Adv. IR Final Project Report

Hybrid Semantic-Lexical Clustering Framework for Improved Document Retrieval

Ben Ben Zvi 325526671 Noam Sasson 214439929

ABSTRACT

Most existing clustering-based retrieval methods rely primarily on lexical features such as TF-IDF, which while effective at capturing term frequency patterns, fail to account for deeper semantic relationships between documents. To address this limitation, we propose a hybrid clustering framework that enhances retrieval by integrating both lexical and semantic information, combining TF-IDF representations with E5 embeddings. Our approach involves KNN clustering and a Cluster-based retrieval process inspired by Kurland & Lee (2004). preliminary findings indicate that incorporating our encoding method improves retrieval effectiveness, particularly in the context of passages, with the Hybrid encoding outperforming baseline methods in both MAP and Recall. However, its performance is more moderate when applied to full documents, suggesting that the Hybrid approach excels when both lexical and semantic features are equally strong.

I. INTRODUCTION

A fundamental challenge in information retrieval is determining the relevance of a document to a given query. In the standard ad hoc retrieval setting, no explicit relevance judgments are provided, making it necessary to explore alternative sources of information or ways to improve retrieval effectiveness. One such approach is **Clustering**, which groups similar documents together to enhance retrieval by leveraging structural relationships within the corpus. This allows for a more flexible relevance assessment, as a document without a query term may still be considered relevant if it belongs to a cluster where most documents contain the term.

Most clustering-based retrieval methods rely on **TF-IDF** representations, which focus on term frequency patterns but fail to capture deeper semantic relationships. As a result, semantically similar documents may remain unconnected due to lexical differences, leading to suboptimal cluster formations. To address this issue, we propose a **Hybrid Clustering Framework** that integrates both lexical and semantic features, balancing corpus-specific term distributions with broader contextual meaning to improve clustering and retrieval effectiveness. Specifically, our approach combines traditional TF-IDF embeddings with E5 embeddings, enabling a richer representation of document content.

We begin by employing the BM25 algorithm to select a relevant subset of documents from the corpus, which is then used as input for a KNN clustering algorithm, utilizing a running window similarity measure to group the documents into clusters. This is followed by a cluster-based selection strategy, inspired by previous work on cluster-based retrieval by Kurland & Lee [2]. Specifically, we adopt **Set-Select** and **Bag-Select** methods for document selection, weighing document scores based on their relevance to the top clusters using KL Divergence.

One of the challenges in evaluating clustering-based retrieval methods lies in the complex relationship between clustering quality and retrieval effectiveness. While traditional clustering metrics such as WSS and Purity provide insight into the internal cohesion of clusters, they do not always correlate with the performance of retrieval systems. In this study, we investigate how the integration of semantic features into the clustering process can enhance **retrieval relevance**. The merging of different encodings inherently involves a trade-off, where each may need to "give" a bit to establish a common ground. While this could potentially result in a slight reduction in traditional clustering quality, it is expected to enhance retrieval performance. By examining this balance, we aim to highlight the potential benefits of hybrid clustering approaches.

II. RELATED WORK

Clustering-based retrieval has been extensively studied in the information retrieval literature, with early research demonstrating its potential to enhance retrieval effectiveness by leveraging document structure. One of the foundational studies in this area is the work of Liu and Croft [1], who proposed a cluster-based retrieval approach using language models. Their method introduced two key techniques: one for ranking clusters in retrieval tasks and another for smoothing document language models with cluster information.

Kurland and Lee [2] further explored offline clustering for document retrieval, introducing the Bag-Select and Set-Select methods to refine cluster-based selection. These methods have since been applied in various retrieval studies and are also utilized in our work. More recently, Djenouri et al. [3] proposed a framework that integrates clustering with frequent itemset mining to enhance retrieval performance.

Most of these approaches rely heavily on lexical similarity measures such as TF-IDF, which, while effective in capturing term frequency patterns, may overlook deeper semantic relationships between documents.

III. METHODOLOGY

Our approach proposes a **Hybrid Semantic-Lexical Clustering Framework** designed to enhance document retrieval by combining the semantic richness of LLM embeddings with the statistical grounding of unsupervised models.

A. Data Sources and Retrieval Pipeline

Our experiments are conducted using the **MS MARCO** dataset, which comprises a **Document Database** containing a corpus of full documents and a **Passage Database** consisting of shorter, pre-segmented passages. Queries are accompanied by relevance judgments, where each document or passage is assigned a score that quantifies its relevance to the corresponding query. The experimental pipeline incorporates several configurable parameters, including the number of nearest neighbors K in the KNN retrieval framework, the Cluster-based Retrieval strategy, and the choice of embedding approach. Additionally, a running window mechanism is employed for full-document retrieval, where documents are segmented into fixed-length windows, and local relevance scores are computed within each window.

B. Encoding Phase

The **Encoding** phase combines two distinct types of embeddings: **TF-IDF embeddings** and **E5 embeddings**. The TF-IDF method provides a sparse representation of documents based on the frequency of terms in a given corpus, highlighting lexical patterns and emphasizing key terms that may have higher significance. In contrast, E5 embeddings, generated through a pre-trained LLM, offer a dense representation that captures the semantic meaning of documents, including the contextual relationships between words and the overall themes of the document. By utilizing both embedding types, we leverage the strengths of each to form a hybrid representation that balances lexical detail and semantic richness.

Once both types of embeddings have been computed, the next step is normalization. Each embedding is normalized by taking the square root of the number of non-zero elements in the embedding vector. This normalization step eliminates dominance of either embedding type due to differing magnitudes, which is crucial for the subsequent phases. After normalization, the embeddings are concatenated to form a hybrid vector representation for each document, combining the lexical and semantic features and allowing for a more comprehensive representation of the document's content. The resulting hybrid embedding balances global semantic patterns with corpus-specific lexical nuances, effectively capturing both the statistical relevance of terms within the document and the deeper semantic meaning conveyed by the LLM, making it a versatile input for the clustering phase.

C. Clustering Phase

We begin by conducting an initial retrieval using the **BM25 algorithm** to select the top $k = 250$ documents or $k = 1000$ passages, with parameters set to $k_1 = 0.9$ and $b = 0.4$. The selected documents serve as the new Corpus and as the input for the clustering process. From this initially retrieved set we further select the top 50 documents to serve as cluster centers, defining 50 clusters, and assign documents to clusters using a **K-Nearest Neighbors** approach with varying values of k ($k = 2, 5, 10$). Notably, this approach allows for overlapping cluster memberships, meaning a single document may belong to multiple clusters, while some documents may be left without any cluster assignment if they do not fall within the neighborhood of any cluster center.

Since E5 embeddings primarily encode only the first $\sim$500 tokens of a document, we also experiment with segmenting documents into non-overlapping windows of size 500 to assess its impact. To compute document similarity we employ a distance metric based on a running window approach. Each segment is encoded using one of the selected embedding methods (TF-IDF, E5 embeddings, or our combined approach) and euclidean distance is computed between the corresponding window embeddings of the two documents, enabling a fine-grained assessment of local textual similarity. These localized similarity scores are aggregated (summed) to produce an overall similarity measure between document pairs, which is then used to construct a distance matrix that serves as the foundation for the KNN clustering algorithm.

D. Cluster-Based Retrieval

The **Retrieval** phase leverages the results of the clustering process to perform more accurate and relevant document retrieval. We adopt a method proposed by Kurland and Lee [2], adapting from offline to online clustering for document retrieval, using two approaches: **Bag-Select** and **Set-Select**.

1) Set-Select Approach: In the Set-Select method, we first extract a group C of the top clusters based on their relevance to the query. This relevance is determined using **KL Divergence**, comparing the query's induced LM to the cluster representation, which is the **Geometric Mean** of the induced LMs of the cluster's documents, smoothed with the corpus ($\lambda = 0.9$). Each cluster's score is computed as $\exp(-KLD(p_q||p_C))$, where p_q is the query's induced LM and p_C is the smoothed LM derived from the cluster's Geometric Mean representation.

Once the relevant clusters are identified, we apply a **binary filter** to select documents that belong to the top $m = 10$ clusters. Each selected document is then individually scored using the same approach— by computing the exponent of the negative KLD between the query LM and the document's LM. This ensures that both clusters and documents are ranked based on their proximity to the query, and that only documents from the most relevant clusters are returned.

2) Bag-Select Approach: The Bag-Select method builds upon the Set-Select approach by incorporating a more nuanced weighting scheme. In this method, we calculate the relevance of documents based on their membership in multiple top clusters. Similar to the Set-Select approach, the score for each document is calculated as the exponent of the negative KLD between the query's LM and the document's LM, however in the Bag-Select method, this score is then multiplied by the number of relevant (top) clusters the document belongs to. This weighting ensures that documents which are associated with multiple relevant clusters are prioritized, promoting a more diversified set of results. By considering documents spread across multiple clusters, this method provides a broader and more comprehensive perspective on document relevance.

E. Clustering Quality Assessment

Our hybrid representation builds on the idea of minimizing the within-cluster sum of squares (WSS) by combining two encodings, f_{tf-idf} and f_{e5}, as:

$$f_{hybrid}(d) = [\sqrt{\alpha} f_{tf-idf}(d), \sqrt{\beta} f_{e5}(d)],$$

where α and β are the means of the non-zero elements in f_{tf-idf} and f_{e5}, respectively. We show that for Clustering C:

$$WSS(f_{hybrid}, C) = \alpha WSS(f_{tf-idf}, C) + \beta WSS(f_{e5}, C),$$

and therefore we can minimize the WSS of the hybrid representation to get the desired effect. A formal proof of this relationship can be found in the appendix.

Experiments on the **Reuters-21578** dataset with different window sizes (30, 100, ∞) show that, while hybrid clustering's purity and entropy are generally worse than those of individual tf-idf and E5 clusterings, the WSS relationship holds as predicted, as shown in Table I

TABLE I
CLUSTERING PERFORMANCE ON REUTERS-21578 DATASET

Window and params	Encoding	WSS	Purity	Mean Entropy
$W.S = \infty$ $\alpha = \frac{1}{79.56}, \beta = \frac{1}{768.0}$	E5	46910.16	0.3092	3.5071
	TF-IDF	5259.51	**0.3108**	**3.3263**
	Hybrid	132.23	0.3107	3.4935
$W.S = 100$ $\alpha = \frac{1}{44.38}, \beta = \frac{1}{768.0}$	E5	249087.12	0.3123	2.9815
	TF-IDF	18240.46	0.3123	**2.3146**
	Hybrid	776.22	**0.3124**	2.8281
$W.S = 30$ $\alpha = \frac{1}{23.16}, \beta = \frac{1}{768.0}$	E5	1258159.15	**0.3127**	2.8804
	TF-IDF	54875.11	0.3123	**2.0391**
	Hybrid	4252.60	0.3115	2.7897

IV. RESULTS AND DISCUSSION

Tables II, III, and IV present the retrieval performance on the Passage and Doc databases across various methods, including BAG and SET retrieval strategies, different values of k for KNN, and encoding techniques (E5, TF-IDF, and Hybrid). The evaluation is based on three key metrics: Mean Average Precision (MAP), Precision at 5, and Recall. Additionally, for benchmarking purposes, the baseline performance of the initial BM25 retrieval method is included. For each configuration, the best result for each metric is highlighted in bold

TABLE II
RETRIEVAL PERFORMANCE FOR PASSAGE DB

Ret Method	k (KNN)	Encoding	MAP	P@5	Recall
BAG	2	E5	0.2263	0.5535	0.2863
		TF-IDF	0.224	**0.5581**	0.296
		Hybrid	**0.2357**	0.5209	**0.3024**
	5	E5	0.2524	0.5302	0.2726
		TF-IDF	0.2584	**0.586**	0.2793
		Hybrid	**0.2636**	0.5535	**0.281**
	10	E5	0.2847	0.5628	0.2756
		TF-IDF	**0.2997**	**0.6279**	**0.2881**
		Hybrid	0.292	0.5674	0.2824
SET	2	E5	0.2196	0.5163	0.2863
		TF-IDF	0.2216	**0.5256**	0.296
		Hybrid	**0.2294**	0.5209	**0.3024**
	5	E5	0.2451	0.5256	0.2726
		TF-IDF	0.2472	0.5395	0.2793
		Hybrid	**0.2553**	**0.5535**	**0.281**
	10	E5	0.2673	**0.5442**	0.2715
		TF-IDF	0.2711	0.5209	**0.2862**
		Hybrid	**0.2759**	**0.5442**	0.2807
Initial BM25			0.2457	0.6930	0.3393

As seen in Table II, for the **Passage DB** retrieval task, our Hybrid encoding mostly outperforms both E5 and TF-IDF in terms of MAP and Recall across the majority of configurations. This indicates that the Hybrid encoding, which integrates both term frequency and semantic features, is particularly well-suited for improving retrieval performance in this context. TF-IDF shows a slight edge in Precision over the hybrid approach, particularly at higher k values, however our method still remains competitive and is often close to TF-IDF in performance. E5, on the other hand, mostly demonstrates weaker performance, suggesting that term-based methods remain more effective in this scenario.

As for the Cluster-based retrieval methods, we can see that the BAG method outperforms the SET method and the BM25 baseline in MAP for most configurations, particularly as k increases. This aligns with the findings of Kurland et al. [2], who observed that the BAG selection method, when compared to the SET method, yields superior results under the same hyperparameters. Also, as k increases, retrieval effectiveness generally improves, although after a certain threshold the increase results in diminished improvements. That said, both Precision and Recall did not improve from the initial BM25 retrieval, suggesting that clustering does not necessarily enhance retrieval performance in these measures.

In the **Document DB** retrieval task as shown in Table III, TF-IDF consistently outperforms both our Hybrid encoding and E5 in all metrics and across most configurations. This suggests that for document-based retrieval tasks, where documents are larger and more diverse, term frequency-based methods are the most effective at capturing relevance. While our Hybrid encoding demonstrates competitive performance in certain cases, it fails to surpass TF-IDF. This is likely due to E5's generally poor performance, particularly for larger k values, which negatively impacts the Hybrid method that incorporates it in its weighting scheme. Furthermore, cluster-based retrieval methods prove ineffective for the Document DB task, as the BM25 baseline significantly outperforms all cluster-based approaches across MAP and P@5 metrics, with

TABLE III
RETRIEVAL PERFORMANCE FOR DOC DB

Ret. Method	k (KNN)	Encoding	MAP	P@5	Recall
BAG	2	E5	0.3015	0.4941	**0.5982**
		TF-IDF	**0.3161**	**0.5529**	0.5975
		Hybrid	0.3014	0.4824	**0.5982**
	5	E5	0.3299	0.5529	0.5772
		TF-IDF	**0.3626**	**0.6118**	**0.5956**
		Hybrid	0.322	0.5529	0.5772
	10	E5	0.3048	0.5882	0.4537
		TF-IDF	**0.3956**	**0.6588**	**0.5986**
		Hybrid	0.3081	0.6	0.4553
SET	2	E5	0.2974	0.4941	**0.5982**
		TF-IDF	**0.3137**	**0.5647**	0.5975
		Hybrid	0.2974	0.4941	**0.5982**
	5	E5	0.3323	0.5882	0.5772
		TF-IDF	**0.3565**	**0.6235**	**0.5956**
		Hybrid	0.3252	0.5882	0.5772
	10	E5	0.2875	0.5882	0.455
		TF-IDF	**0.3723**	**0.6118**	**0.5986**
		Hybrid	0.291	0.5882	0.4578
Initial BM25			0.4161	0.6941	0.5812

TABLE IV
RETRIEVAL PERFORMANCE FOR DOC DB WITH **WINDOW SIZE 500**

Ret. Method	k (KNN)	Encoding	MAP	P@5	Recall
BAG	2	E5	0.3235	**0.5647**	0.5883
		TF-IDF	0.3149	0.5529	**0.5947**
		Hybrid	**0.3311**	**0.5647**	0.5898
	5	E5	0.3528	0.5647	0.5844
		TF-IDF	0.3562	**0.6235**	**0.5982**
		Hybrid	**0.365**	0.6118	0.5878
	10	E5	0.3544	0.6118	0.5147
		TF-IDF	**0.3705**	**0.6824**	**0.5518**
		Hybrid	0.3591	0.6118	0.5311
SET	2	E5	0.3123	0.5529	0.5883
		TF-IDF	0.3075	**0.5647**	**0.5947**
		Hybrid	**0.3159**	**0.5647**	0.5898
	5	E5	0.3532	**0.6**	0.5844
		TF-IDF	0.3416	0.5647	**0.5982**
		Hybrid	**0.3562**	**0.6**	0.5878
	10	E5	0.332	0.6235	0.5132
		TF-IDF	**0.3436**	0.5765	**0.5518**
		Hybrid	0.3411	**0.6471**	0.5299
Initial BM25			0.4161	0.6941	0.5812

our Hybrid and E5 encodings only having a slight edge on recall for smaller k values. This suggests that while the clustering techniques may offer some benefits for recall, they are not well-suited for this retrieval scenario.

In contrast to the previous task, we observe that as k increases, the results do not consistently improve. While the MAP values for TF-IDF increase with k, the results for the other encodings exhibit varying trends. Notably, for Recall, there is a significant decline in performance for the E5 encoding as k increases. This suggests that increasing the number of clusters does not necessarily enhance retrieval effectiveness in the Document DB task. Instead, the optimal number of clusters may be more limited, with the inclusion of additional clusters potentially introducing noise or irrelevant information, thereby diminishing overall performance, particularly Recall.

while our Hybrid Encoding approach aims to balance semantic and term-based retrieval, its effectiveness is hindered by the relatively poor performance of the E5 encoding, which struggles to provide a sufficiently robust semantic signal, due to its limitation of encoding only the first $\sim$500 tokens of a document. This shortcoming highlights a potential loss of critical information, leading us to explore a sliding window approach with a size of 500 tokens.

As shown in Table IV, the introduction of a sliding window approach in the Document DB retrieval task resulted in significant improvements in the performance of our Hybrid encoding over the full document approach, particularly in terms of MAP. The Hybrid method outperformed TF-IDF, which exhibited a decline in performance with the sliding window approach applied, whereas E5 showed notable improvements, especially for smaller values of k. These gains suggest that segmenting documents into smaller, more manageable windows, while aggregating local similarities, enhances retrieval accuracy by emphasizing the most relevant content within each window.

Despite these advancements, TF-IDF consistently outperformed the Hybrid encoding in Precision and Recall across most configurations. Additionally, cluster-based retrieval methods continued to underperform the BM25 baseline in terms of MAP and Precision. However, the ability of cluster-based methods to improve Recall highlights their utility in retrieving relevant documents, even if they are not as effective in ranking them as highly as BM25. Once again, the BAG method proved superior to SET, and increasing k led to improvements in MAP and Precision but had a mixed impact on Recall.

V. CONCLUSION

Our study introduces a Hybrid clustering framework that effectively integrates lexical and semantic features to enhance retrieval performance. The results highlight the superior effectiveness of our Hybrid encoding in the Passage DB retrieval task, where it consistently surpasses both E5 and TF-IDF in terms of Mean Average Precision and Recall. In contrast in the Document DB task, while our Hybrid encoding demonstrates competitive performance when paired with a 500-token sliding window approach, it still falls short of the TF-IDF method and the BM25 baseline in overall retrieval effectiveness.

The advantages of our Hybrid encoding are most pronounced in scenarios where both E5 and TF-IDF yield comparable results, as it generally delivers the highest performance under these conditions. However, in cases where one encoding method significantly underperforms the other, Hybrid encoding tends to produce more moderate results, suggesting that its primary strength lies in contexts where both term-based and semantic features contribute equally to retrieval performance. This underscores the importance of leveraging complementary representations to optimize information retrieval tasks.

Furthermore, our experiments underscore the critical role of parameter selection, particularly in the context of KNN clustering. While increasing the number of neighbors enhances retrieval performance up to a certain threshold, further expansion beyond this point results in diminishing results, and in some cases a decline in effectiveness. This finding highlights the necessity of fine-tuning model parameters to achieve an optimal balance between retrieval accuracy and computational efficiency. Also, consistent with the findings of Kurland et al. [2], our results validate the superiority of the BAG selection method over the SET method when evaluated under identical hyperparameters, reinforcing the effectiveness of BAG-based retrieval strategies.

In summary, this study underscores the benefits of Hybrid encoding in retrieval tasks where lexical and semantic features play complementary roles. While it excels in the Passage DB retrieval task, its effectiveness in the Document DB task remains context-dependent, particularly in comparison to BM25. Additionally, our findings highlight the importance of selecting optimal clustering parameters to maximize retrieval performance. These insights contribute to the ongoing development of hybrid retrieval methods and offer a foundation for future research in refining information retrieval strategies.

LIMITATIONS

This study presents several limitations that should be acknowledged. In light of the success achieved with the 500-token sliding window approach in the document-based retrieval task, we initially aimed to also test smaller window sizes (100 and 250 tokens), hoping to gain further improvements. However, these configurations proved to be computationally demanding, which led to their exclusion from the final experiments. Nonetheless, these smaller window sizes may still offer valuable insights for future research.

Additionally, our initial approach involved single-relevance queries, where each query was associated with only one relevant document. This resulted in suboptimal performance, as clustering techniques may not be as effective in such scenarios, where the limited number of relevant documents reduces the value of grouping similar ones. Nonetheless, addressing this limitation is essential for improving the handling of such cases in retrieval tasks.

REFERENCES

[1] X. Liu and W. B. Croft, *Cluster-Based Retrieval Using Language Models*, In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2004)*, Sheffield, UK, 2004, pp. 186–193. Available: https://ciir.cs.umass.edu/pubfiles/ir-347.pdf

[2] O. Kurland and L. Lee, *Corpus Structure, Language Models, and Ad Hoc Information Retrieval*, In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2004)*, Sheffield, UK, 2004, pp. 194–201. Available: https://arxiv.org/pdf/cs/0405044

[3] Y. Djenouri, B. Djenouri, and M. Belhadi, *Fast and Effective Cluster-Based Information Retrieval Using Frequent Closed Itemsets*, In *Information Sciences*, vol. 453, pp. 154–167, 2018. Available: https://findresearcher.sdu.dk/ws/portalfiles/portal/142085072/Fast_and_Effective_Cluster_based_Information_Retrieval_using_Frequent_Closed_Itemsets.pdf

APPENDIX

Formal Problem Definition

Given a set of documents $D = \{d_1, d_2, ..., d_n\}$ and two mappings $f1 : D \to \mathbb{R}^m$ and $f2 : D \to \mathbb{R}^n$ we seek a new mapping $f3(f1, f2) : D \to \mathbb{R}^k$ that combines the information from the two mappings such that minimizing the intra-cluster distance in $f3$ also minimizes the intra-cluster distance in the original spaces $f1$ and $f2$. Formally, let $C = \{c_1, c_2, ..., c_k\}$ be any given clustering of the documents, the Within-Sum-of-Squares (WSS) of C in f is defined as :

$$WSS(f, C) = \sum_{i=1}^{k} \sum_{d \in c_i} ||f(d) - \mu_{f,i}||^2$$

where $\forall i \ \mu_{f,i} = \frac{1}{|c_i|} \sum_{d \in c_i} f(d)$ and are stricltly defined by the clustering C and the mapping f. We want f_3 to satisfy:

$$\forall C \ WSS(f_3, C) \propto \alpha WSS(f_1, C) + \beta WSS(f_2, C)$$

We will show how simple concatenation of the two embeddings can statisfy the above condition:

$$\alpha WSS(C, f1) + \beta WSS(C, f2)$$

$$= \alpha \sum_{c_i \in C} \sum_{d \in C_i} \|f_1(d) - \mu_i^{f1}\|_2^2 + \beta \sum_{c_i \in C} \sum_{d \in C_i} \|f_2(d) - \mu_i^{f2}\|_2^2$$

$$= \sum_{c_i \in C} \sum_{d \in C_i} \sum_{j=0}^{m} (\sqrt{\alpha} f_1(d)_j - \sqrt{\alpha} \mu_{i,j}^{f1})^2 + \sum_{c_i \in C} \sum_{d \in C_i} \sum_{j=0}^{n} (\sqrt{\beta} f_2(d)_j - \sqrt{\beta} \mu_{i,j}^{f2})^2$$

Now, defining the new feature vector $f^{new}(d)$ as:

$$f^{new}(d) = \begin{pmatrix} \sqrt{\alpha}\, f_1(d) \\ \sqrt{\beta}\, f_2(d) \end{pmatrix} \text{ we get } \mu^{f_{new}} = \begin{pmatrix} \sqrt{\alpha}\mu_i^1 \\ \sqrt{\beta}\mu_i^2 \end{pmatrix}$$

and by ubstituting these into the equation, we obtain:

$$= \sum_{c_i \in C} \sum_{d \in C_i} \sum_{j=0}^{m+n} \left(f^{new}(d)_j - \mu_{i,j}^{f^{new}} \right)^2$$

$$= \sum_{c_i \in C} \sum_{d \in C_i} \|f^{new}(d) - \mu_i^{f^{new}}\|_2^2 = \text{WSS}(C, f^{new})$$

Thus, we have shown that the WSS in the combined space is indeed proportional to the linear combination of the WSSs in the original spaces.