents to ensure $|I| = P$ before the quantum stage.

Method B: *Farthest-Point Sampling (FPS).* FPS adopts a simple yet powerful greedy rule: starting from one random seed, it maintains each document's distance to its nearest selected pivot (measured in cosine distance) and repeatedly adds the document with the largest recorded distance until exactly P pivots have been chosen [28]. Each iteration is a single pass over the corpus, yielding a cost of $\mathcal{O}(NP)$ time and $\mathcal{O}(N)$ memory. The resulting pivots provide broad geometric coverage, so the subsequent QUBO receives a well-spread candidate pool for the final *k*-medoids step.

Method C: *MiniBatch-KMeans.* Running MiniBatch-KMeans [29] with P clusters, we pick as pivot the document closest to each centroid μ_p in cosine distance. These representatives reflect data density, furnishing the annealer with high-quality starting points at a per-epoch cost of $\mathcal{O}(Nd)$.

Method D: *CLARA–CLARANS hybrid.* We combine two classic medoid heuristics in a single, two-step routine. *CLARA* first draws several disjoint subsamples (each roughly 20 % of the corpus), applies PAM to every subsample, and retains the P medoids that minimise the within-subset distance. Building on this provisional set, *CLARANS* performs up to twenty random medoid–non-medoid swaps, accepting a swap only when it reduces the overall clustering cost [30]. The combination lets CLARA explore diverse regions of the embedding space, while CLARANS fine-tunes the most promising configuration, yielding a pivot set that achieves both broad coverage and strong internal cohesion.

Experimental workflow common to all methods (A–D)

For every pivot selector—*qIIMAS strategy* (A), FPS (B), MiniBatch-KMeans (C), and CLARA–CLARANS (D)—and each target size $k \in \{10, 25, 50\}$, we run the six-step pipeline below:

1. **Initial set-up.** Start with the full corpus of n ℓ_2-normalized sentence embeddings and fix k.
2. **Coarse compression.** Apply the chosen method to obtain $m \ll n$ coarse clusters; their centroids define three candidate pivot budgets $P \in \{100, 125, 150\}$. Validation retains $P{=}150$ for A and $P{=}100$ for B–D (seed = 42).
3. **Fast QUBO sweep (100 SA reads).** For every $\langle k, \text{method}, P\rangle$ build the exact-P QUBO, solve it once with 100 SA reads, and log validation Davies–Bouldin and nDCG@10.
4. **Pivot-budget selection.** Choose the P that minimizes Davies–Bouldin (nDCG@10 breaks ties); this reproduces the preferences in step 2.
5. **Full Annealing run (2 000 reads).** Rebuild the exact-k QUBO with the selected P and solve it with 2 000 reads on the designated annealer (SA or QA) to obtain the k medoids.
6. **Global assignment and metrics.** Attach each of the n embeddings to its nearest medoid, then compute final Davies–Bouldin and mean nDCG@10.

4. Experiments

This section presents the experimental setup and results for evaluating our proposed approaches across two main tasks: *Task 2 (Instance Selection)* and *Task 3 (Clustering)*. We detail the evaluation scenarios, including datasets, submission configurations, and whether Simulated Annealing (SA) or Quantum Annealing (QA) was used. The results are reported as computed by the challenge organizers. Finally, we provide a discussion analyzing the trade-offs between performance and efficiency achieved by our methods compared to baselines and competing submissions.

4.1. Evaluation Scenarios

The evaluation was divided into two main scenarios corresponding to the official tasks defined in the challenge. The selected tasks were: (i) **Task 2: Instance Selection** and (ii) **Task 3: Clustering**. Each task involved the use of SA and QA approaches.

4.1.1. Task 2: Instance Selection

Task 2 focuses on selecting the most representative subsets of training data for fine-tuning a sentiment classification model (Llama3.1), aiming to reduce computational cost while preserving effectiveness. The evaluation considered both the macro F_1 score and the reduction rate. We submitted several configurations using both SA and QA methods on the Vader NYT and Yelp datasets. Table 2 summarizes our submitted runs for Task 2, indicating which approaches were applied to each dataset using SA and QA.

Table 2
Summary of submissions for Task 2: Instance Selection. Columns *SA* and *QA* stand for Simulated Annealing and Quantum Annealing, respectively. *Incomplete submissions (< 5 folds) due to issues in the challenge's execution infrastructure.

Submission ID	Vader		Yelp	
	SA	QA	SA	QA
LocalSets	✓	✗	✓	✗
emoconflictCard	✗	✗	✓	✓
SentimentKmeansCard	✗	✗	✓	✓
SentimentPairs(docs=pair-related,reads=2000,limit=True)	✓	✓*	✓	✗
SentimentPairs(docs=just-final,reads=2000,limit=True)	✓	✓*	✓	✗

4.1.2. Task 3: Clustering

Task 3 involves clustering sentence embeddings to support efficient document retrieval and exploration. Quality was evaluated both intrinsically with the Davies–Bouldin Index and extrinsically with nDCG@10 on test queries. We submitted clustering results on the large and small variants of the ANTIQUE dataset using both SA and QA. Table 3 summarizes our submitted runs for Task 3, indicating which approaches were applied to each dataset using SA and QA.

Table 3
Summary of submissions for Task 3: Clustering. Columns *SA* and *QA* stand for Simulated Annealing and Quantum Annealing, respectively.

Submission ID	10		25		50	
	SA	QA	SA	QA	SA	QA
FPS-Medoids	✓	✗	✓	✗	✓	✗
SubMedoidsQUBO	✓	✗	✓	✗	✓	✗
CLARA-CLARANS	✓	✗	✓	✗	✓	✗
MBK-Medoids	✓	✗	✓	✗	✓	✗

4.2. Results

Tables 4 and 5 present a summary of the results for Task 2, using the Vader and the Yelp datasets respectively. They show the performance of our approaches—**SentimentPairs (SP)** and **LocalSet** for both datasets, and **Pre-Selection and Statistical Tuning** submitted in the variants *emoconflictCard* and *SentimentKmeansCard* only for the Yelp dataset—compared to the baseline and the top-performing submission in terms of average macro F_1 score. Results are averaged over five cross-validation folds, where available. The *Evaluation Time* column combines fine-tuning and prediction time (in seconds), while the *Annealing Time* refers to the total execution time for QA approaches, including programming, sampling, and post-processing. Submissions marked with an asterisk (*) did not provide results for all five folds due to issues in the challenge's execution infrastructure, which were beyond the control of both participants and organizers. The SentimentPairs (SP) variant was executed with `reads=2000` and `limit=True`.

Table 6 summarises the Task 3 outcomes on the *ANTIQUE* corpus obtained using only SA. For each target panel size $k \in \{10, 25, 50\}$ we report the NDCG@10 and the Davies-Bouldin computed on the *entire* dataset. The *Annealing time* column aggregates programming, sampling, and post-processing of the SA search, executed with the default `neal` schedule (2000 reads, defaults settings).

Table 4
Summary of *Task 2 - Vader dataset* results (averaged over 5 folds). Types *SA* and *QA* stand for Simulated Annealing and Quantum Annealing, respectively. *Incomplete submissions (< 5 folds) due to issues in the challenge's execution infrastructure. SentimentPairs (SP): reads=2000, limit=True.

Group	SID	Macro F_1	Reduction	Evaluation Time (s)	Annealing Time (us)	Type
BASELINE	BASELINE_ALL	88.9(0.8)	–	1997.3(5.7)	–	–
HIGHEST F_1	Vader_SA_qclef_combined_075	65.9(4.7)	0.25	1529.4(3)	25299992	SA
GPLSI	Vader_SA_gplsi_2-LocalSets	63.3(4.9)	0.505	1048.3(6.7)	29109748	SA
GPLSI	Vader_SA_gplsi_2-SP(docs=pair-related)	62.2(4.1)	0.7	671.8(352.8)	42407652	SA
GPLSI	Vader_QA_gplsi_2-SP(docs=pair-related)	62.1(1.8)*	0.658*	750.7(2653.2)*	545303*	QA
GPLSI	Vader_QA_gplsi_2-SP(docs=just-final)	50(64)*	0.835*	172.9(26.9)*	545303*	QA
GPLSI	Vader_SA_gplsi_2-SP(docs=just-final)	47.4(5.4)	0.962	172.8(5.7)	42407652	SA

Table 5
Summary of *Task 2 - Yelp dataset* results (averaged over 5 folds). Types *SA* and *QA* stand for Simulated Annealing and Quantum Annealing, respectively. SentimentPairs (SP): reads=2000, limit=True.

Group	SID	Macro F_1	Reduction	Evaluation Time (s)	Annealing Time (us)	Type
HIGHEST F_1	Yelp_SA_qclef_bcos_075	99.5(0.2)	0.25	1548.5(2.8)	25996916	SA
GPLSI	Yelp_SA_gplsi_2-LocalSets	99.4(0.2)	0.512	1045.5(5.3)	28788950	SA
BASELINE	BASELINE_ALL	99.4(0.1)	–	2027.1(1.1)	–	–
GPLSI	Yelp_SA_gplsi_2-SP(docs=pair-related)	99.2(0.3)	0.627	822.2(395)	35810108	SA
GPLSI	Yelp_QA_gplsi_2-emoconflictCard	98.8(0.6)	0.702	678.8(80.9)	34024297	QA
GPLSI	Yelp_QA_gplsi_2-SentimentKmeansCard	98.7(0.2)	0.869	351(25.1)	553306	QA
GPLSI	Yelp_SA_gplsi_2-emoconflictCard	98.6(0.5)	0.728	628.2(65.9)	549364	SA
GPLSI	Yelp_SA_gplsi_2-SentimentKmeansCard	98.5(1.1)	0.875	338.8(21)	17823652	SA
GPLSI	Yelp_SA_gplsi_2-SP(docs=just-final)	90.8(5.7)	0.963	170.8(3.8)	35810108	SA

Table 6
Summary of *Task 3 - ANTIQUE dataset* results. Only Simulated Annealing (SA) results.

k	Group	Submission ID	nDCG@10	DBI	Anneal time (μs)	Type
10	BASELINE	BASELINE_10	0.5509	7.9892	–	–
10	GPLSI	10_SA_gplsi_3-FPS-Medoids	0.5783	7.5147	15374699	SA
10	GPLSI	10_SA_gplsi_3-SubMedoidsQUBO	0.5579	6.8779	15304643	SA
10	GPLSI	10_SA_gplsi_CLARA-CLARANS	0.5444	6.6710	15395337	SA
10	GPLSI	10_SA_gplsi_MBK-Medoids	0.5600	6.4258	15510116	SA
10	BEST $nDCG$@10	10_SA_ds-at-gt-qclef_1	**0.5800**	7.4776	83073	SA
10	BEST DBI	10_SA_ds-at-gt-qclef_2	0.0172	**4.4706**	82843	SA
25	BASELINE	BASELINE_25	0.5284	6.1201	–	–
25	GPLSI	25_SA_gplsi_3-FPS-Medoids	**0.5475**	5.5577	20875484	SA
25	GPLSI	25_SA_gplsi_3-SubMedoidsQUBO	0.5298	5.6255	40686713	SA
25	GPLSI	25_SA_gplsi_CLARA-CLARANS	0.5310	5.6507	20531723	SA
25	GPLSI	25_SA_gplsi_MBK-Medoids	0.5193	**5.3755**	20757746	SA
50	BASELINE	BASELINE_50	0.4656	5.3679	–	–
50	GPLSI	50_SA_gplsi_3-FPS-Medoids	**0.5592**	4.4531	9869029	SA
50	GPLSI	50_SA_gplsi_3-SubMedoidsQUBO	0.5148	4.9325	23718874	SA
50	GPLSI	50_SA_gplsi_CLARA-CLARANS	0.5017	5.1703	9976090	SA
50	GPLSI	50_SA_gplsi_MBK-Medoids	0.5383	4.5025	24003792	SA
50	BEST DBI	10_SA_ds-at-gt-qclef_3	0.0064	**3.4217**	228376	SA

4.3. Discussion

This section discusses the main findings of our participation in the QuantumCLEF Lab, organized by task. We analyze the results obtained in the instance selection task (Section 4.3.1) and the clustering task (Section 4.3.2), highlighting the effectiveness of our proposed methods, their trade-offs, and their relative performance compared to other participating systems and the baseline.

4.3.1. Task 2: Instance Selection

As shown in Table 4, our approaches—**SentimentPairs (SP)** and **LocalSet**—demonstrate effective trade-offs between data reduction and performance. Compared to the baseline, which uses the full

dataset and achieves the highest macro F_1, our best-performing approach enables a 50% reduction in data, translating into nearly half the fine-tuning and prediction time, at the cost of a moderate decrease in macro F_1. When compared to the top competing system in terms of macro F_1, our approach achieves only 2.6% lower macro F_1 on average while using half as much data, effectively doubling the reduction rate ($25\% \rightarrow 50\%$).

Among our methods, the *LocalSet* approach yields the highest macro F_1 (63.3) with an average data reduction of 50%, striking a strong balance between efficiency and accuracy. Our *SentimentPairs* variants that include *pair-related* documents trade only 1.1% of macro F_1 for up to 70% data reduction. In contrast, the strictest SentimentPairs configurations, which retain only the most relevant documents, reach reduction rates of 83–95%, though at a more pronounced cost in performance (macro F_1 drops from 62% to 47%). These results highlight the flexibility of our methods in tailoring the reduction-performance trade-off for different application needs.

The results in Table 5 also reveal a good balance between effectivity and reduction for our methods. Our best-performing method on this dataset achieves a 50% reduction of the dataset without affecting the macro F_1 score with respect to the baseline. As compared to the top-performing submission in terms of macro F_1, our approach supposes only a 0.1% decrease of this metric, and doubles the dataset reduction.

Our methods present varied and versatile solutions to the instance selection bi-objective problem using the Yelp dataset. Our *LocalSet* approach achieves the highest macro F_1 score (99.4) while halving the Yelp dataset size. Our *SentimentPairs* variant including *pair-related* documents achieves a higher reduction (62.7%), resulting in a decrease of the 59.4% of the fine-tuning and prediction time by compromising only a 0.2% of the F_1 score with respect to the baseline. The maximum evaluation time reduction (91.6%) is achieved by our *SentimentPairs* variant including *just-final* documents, at the expense of decreasing the F_1 metric only an 8.6%. Finally, our two variants of the *Pre-Selection and Statistical Tuning* method are at a middle ground between our methods achieving high performance (*LocalSet* and *SP pair-related*) and high reduction (*SP just-final*). These variants have been executed using both SA and QA. The *emoconflictCard* achieves slightly better results, with an average macro F_1 score only 0.7% behind the baseline and an average reduction of 71.5% of the dataset, resulting in an average reduction of 67.8% of the fine-tuning and prediction time.

Overall, our results in Task 2 show that it is possible to substantially reduce the size of sentiment-labeled datasets while maintaining competitive classification performance. Among our methods, the *LocalSet* approach stands out for achieving the highest effectiveness across both datasets, consistently delivering top macro F_1 scores while halving the training data. This makes it particularly attractive for scenarios requiring balanced trade-offs between performance and computational efficiency. The diversity of our approaches—ranging from aggressive reduction strategies to more conservative selections—also highlights the adaptability of quantum-inspired instance selection techniques to various application needs and constraints.

4.3.2. Task 3: Clustering

As shown in Table 6, the SA runs submitted by our team outperform the official baseline for every tested cluster size k and in every evaluation metric. Compared to other submissions, ours match or outperform them in all scenarios except when $k = 10$, where the difference is negligible. For $k = 10$, our *3-FPS-Medoids* configuration achieves an *nDCG@10* of 0.578, a relative gain of $5\,\%$ over the baseline (0.551) and virtually the same effectiveness as the best performing submission regarding *nDCG@10* (0.580). Additionally, all our variants reduce the Davies–Bouldin from 7.99 to at most 6.43; this improvement comes at a computational cost of approximately 15 s (15 M μs) of annealing time versus the 0.008 second runtime of other participant submissions. By increasing the cluster size to $k = 25$, our *3-FPS-Medoids* approach remains in the lead with *nDCG@10* = 0.547 ($+4\,\%$ over baseline), and our remaining methods collectively reduce the DBI from 6.12 to 5.56, although their runtime increases to 20–40 s (20–40 M μs). The difference increases for $k = 50$: our *3-FPS-Medoids* method reaches 0.559 *nDCG@10*, corresponding to a gain of $20\,\%$ over the baseline (0.466), and achieves the lowest DBI (4.45).

In contrast, the submission with the best *DBI* has a runtime of less than 0.3 s, but its *nDCG@10* drops dramatically to 0.006.

Overall, each one of our variant offers a superior effectiveness—cohesion profile relative to the baseline, and the SA framework scales robustly with list length, unlike the faster but markedly less stable that the other submissions approaches. Among our methods, *3-FPS-Medoids* provides the best trade-off between retrieval quality and cluster compactness; *MBK-Medoids* consistently yields the tightest clusters; *CLARA–CLARANS* occupies a middle ground, delivering effectiveness and cohesion above the baseline with the lowest runtime of our SA runs, making it attractive when computational budget is limited; and *SubMedoids-QUBO* maintains solid effectiveness at the expense of the highest computational cost.

It is important to note that, while the comparison with other submissions provides useful insights, it may not be entirely conclusive due to a structural difference in how centroids were represented. Three of our four submissions—*SubMedoids-QUBO*, *CLARA–CLARANS*, and *MBK-Medoids*—generate centroids with lower dimensionality than the original embedding. This dimensionality reduction, although beneficial for improving cluster compactness and interpretability, may have introduced a disadvantage under the evaluation protocol, particularly affecting metrics like *nDCG@10* that are sensitive to representation format. As the evaluation setup did not anticipate lower-dimensional centroids, the relative effectiveness of these methods should be interpreted with care.

5. Conclusions

This work presented our approaches and findings for the instance selection and clustering tasks defined in the *Quantum CLEF Lab*. Our methods achieved competitive results in both scenarios, obtaining top-ranked performance in multiple evaluations. In Task 2, our *LocalSets*-based strategy achieved the highest effectiveness while maintaining a substantial reduction in training data. In Task 3, our *FPS-Medoids* approach yielded the best clustering results in terms of *nDCG@10*. Overall, our results demonstrate effective trade-offs between performance and reduction, showing that it is possible to significantly reduce the amount of training data or clustering complexity without incurring major performance losses.

Key contributions of this work include the following.

- We present three multi-paradigm approaches to the instance selection problem, each leveraging complementary principles: (i) sentiment-aware document pairing, (ii) local similarity-based criteria, and (iii) classical heuristic methods for data reduction.
- We propose a unified clustering framework specifically designed for quantum annealing, which integrates four distinct pivot selection strategies to optimize centroid-based grouping in high-dimensional embedding spaces.

Future Work. Future work will focus on addressing current limitations and exploring extensions of our proposed methods. Due to infrastructure constraints during the challenge, quantum annealing could not be fully leveraged across all configurations; future experiments will aim to complete the evaluation of QA-based approaches. In Task 2, our SentimentPairs and LocalSets methods relied solely on precomputed embeddings, omitting potentially informative textual features such as readability or syntactic structure. Preliminary results from other configurations suggest that integrating custom embeddings or textual signals may enhance performance. Additionally, solving Task 2 involved tackling several clustering-related subproblems. Some of the techniques developed, such as *LocalSets*, could be adapted for Task 3. However, this would require automatically labeling the unlabeled documents in Task 3 to enable the application of supervised or semi-supervised strategies.

Acknowledgments

Funding: The research work conducted is part of the R&D projects “CORTEX: Conscious Text Generation” (PID2021-123956OB-I00), funded by MCIN/ AEI/10.13039/501100011033/ and by “ERDF A way of making Europe”; “CLEAR.TEXT:Enhancing the modernization public sector organizations by deploying Natural Language Processing to make their digital content CLEARER to those with cognitive disabilities” (TED2021-130707B-I00), funded by MCIN/AEI/10.13039/501100011033 and “European Union NextGenerationEU/PRTR”; “VIVES: “Pla de Tecnologies de la Llengua per al valencià” project (2022/TL22/00215334) as subproject of “ILENIA” project (grant number 2022/TL22/00215337) from the Ministerio de Transformación Digital y por el Plan de Recuperación, Transformación y Resiliencia – Financiado por la Unión Europea – NextGenerationEU; and the project “NL4DISMIS: Natural Language Technologies for dealing with dis- and misinformation with grant reference (CIPROM/2021/021)" funded by the Generalitat Valenciana and “GEO.IA PLATAFORMA DE GEOINTELIGENCIA ARTIFICIAL PARA RESOLVER PROBLEMAS A LOS CIUDADANOS Y FACILITAR LA TOMA DE DECISIONES ESTRATÉGICAS EN LAS ADMINISTRACIONES PÚBLICAS” (INNEST/2023/344) funded by Agencia Valenciana de la Innovación under the “Programa Comunitat Valenciana Fondo Europeo de Desarrollo Regional (FEDER) 2021-2027”. Additionally, this work is funded by the Ministerio para la Transformación Digital y de la Función Pública and Plan de Recuperación, Transformación y Resiliencia - Funded by EU – NextGenerationEU within the framework of the project Desarrollo Modelos ALIA.

Declaration on Generative AI

During the preparation of this work, the author(s) used ChatGPT, Grammarly in order to: Grammar and spelling check. After using these tools/services, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication’s content.

References

[1] A. Pasin, M. F. Dacrema, W. Cuhna, M. A. Gonçalves, P. Cremonesi, N. Ferro, Quantumclef 2025: Overview of the second quantum computing challenge for information retrieval and recommender systems at CLEF, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR Workshop Proceedings, 2025.

[2] A. Pasin, M. F. Dacrema, W. Cuhna, M. A. Gonçalves, P. Cremonesi, N. Ferro, Overview of quantumclef 2025: The second quantum computing challenge for information retrieval and recommender systems at CLEF, in: J. Carrillo-de-Albornoz, J. Gonzalo, L. Plaza, A. G. S. de Herrera, J. Mothe, F. Piroi, P. Rosso, D. Spina, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Sixteenth International Conference of the CLEF Association (CLEF 2025), Lecture Notes in Computer Science, 2025.

[3] L. Kaufman, P. J. Rousseeuw, Finding groups in data: an introduction to cluster analysis, John Wiley & Sons, 2009.

[4] F. Murtagh, A survey of recent advances in hierarchical clustering algorithms, The computer journal 26 (1983) 354–359.

[5] S. RR, A statiscal method for evaluating systematic relationships, Univ Kans sci bull 38 (1958) 1409–1438.

[6] F. Glover, G. Kochenberger, Y. Du, A tutorial on formulating and using qubo models, arXiv preprint arXiv:1811.11538 (2018).

[7] G. Kochenberger, J.-K. Hao, F. Glover, M. Lewis, Z. Lü, H. Wang, Y. Wang, The unconstrained binary quadratic programming problem: a survey, Journal of combinatorial optimization 28 (2014) 58–81.

[8] H. Brighton, C. Mellish, Advances in instance selection for instance-based learning algorithms, Data

Mining and Knowledge Discovery 6 (2002) 153–172. URL: https://doi.org/10.1023/A:1014043630878. doi:10.1023/A:1014043630878.

[9] E. Leyva, A. González, R. Pérez, Three new instance selection methods based on local sets: A comparative study with several approaches from a bi-objective perspective, Pattern Recognition 48 (2015) 1523–1537. URL: https://www.sciencedirect.com/science/article/pii/S0031320314003859. doi:https://doi.org/10.1016/j.patcog.2014.10.001.

[10] Y. Caises, A. González, E. Leyva, R. Pérez, Combining instance selection methods based on data characterization: An approach to increase their effectiveness, Information Sciences 181 (2011) 4780–4798. URL: https://www.sciencedirect.com/science/article/pii/S0020025511002878. doi:https://doi.org/10.1016/j.ins.2011.06.013, special Issue on Interpretable Fuzzy Systems.

[11] B. Frenay, M. Verleysen, Classification in the presence of label noise: A survey, IEEE Transactions on Neural Networks and Learning Systems 25 (2014) 845–869. doi:10.1109/TNNLS.2013.2292894.

[12] X. Zhang, J. Zhao, Y. LeCun, Character-level convolutional networks for text classification, in: C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, R. Garnett (Eds.), Advances in Neural Information Processing Systems, volume 28, Curran Associates, Inc., 2015. URL: https://proceedings.neurips.cc/paper_files/paper/2015/file/250cf8b51c773f3f8dc8b4be867a9a02-Paper.pdf.

[13] C. Hutto, E. Gilbert, Vader: A parsimonious rule-based model for sentiment analysis of social media text, Proceedings of the International AAAI Conference on Web and Social Media 8 (2014) 216–225. URL: https://ojs.aaai.org/index.php/ICWSM/article/view/14550. doi:10.1609/icwsm.v8i1.14550.

[14] S. Garcia, J. Derrac, J. Cano, F. Herrera, Prototype selection for nearest neighbor classification: Taxonomy and empirical study, IEEE Transactions on Pattern Analysis and Machine Intelligence 34 (2012) 417–435. doi:10.1109/TPAMI.2011.142.

[15] F. W. Glover, G. A. Kochenberger, A tutorial on formulating QUBO models, CoRR abs/1811.11538 (2018). URL: http://arxiv.org/abs/1811.11538. arXiv:1811.11538.

[16] O. Arbelaitz, I. Gurrutxaga, J. Muguerza, J. M. Pérez, I. Perona, An extensive comparative study of cluster validity indices, Pattern Recognition 46 (2013) 243–256. URL: https://www.sciencedirect.com/science/article/pii/S003132031200338X. doi:https://doi.org/10.1016/j.patcog.2012.07.021.

[17] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, CoRR abs/1908.10084 (2019). URL: http://arxiv.org/abs/1908.10084. arXiv:1908.10084.

[18] C. C. Aggarwal, C. Zhai, A Survey of Text Clustering Algorithms, Springer US, Boston, MA, 2012, pp. 77–128. URL: https://doi.org/10.1007/978-1-4614-3223-4_4. doi:10.1007/978-1-4614-3223-4_4.

[19] A. Tandon, A. Albeshri, V. Thayananthan, W. Alhalabi, F. Radicchi, S. Fortunato, Community detection in networks using graph embeddings, Physical Review E 103 (2021). URL: http://dx.doi.org/10.1103/PhysRevE.103.022316. doi:10.1103/physreve.103.022316.

[20] B. Probierz, A. Hrabia, J. Kozak, A new method for graph-based representation of text in natural language processing, Electronics 12 (2023) 2846. doi:10.3390/electronics12132846.

[21] Y. Kim, B. Shin, In defense of core-set: A density-aware core-set selection for active learning, 2022. URL: https://arxiv.org/abs/2206.04838. arXiv:2206.04838.

[22] J. Barnes, Sentiment and emotion classification in low-resource settings, in: J. Barnes, O. De Clercq, R. Klinger (Eds.), Proceedings of the 13th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis, Association for Computational Linguistics, Toronto, Canada, 2023, pp. 290–304. URL: https://aclanthology.org/2023.wassa-1.26/. doi:10.18653/v1/2023.wassa-1.26.

[23] K. Wei, R. Iyer, J. Bilmes, Submodularity in data subset selection and active learning, in: F. Bach, D. Blei (Eds.), Proceedings of the 32nd International Conference on Machine Learning, volume 37 of *Proceedings of Machine Learning Research*, PMLR, Lille, France, 2015, pp. 1954–1963. URL: https://proceedings.mlr.press/v37/wei15.html.

[24] A. Siddhant, Z. C. Lipton, Deep bayesian active learning for natural language processing: Results

of a large-scale empirical study, 2018. URL: https://arxiv.org/abs/1808.05697. arXiv:1808.05697.

[25] A. Lucas, Ising formulations of many np problems, Frontiers in Physics 2 (2014). URL: http://dx.doi.org/10.3389/fphy.2014.00005. doi:10.3389/fphy.2014.00005.

[26] C. Bauckhage, N. Piatkowski, R. Sifa, D. Hecker, S. Wrobel, A qubo formulation of the k-medoids problem, Fraunhofer Publication Server (2019). URL: https://publica.fraunhofer.de/handle/publica/405469, accessed: 2025-05-29.

[27] W. Alvarez-Girón, J. Téllez-Torres, J. Tovar-Cortes, H. Gómez-Adorno, Notebook for the quantumclef lab at clef 2024, in: Conference and Labs of the Evaluation Forum (CLEF 2024), CEUR Workshop Proceedings, CEUR-WS.org, 2024. URL: https://ceur-ws.org/, use permitted under CC BY 4.0.

[28] M. Shi, K. Hua, J. Ren, Y. Cao, Global optimization of k-center clustering, in: K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, S. Sabato (Eds.), Proceedings of the 39th International Conference on Machine Learning, volume 162 of *Proceedings of Machine Learning Research*, PMLR, 2022, pp. 19956–19966. URL: https://proceedings.mlr.press/v162/shi22b.html.

[29] S. Hicks, R. Liu, Y. Ni, E. Purdom, D. Risso, mbkmeans: fast clustering for single cell data using mini-batch k -means, 2020. doi:10.1101/2020.05.27.119438.

[30] E. Schubert, P. J. Rousseeuw, Fast and eager k-medoids clustering: O(k) runtime improvement of the pam, clara, and CLARANS algorithms, CoRR abs/2008.05171 (2020). URL: https://arxiv.org/abs/2008.05171. arXiv:2008.05171.