Algorithm 5: WCKMeans

Input: Corpus $\mathcal{T}$, clusters K, MLS $\mathcal{M}$, CLS $\mathcal{C}$, max iterations T
Output: Labels Γ

Normalize $\mathcal{T}$ to get embeddings $\mathbf{X}$;
Construct constraint matrix $\mathbf{R}$ with $\mathcal{M}$, $\mathcal{C}$, $|\mathcal{T}|$, and $\mathbf{X}$;
Initialize centers $\mathcal{U} \leftarrow \{\mathbf{u}_1, \ldots, \mathbf{u}_K\}$;
while *not converged* & $t < T$ **do**
  foreach $x_i \in \mathbf{X}$ **do**
    $$\Gamma_i \leftarrow \arg\min_{1 \le k \le K} \|x_i - \mathbf{u}_k\|^2 + \sum_{\substack{(i,j)\in\mathcal{M} \\ \Gamma_j \neq k}} R_{i,j} + \sum_{\substack{(i,j)\in\mathcal{C} \\ \Gamma_j = k}} |R_{i,j}|$$
  Update centers $\mathbf{u}_k \leftarrow \texttt{Mean}(\{\, x_i \mid \Gamma_i = k \})$ for each k;
return Γ

A Proof of Lemma 5.2

Proof. Let $(t_i, t_j) = \arg\max_{(t_a,t_b)\in\mathcal{S},\, a<b} a$. In the extreme case, $j = i+1$ and only edges $(t_a, t_{a+1}), (t_a, t_{a+2}), \ldots, (t_a, t_j)$ are added into $\mathcal{S}$ for each $a \in [1, i-1]$ as they are surely larger than (t_i, t_j). Put in another way, for each t_a with $a \in [1, i]$, we can select t_b with $a+1 \le b \le j$, i.e., which corresponds to $j - a = i + 1 - a$ text instances. Thus, the total number of edges satisfy:

$$1 + 2 + \ldots + i = \frac{(i+1)i}{2} = N.$$

Then, we can derive

$$i^2 + i - 2N = 0 \Rightarrow i = \frac{\sqrt{8N+1} - 1}{2},$$

which finishes the proof. □

B Details of Datasets and Baselines

Datasets. The *BBC News* [27] dataset contains 2,225 headlines collected from the BBC News website and categorized into 5 high-level topics. It serves as a benchmark for multi-class classification tasks in the news field. The *Tweet* [71] dataset consists of 2,472 tweets annotated for relevance to 89 queries from the TREC Microblog track, and is widely used for short-text retrieval and ranking. The *Bank77* [74] dataset includes 3,080 customer service utterances mapped to 77 intent categories, enabling fine-grained intent classification in the financial category. The *Reddit* [36] dataset, with 3,217 posts from online communities, is used primarily for unsupervised topic discovery. The *CLINC* [78] dataset comprises 4,500 user requests labeled across 10 distinct domains for domain-level text clustering. Finally, the *Massive* [20] dataset contains 11,514 utterances spanning 18 scenario-based categories, and is often used in multilingual and multi-scenario classification settings.

Baselines. We include two sets of baselines in the experiments: *embedding-based clustering methods* and *LLM-assisted clustering methods*. In embedding-based clustering methods, we include seven representative baselines that follow a two-stage paradigm: generating fixed-dimensional text embeddings followed by conventional clustering algorithms, such as K-Means++ or *spectral clustering* [63]. The selected embedding models encompass both traditional and neural models.

- **TF-IDF**: A classic sparse representation based on term frequency-inverse document frequency, widely used in earlier text clustering studies.
- **E5** [65]: A recent retrieval-oriented embedding model pretrained with a multi-task objective, shown to perform well in clustering tasks.
- **DistilBERT** and **Sentence-BERT** [46]: Popular lightweight transformers optimized for sentence-level embeddings.
- **Instructor-Large** [56]: A powerful instruction-tuned model designed to align embeddings with various downstream tasks; we use it as the default embedder in most baselines due to its strong performance.
- **OpenAI-GPT** [6] and **LLaMA-2 (7B)** [60]: Large language models (LLMs) whose embeddings are obtained by averaging final-layer token representations. As LLMs are often optimized for generation, we treat them as black-box embedding extractors here.

These embedding models are coupled with K-Means++ or *spectral clustering*, depending on the context.

In LLM-assisted clustering methods, we additionally consider three recent strong methods that leverage LLMs beyond embedding extraction.

- **SCCL** [73]: A self-supervised contrastive clustering method that improves cluster discrimination by minimizing intra-cluster distance and maximizing inter-cluster margins in embedding space.
- **ClusterLLM** [77]: Utilizes LLMs to provide clustering feedback and refines smaller embedding models through LLM guidance. We also include a variant that excludes fine-tuning to assess the impact of feedback alone.
- **PO-PCKMeans** [62]: A semi-supervised approach that constructs pairwise constraints from an LLM based on a small set of labeled examples and integrates them into a constrained clustering framework.

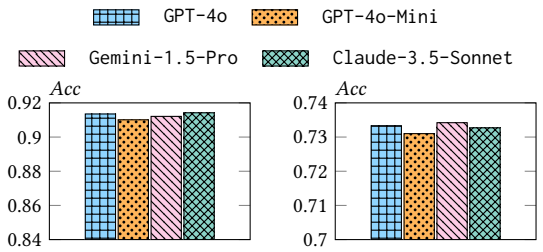
C Additional Algorithmic Details

C.1 WCKMeans

Algorithm 5 integrates well-designed weighted constraints into standard K-Means by creating a constraint matrix $\mathbf{R}$, where $R_{i,j} > 0$ for must-links and $R_{i,j} < 0$ for cannot-links. In each iteration, each point x_i is assigned to the cluster k that minimizes its squared distance to centroid $\mathbf{u}_k$ plus penalties: add $R_{i,j}$ if a must-link partner j is in a different cluster, and add $|R_{i,j}|$ if a cannot-link partner j is in the same cluster. After assignments, centroids are updated as the mean of their assigned points. This repeats until convergence or T iterations.

C.2 WCSC

Algorithm 6 starts by building a graph that captures how similar each pair of points is in the normalized embedding space. Then selecting top-K eigenvectors finds a low-dimensional embedding that balances the natural data structure with must-links and cannot-links. The α [66] serves as a balance factor between the original information captured by the graph and the constraints. Finally, K-Means is applied to produce the cluster labels.



(a) *BBC News* **(b) *Tweet***

Figure 6: The performance of Cequel with various LLMs

Algorithm 6: WCSC

Input: Corpus $\mathcal{T}$, clusters K, MLS $\mathcal{M}$, CLS $\mathcal{C}$
Output: Labels Γ

Normalize $\mathcal{T}$ to get embeddings $\mathbf{X}$;
Build affinity matrix $\mathbf{A}$ with $\mathbf{X}$;
Construct constraint matrix $\mathbf{R}$ with $\mathcal{M}$, $\mathcal{C}$, $|\mathcal{T}|$, and $\mathbf{X}$;
$\mathbf{L} \leftarrow \mathbf{D}^{-1/2}(\mathbf{D}-\mathbf{A})\mathbf{D}^{-1/2}$, $\mathbf{R}_\alpha \leftarrow \mathbf{D}^{-1/2}\mathbf{R}\mathbf{D}^{-1/2} - \alpha\mathbf{I}$;
Compute top-K generalized eigenvectors $\mathbf{V}$ of $\mathbf{L}$ and $\mathbf{R}_\alpha$;
$\Gamma \leftarrow$ K-Means$(\mathbf{D}^{-1/2}\mathbf{V})$;
return Γ

Algorithm 7: Edge Query

Input: Corpus $\mathcal{T}$, the number of edges N, Oracle $\mathbf{O}$
Output: MLS $\mathcal{M}$, CLS $\mathcal{C}$

$\mathcal{S} \leftarrow$ GreedyEdgeSelection$(\mathcal{T}, N)$;
foreach $(t_i, t_j) \in \mathcal{S}$ **do**
  switch $\mathbf{O}(t_i, t_j)$ **do**
    case *ML* **do**
      $\mathcal{M} \leftarrow \mathcal{M} \cup \{(t_i, t_j)\}$
    case *CL* **do**
      $\mathcal{C} \leftarrow \mathcal{C} \cup \{(t_i, t_j)\}$
return $\mathcal{M}$, $\mathcal{C}$

Algorithm 8: Triangle Query

Input: Corpus $\mathcal{T}$, the number of triangles $N_\triangle$, Oracle $\mathbf{O}$
Output: MLS $\mathcal{M}$, CLS $\mathcal{C}$

$\mathcal{S}_\triangle \leftarrow$ GreedyTriangleSelection$(\mathcal{T}, N_\triangle)$;
foreach $(t_i, t_j, t_k) \in \mathcal{S}_\triangle$ **do**
  switch $\mathbf{O}(t_i, t_j, t_k)$ **do**
    case *all-same* **do**
      $\forall\,(p,q) \in \{(i,j),(i,k),(j,k)\}: \mathcal{M} \leftarrow \mathcal{M} \cup \{(t_p, t_q)\}$
    case *ij-same* **do**
      $\mathcal{M} \leftarrow \mathcal{M} \cup \{(t_i, t_j)\}$, $\mathcal{C} \leftarrow \mathcal{C} \cup \{(t_i, t_k), (t_j, t_k)\}$
    case *ik-same* **do**
      $\mathcal{M} \leftarrow \mathcal{M} \cup \{(t_i, t_k)\}$, $\mathcal{C} \leftarrow \mathcal{C} \cup \{(t_i, t_j), (t_j, t_k)\}$
    case *jk-same* **do**
      $\mathcal{M} \leftarrow \mathcal{M} \cup \{(t_j, t_k)\}$, $\mathcal{C} \leftarrow \mathcal{C} \cup \{(t_i, t_j), (t_i, t_k)\}$
    case *all-diff* **do**
      $\forall\,(p,q) \in \{(i,j),(i,k),(j,k)\}: \mathcal{C} \leftarrow \mathcal{C} \cup \{(t_p, t_q)\}$
return $\mathcal{M}$, $\mathcal{C}$

C.3 Edge Query

Algorithm 7 uses Algorithm 1 (GreedyEdgeSelection) to select the top N instance pairs most informative for querying. Each chosen pair (t_i, t_j) is queried to the oracle $\mathbf{O}$, and the corresponding constraints are added based on the responses.

C.4 Triangle Query

Algorithm 8 utilizes Algorithm 4 (GreedyTriangleSelection) to select $N_\triangle$ triangles (t_i, t_j, t_k). For each triangle, the oracle $\mathbf{O}$ returns five outcomes—"all-same", "ij-same", "ik-same", "jk-same", or "all-diff", and the corresponding must-link or cannot-link constraints are added to $\mathcal{M}$ and $\mathcal{C}$.

D Additional Experiments

Various Text Encoders. Table 7 reports the clustering performance of Cequel and strong baselines using two representative encoders: Sentence-BERT and Instructor-Large. Results demonstrate the consistent superiority of Cequel across datasets and embeddings. For instance, with Instructor-Large on *Tweet*, Cequel reaches 73.10% ACC and 89.08% NMI, exceeding ClusterLLM by 7.29% and 0.47%. With Sentence-BERT on *BBC News*, Cequel achieves 84.45% ACC and 61.65% NMI, outperforming PO-PCKMeans by 1.75% and 4.22%, respectively. These gains are consistent across diverse corpora, from structured news to informal tweets and task-oriented texts like *Bank77* and *Reddit*. The results highlight the adaptability of Cequel and its ability to take advantage of powerful pre-trained encoders and informative constraints without task-specific tuning.

Queried LLMs. Fig. 6 illustrates the clustering results of Cequel when querying four popular LLMs including GPT-4o, GPT-4o-Mini, Gemini-1.5-Pro, and Claude-3.5-Sonnet. Key observations include: (1) Minimal differences in the adoption of various LLMs. (2) LLMs influence answer accuracy, particularly regarding constraints, without affecting query triangle selection; for example, GPT-4o enhances constraint accuracy compared to GPT-4o-Mini, resulting in increased overall accuracy. (3) The advantages of more expensive LLMs do not warrant their costs, underscoring the necessity for more cost-effective models capable of managing complex tasks like triangle queries.

E Additional Prompts

We provide additional example prompts in Table 8 for edge and triangle queries across multiple datasets, including *BBC News*, *Tweet*, *Bank77*, *Reddit*, and *Massive*. Edge queries ask whether two documents belong to the same category (Yes/No), while triangle queries assess the categorical relationship among three documents via a five-way multiple-choice format. These prompts are designed to support constraint construction using LLMs for clustering tasks.

Table 7: Clustering performance with various text encoders

Encoder	Method	BBC News		Tweet		Bank77		Reddit	
		ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI
Sentence-BERT	SCCL	82.00	58.33	53.60	72.80	27.80	53.21	36.60	45.71
	ClusterLLM	80.08	54.13	57.47	83.74	58.61	77.06	36.09	44.74
	PO-PCKMeans	82.70	57.43	61.39	86.12	57.73	78.09	38.48	48.56
	Cequel	**84.45**	**61.65**	**78.48**	**89.30**	**67.05**	**80.46**	**40.94**	**49.32**
Instructor-Large	SCCL	83.60	62.00	36.50	68.50	36.00	61.30	28.70	36.00
	ClusterLLM	88.40	69.73	65.81	88.61	68.27	82.45	54.80	62.47
	PO-PCKMeans	89.53	72.16	65.58	88.20	67.79	81.92	55.10	62.51
	Cequel	**91.01**	**75.22**	**73.10**	**89.08**	**69.03**	**83.20**	**57.78**	**64.25**

Table 8: Example prompts for edge and triangle queries on different datasets.

Dataset	Type	Prompt
BBC News	Edge Query	Cluster BBC News docs by whether they belong to the same news category. For each pair, respond with Yes or No without explanation. - News #1: Ad sales boost Time Warner profit - News #2: Air passengers win new EU rights Given this context, do News #1 and News #2 likely correspond to the same news category?
	Triangle Query	Cluster BBC News docs by whether they belong to the same news category. For each triangle, respond with a, b, c, d, or e without explanation. - News #1: REM announce new Glasgow concert - News #2: Moreno debut makes Oscar mark - News #3: Last Star Wars 'not for children' Given this context, do News #1, News #2, and News #3 likely correspond to the same news category? **(a)** All are same category. **(b)** Only #1 and #2 are same category. **(c)** Only #1 and #3 are same category. **(d)** Only #2 and #3 are same category. **(e)** None.
Tweet	Edge Query	Cluster Tweet docs by whether they belong to the same tweet category. For each pair, respond with Yes or No without explanation. - Tweet #1: super bowl commercial - Tweet #2: kung pao chicken recipe Given this context, do Tweet #1 and Tweet #2 likely correspond to the same tweet category?
	Triangle Query	Cluster Tweet docs by whether they belong to the same tweet category. For each triangle, respond with a, b, c, d, or e without explanation. - Tweet #1: weight loss diet plan acai juice equal nutritional claim acai - Tweet #2: watch christina aguilera screw national anthem super bowl post - Tweet #3: yell cast jr hartley ad digital era author search fly fishing book dj hunt book read Given this context, do Tweet #1, Tweet #2, and Tweet #3 likely correspond to the same tweet category? **(a)** All are same category. **(b)** Only #1 and #2 are same category. **(c)** Only #1 and #3 are same category. **(d)** Only #2 and #3 are same category. **(e)** None.
Bank77	Edge Query	Cluster Bank77 docs by whether they belong to the same intent category. For each pair, respond with Yes or No without explanation. - Intent #1: I think something went wrong with my card delivery as I haven't received it yet. - Intent #2: How do I link to my credit card with you? Given this context, do Intent #1 and Intent #2 likely correspond to the same intent category?
	Triangle Query	Cluster Bank77 docs by whether they belong to the same intent category. For each triangle, respond with a, b, c, d, or e without explanation. - Intent #1: What is the current exchange rate for me? - Intent #2: I believe my card payment exchange rate is incorrect. - Intent #3: I think my child used my card while I wasn't home. Given this context, do Intent #1, Intent #2, and Intent #3 likely correspond to the same intent category? **(a)** All are same category. **(b)** Only #1 and #2 are same category. **(c)** Only #1 and #3 are same category. **(d)** Only #2 and #3 are same category. **(e)** None.
Reddit	Edge Query	Cluster Reddit docs by whether they belong to the same topic category. For each pair, respond with Yes or No without explanation. - Topic #1: CTA to make all rail stations accessible within 20 years - Topic #2: A Krazy Mug Oyee Balle Balle!!! Kettle with set of two glasses Given this context, do Topic #1 and Topic #2 likely correspond to the same topic category?
	Triangle Query	Cluster Reddit docs by whether they belong to the same topic category. For each triangle, respond with a, b, c, d, or e without explanation. - Topic #1: Batman and Batwoman Fan art - Topic #2: Books on American Union and Labour Movements? - Topic #3: When Taliban offer you gold: Afghan youth in crisis? Given this context, do Topic #1, Topic #2, and Topic #3 likely correspond to the same topic category? **(a)** All are same category. **(b)** Only #1 and #2 are same category. **(c)** Only #1 and #3 are same category. **(d)** Only #2 and #3 are same category. **(e)** None.
Massive	Edge Query	Cluster Massive docs by whether they belong to the same scenario category. For each pair, respond with Yes or No without explanation. - Scenario #1: here is something from today - Scenario #2: remind me of the meeting on tuesday Given this context, do Scenario #1 and Scenario #2 likely correspond to the same scenario category?
	Triangle Query	Cluster Massive docs by whether they belong to the same scenario category. For each triangle, respond with a, b, c, d, or e without explanation. - Scenario #1: alexa play my country playlist - Scenario #2: write email to my company mate to submit the task tomorrow - Scenario #3: remind me about my business meeting at three and forty five p. m. Given this context, do Scenario #1, Scenario #2, and Scenario #3 likely correspond to the same scenario category? **(a)** All are same category. **(b)** Only #1 and #2 are same category. **(c)** Only #1 and #3 are same category. **(d)** Only #2 and #3 are same category. **(e)** None.