Where A Small Language Model Helps in Invoice Categorisation, Understood Through Embedding Geometry

Emma Ceccherini
University of Bristol
emma.ceccherini@bristol.ac.uk

Daniel Lawson
University of Bristol
dan.lawson@bristol.ac.uk

Anjulika Salhan
System Holdings Limited
anjulika.salhan@salhanaccountants.co.uk

Abstract

Categorising invoices into the correct General Ledger (GL) code underpins financial reporting and tax compliance. This is a skilled accounting judgement rather than a routine task: the correct category depends subtly on the nature of the purchasing business, the vendor and the invoice text. Whilst AI is increasingly being adopted across industries to automate tasks, including invoice categorisation, implementations built on in-house small language models (SLMs) can simultaneously reduce cost and improve data security, confidentiality, and interpretability. We investigate this approach by first analysing the pre-trained embedding geometry of a small sentence transformer (SBERT) and classic SLM (DeBERTa). The sentence-embedding space of this financial corpus is globally anisotropic but composed of locally isotropic clusters, extending prior token-level findings to sentence embeddings in a financial setting, and these clusters are strongly correlated with the vendor identity. SBERT fine-tuned on a single GPU reaches 0.96 accuracy on invoice classification, above both a zero-shot locally deployable LLM and a vendor identity baseline, increasing performance for smaller, challenging categories and new clients. For this important generalisation problem, SBERT reaches 0.9 F1 with roughly 100 client-specific invoices, showing that an in-house SLM implementation is promising. Combining these results with geometric analysis shows that pre-trained embedding geometry is associated with classification performance and reveals a counterintuitive finding that a structured input that would help a human reader does not improve the SLM performance.

1 Introduction

Every invoice generated by a business needs to be assigned the correct General Ledger (GL) code. This is a nuanced accounting task that requires specialised skills, and its success underpins financial reporting and tax compliance, under UK standards (FRS 105, FRS 102 Section 1A). A reduction of errors in invoice classification benefits small and medium enterprises (SMEs) through improved financial management, reduced compliance costs and exposure to risk through penalties. The UK tax gap is estimated around £59 billion, for which roughly half is attributed to errors or failure to take reasonable care (HM Revenue and Customs, 2026). Hence, the motivation for this research is clear, as improving automated mapping of invoices to GL codes would improve compliance, benefiting HM Revenue and Customs (HMRC) through increased compliance and SMEs through reduced exposure to fines and compliance costs.

AI is increasingly adopted to automate exactly these types of tasks. Large language models (LLMs) are the default choice, and some could perform well on this task. However, we find that off-the-shelf zero-shot (small) LLMs are not well suited to this specialised task (Section 4.1). On the other hand, in-house small language models (SLMs) are competitive in the financial domain (Bucher and Martini, 2024; Alessandro et al., 2025; Garcia-Mendez et al., 2020) and offer concrete advantages. They mitigate serious concerns around data security and confidentiality (Das et al., 2025; Wang et al., 2025) for a regulated domain like accounting, and significantly reduce costs (Belcak et al., 2025). Finally, SLMs improve interpretability (Anwar et al., 2024; Luo and Specia, 2024) through a large suite of tools available for encoder-only SLMs like BERT (Rogers et al., 2020; Hewitt and Manning, 2019; Clark et al., 2019), which are not available for closed external LLMs and which may eventually be required both for audit and regulatory purposes and for public trust.

Classification performance is a critical benchmark but an examination of the internal representation of data can offer a pathway towards intrinsic interpretability (Doshi-Velez and Kim, 2017; Rudin,

Proceedings of the 11th Workshop on Financial Technology and Natural Language Processing, pages 201–223
October 28, 2026 ©2026 Association for Computational Linguistics

2019). For the end user, interpretability promises a quantification of uncertainty (Liang et al., 2023), increased robustness (Geirhos et al., 2020), control over associations with protected categories, and human-intuitive representation of semantic meaning (Saxe et al., 2019). These properties can be derived in part from the intermediary of geometric properties, i.e. clustering, separability and manifold structure of the neural network (Gao et al., 2019; Cai et al., 2021; Nastase and Merlo, 2025) and can be linked to downstream behaviour (Rajaee and Pilehvar, 2021; Hämmerl et al., 2023) relating to features and labels. We therefore complement classification metrics with a geometric analysis, not as an end in itself but as a foundation for developing reliable, auditable and interpretable SLMs.

Contribution.

1. We show *where* a language model can make a valuable contribution over a vendor-only baseline, i.e. in smaller and challenging categories, which informs deployment implementations.
2. We show that a small *sentence transformer*, SBERT (∼22M parameters) (Reimers and Gurevych, 2019), fine-tuned on a single GPU, reaches 0.96 accuracy, substantiating SLMs as a potentially reliable and cost-effective route for deployment-level classification.
3. We extend prior token-level findings on globally anisotropic representation spaces with isotropic clusters to sentence embeddings; to the best of our knowledge, the first such result in a financial setting, and we show these clusters are highly associated with the vendors.
4. We find that embedding geometry helps explain SBERT's increased performance over larger *token transformer* DeBERTa (∼184M) (He et al., 2023): SBERT's pre-trained space has a higher effective dimensionality and lower average cosine similarity than DeBERTa's. And DeBERTa's CLS sentence embedding collapses as the input grows richer.
5. We find that on unseen clients the otherwise competitive vendor baseline collapses, and SBERT improves macro F1 over DeBERTa. This suggests that line-item text in learned representations with 'good geometry' carries semantic signal that transfers to unseen vendors. This is deployment-relevant as it directly measures client onboarding.
6. The *utility* of geometric analysis is twofold. Firstly, for this financial corpus, pre-trained embedding geometry is consistent with downstream performance and robustness. Secondly, it guides input representation: a more costly structured input useful for human reading does not improve the SLM performance due to increased similarity in the embedding space, directly relevant for practitioners.

Related work. Several works have used machine learning in the accounting domain (Berghaus et al., 2025; Chen et al., 2025; Noels et al., 2024; Tarawneh et al., 2019). Some specifically address the classification of invoices into GL codes, including frameworks based on static word embedding (Jørgensen and Igel, 2021; Kotepuchai and Limpiyakorn, 2024; Bardelli et al., 2020) or deep learning (Kieckbusch et al., 2020; Munoz et al., 2022; Chi et al., 2024; Liu et al., 2021). Only a few approaches exploited modern language models (Munoz et al., 2022; Chi et al., 2024).

The geometry of language models' representation spaces has been well studied (Ethayarajh, 2019; Cai et al., 2021; Ait-Saada and Nadif, 2023), with attempts to formalise representation (Allen-Zhu et al., 2019; Gao et al., 2019; Mickus et al., 2024). To the best of our knowledge, we are the first to use geometric tools to offer insight into classification performance on a financial corpus of invoices for sentence-based SLMs.

2 Methods

2.1 Dataset

We analyse the problem of automatic invoice classification into GL codes. Our confidential dataset consists of real invoices from seven clients of a UK accountancy firm that span various business domains. The firm provides its clients with software that uses OCR to extract relevant textual information from a PDF or an image of the invoice; we take that OCR output as our input. Each invoice is associated with the correct accounting category, manually labelled by professional accountants.

The full dataset comprises 2,423 invoices over 42 categories (Figure A1), with an average of 75 tokens per invoice. The data exhibits the severe class imbalance typical of this domain: only 14 categories contain more than 20 samples, and the largest category (*Purchases*) accounts for 705 invoices. We restrict our experiments to well-populated categories: the 12 categories with more than 40 samples (2,168 invoices) defined in Table A1. These contain enough labelled data to train and evaluate reliably, and together they account for

the majority of invoice volume.

From the OCR output, we construct three different textual inputs, with examples in Figure 1:

(A) Vendor name, date, and price only, excluding the line-item descriptions.

(B) Vendor name, date, line-item descriptions, and price, as returned by the OCR.

(C) As (B) with an explicit label (*Vendor Name:*, *Invoice Date:*, *Line Items:*, *Price:*).

Example — Label: Purchases

(A) TESCO, 12-05-2026, 4.45

(B) TESCO, 12-05-2026, 75892 Red Pepper 1 0.70 27836 Chocolate Cake PLT PION 390g 3.75, 4.45

(C) Vendor Name: TESCO, Invoice Date: 12-05-2026, Line Items: 75892 Red Pepper 1 0.70 27836 Chocolate Cake PLT PION 390g 3.75, Price: 4.45

Figure 1: Fictitious example of the three input variants.

2.2 Experimental set up

We use the sentence transformer **SBERT** (Reimers and Gurevych, 2019) (*all-MiniLM-L6-v2*), an encoder-only transformer designed and trained for sentence embedding with ≈22M parameters and 384 model dimension. Each invoice input is embedded from the last layer extracted with SBERT native pooling. We compare it to **DeBERTa** (*microsoft/deberta-v3-base*) with ≈184M parameters and 768 model dimensions. DeBERTa computes token embeddings which must be pooled to compute sentence embeddings. We consider the two most popular solutions: pooling using the 'CLaSsification' token embedding [CLS], vs pooling using the mean embedding of all tokens [mean], both in the last layer. The code can be found in this repository.

3 Geometric analysis

We analyse the pretrained embedding geometry along four diagnostics: effective dimensionality, global anisotropy, unsupervised clustering structure, and local isotropy.

3.1 Effective dimensionality

Inspired by Cai et al. (2021), we first look at the effective dimensionality of the embedding spaces, via centred PCA. For each combination of model and input type, we start with the embedding matrix $M \in \mathbb{R}^{n\times m}$, where n = 2,168 and m = 384 for SBERT and 768 for DeBERTa. Let $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_m$ be the eigenvalues of M's covariance matrix, and define the explained variance ratio $r_k = \sum_{i=1}^{k} \sigma_i \,/\, \sum_{i=1}^{m} \sigma_i$ and the ϵ-effective dimension to be $d(\epsilon) := \text{argmin}_k r_k \geq \epsilon$. For each model and input type, we report $d(0.8)$, the number of dimensions necessary to explain 80% of the variance in the embedding space, in Table 1.

Model	A	B	C	d_{model}
SBERT	42	61	57	384
DeBERTa mean	13	7	6	768
DeBERTa CLS	10	1	1	768

Table 1: Effective dimensionality (80% variance explained) by model and input type.

SBERT has higher effective dimensionality across all input types, as it's trained to produce semantically rich embeddings. SBERT embeddings for input types B and C have a higher $d(0.8)$ than A, reflecting the additional textual content, with little difference between B and C themselves.

Figure 2 shows the first 3 principal components of SBERT (B) embeddings. The embedding of *Accountancy fee* forms a tight cluster. Since every client in our dataset uses the same UK accountancy firm, these invoices are highly similar to each other. The rest of the data is distributed along an X-shaped manifold. The three biggest branches correspond to the three largest categories *Purchases* (green), *Motor Repairs and Servicing* (blue) and *Petrol and Oil* (orange). The fourth, smaller, branch is made up of *Light, Heat & Power* (red) *Subscriptions* (purple), *Telephone, Fax & Internet* (light green). Notably, SBERT, without any fine-tuning, picked up the semantic similarity that groups these last three categories as 'recurring bills'. Figure A2 shows the association with Vendor ID.

DeBERTa has a larger embedding dimension with low effective dimension, especially for CLS pooling. This reflects the well-known fact that CLS embeddings tend to have semantically 'meaningless' dominant directions (Liang et al., 2021); here, the first PC is highly correlated (> 90%) with the norm of the embeddings for all input types.

3.2 Anisotropy

Extensive literature has demonstrated that, counter-intuitively, well-performing SLMs and LLMs do not use all of the embedding space; this is called anisotropy (Cai et al., 2021; Gao et al., 2019; Nastase and Merlo, 2025). The degree to which the embeddings exist along shared directions is quan-

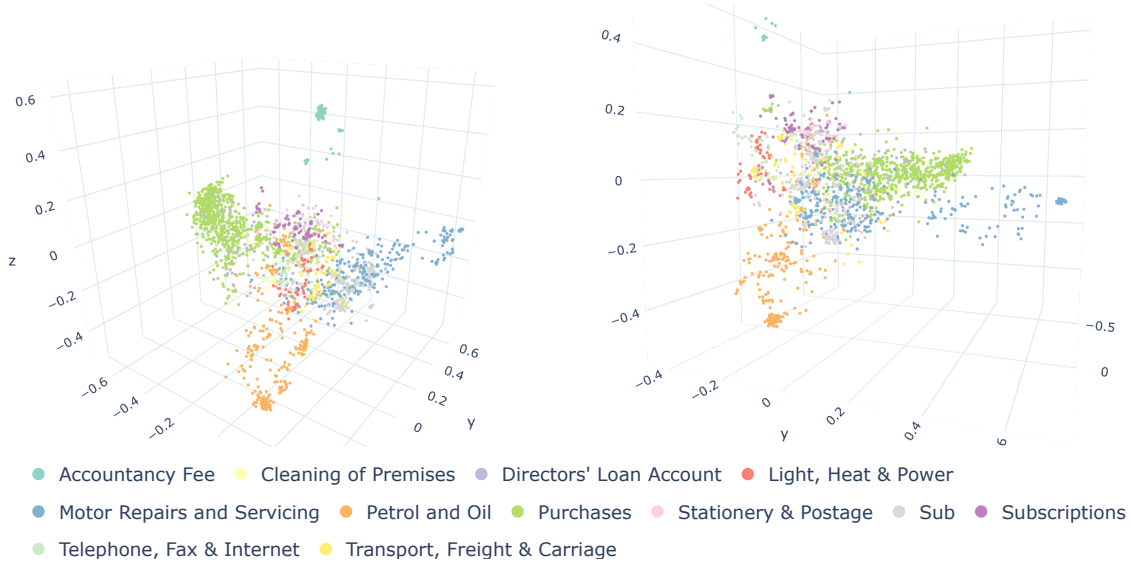

Figure 2: **Frozen transformer** visualisation of the first three principal components of the representation space of pre-trained SBERT with input B. Note the overlap between Purchases (green) and Directors Loan Account (light purple).

tified globally using the cosine similarity between any two pairs of embeddings in Table 2.

Model	A	B	C
SBERT	0.365	0.282	0.496
DeBERTa mean	0.973	0.933	0.949
DeBERTa cls	0.999	0.992	0.993

Table 2: **Frozen transformers** average cosine similarity between any embedding pairs by model and input type.

Consistent with the effective dimensionality results, both DeBERTa pooling strategies produce extremely high average cosine similarity, i.e. these sentence embeddings live in a narrow cone of the representation space. SBERT embeddings have substantially lower average cosine similarity, i.e., they are more homogeneously distributed, though still anisotropic (cosine similarity 0.3-0.5).

To analyse class-level structure, let C_k denote the set of indices in class k, and define the within-class and between-class cosine similarity:

$$\overline{s}_w = \frac{1}{K}\sum_{k=1}^{K}\frac{1}{|C_k|(|C_k|-1)}\sum_{\substack{i,j\in C_k,\\ i\neq j}}\cos(M_i, M_j),$$

$$\overline{s}_b = \frac{1}{K}\sum_{k=1}^{K}\frac{1}{|C_k|\,(n-|C_k|)}\sum_{\substack{i\in C_k,\\ j\notin C_k}}\cos(M_i, M_j).$$

Table 4 reports SBERT and DeBERTa-mean $\overline{s}_w$ and $\overline{s}_b$ with selected models, cluster definitions and

Model	A	B	C
SBERT	54	59	60
DeBERTa mean	52	2	2
DeBERTa CLS	52	2	2

Table 3: **Frozen transformers** number of K-means clusters $|G|$ selected by highest silhouette score, by model and input type.

origin; Table A2 shows every pairing, with example distributions in Figure A3.

As expected, the cosine similarity within classes is higher than that between classes (True labels, uncentered). Notably, Input C has both larger within and between cosine similarity than B (and A). This is not an artefact: the input texts in dataset C are genuinely more similar to one another, because the added field labels (*Vendor name*, *Price*, etc.) introduce seven identical words for each document. Unintuitively, Input C, the structured input, which would be helpful for a human reader, makes the inputs harder for a language model to distinguish.

3.3 Clustering

We now repeat the analysis of Section 3.2 with unsupervised clustering, instead of true labels. We perform K-means clustering of the embeddings and report the number of clusters $|G|$ in Table 3 which yields the 'most separable clusters' as measured by mean silhouette score for $G \in [1, 60]$.

The DeBERTa-CLS clustering on inputs B and C collapses to two tight (silhouette > 0.6) clusters, and cluster membership is highly correlated with

		True labels				K-means clustering			
		Uncentered		Centred		Uncentered		Centred	
Type	Model	Within	Between	Within	Between	Within	Between	Within	Between
B	SBERT	0.478	0.249	0.003	0.000	0.708	0.265	-0.015	0.000
B	DeBERTa mean	0.938	0.930	0.013	0.009	0.942	0.841	0.002	0.001
C	SBERT	0.641	0.485	0.003	0.000	0.823	0.476	-0.031	0.000
C	DeBERTa mean	0.956	0.948	0.019	0.011	0.953	0.864	0.000	0.001

Table 4: **Frozen transformer** average within-class/cluster and between-class/cluster cosine similarity, using the true labels and the K-means clustering partition.

embedding norm: the two clusters simply separate 'large' and 'small' embeddings. We discuss the collapse in more detail in Section 3.5.

SBERT embeddings are partitioned into a similar number of clusters for all input types, with tighter clusters for input A (silhouette score 0.5 vs. $\sim$0.3 for B and C). This clustering is strongly aligned with vendor identity: normalised mutual information (NMI) between vendor identity and cluster membership is > 0.8 across input types. NMI with true labels is lower (0.5-0.6), indicating some alignment between the clustering and the true labels. The clustering of input A is the most aligned with vendor identity as expected, since the vendor name is the only textual signal in input A. Inputs B and C clusterings are slightly more label-aligned, suggesting that adding the line-item text spreads the embeddings along directions aligned with the true accounting categories.

In the best-case scenario, SBERT with input B (SBERT-B), the clustering accounts for 61% of the uncertainty in the labels. Combined with the lowest average cosine similarity and the highest effective dimensionality (among SBERT variants), this shows that SBERT-B displays the best geometric properties, which we later show aligns with the classification results in Section 4.

Table 4 - clustering, uncentered - (full Table A3) reports the within- and between-cluster cosine similarity. Compared to the results with true labels, the within-cluster similarity is higher, while the between-cluster similarity is similar. This supports the hypothesis that the pre-trained embeddings identify the vendor name partition more sharply than the accounting category.

3.4 Local isotropy

Following Cai et al. (2021), we recompute the within- and between-class (or cluster) after subtracting the class (or cluster) mean. In this *local* sense, the sentence embeddings are almost perfectly isotropic (Table 4).

Cai et al. (2021) found local isotropy for *token* embedding. Our result extends theirs to *sentence* embeddings of invoices: the embedding space is a globally anisotropic manifold composed of locally isotropic clusters.

We further observe that centring by true labels and centring by K-means clusters recover isotropy to a similar degree, suggesting that both partitions are, in a different way but to an equal degree, the cause of the anisotropy. This can be explained by the near-perfect many-to-one mapping between vendors and accounting labels. Businesses repeatedly purchase similar goods or services from each vendor, so the vendor partition and the label partition expose nearly the same directions of variation in the embedding space.

3.5 CLS collapse

DeBERTa CLS embeddings show a striking pattern across all diagnostics. Moving from input type A to B/C, the best number of clusters drops from 52 to 2, the effective dimensionality goes from 10 to 1, and the cosine similarity is > 0.99. The CLS embeddings exhibit a consistent and severe collapse under richer textual inputs (B/C). The first principal component and the clustering partition are highly correlated with the embedding norm: the CLS embeddings recover the norm rather than any semantic property (Liang et al., 2021).

A likely explanation is that the CLS token, unlike content tokens, attends nearly uniformly to all the other tokens in the sentence. As input length grows from A to B and C, the CLS representation is the result of iterated weighted averages over many high-dimensional embedding vectors, which completely dilutes the signal. Indeed, the sentence embedding constructed with mean pooling exhibits a less extreme but similar pattern. In the latter, the

average is computed only once at the final layer rather than within every layer.

This collapse is consistent with the DeBERTa-CLS classifier degenerating to predicting only the larger classes (Section 4) while mean pooling retains some discriminative ability but still underperforms SBERT.

4 Classification

4.1 Frozen transformers and vendor baseline

For each model and input type, we freeze the encoder and train a linear classification head on top of the sentence embedding; for full implementation see A.1. In Section 3 we found that there was a strong association between the vendor identity and the labels, as businesses routinely repeat orders from the same suppliers. So we compare the language models to a parameter-free baseline that maps each vendor to its most frequent GL code in the training dataset and unseen vendors to the majority class. We also compare with a zero-shot open weight LLM *Qwen3-4B* baseline, for full implementation see A.1.

Table 5 reports the results. Without any encoder training, the vendor mapping baseline outperforms both SBERT and DeBERTa and the LLM baseline on any input type. Although the pre-trained embedding space of both models is highly correlated with vendor identity, the linear head is not enough to learn the map. The vendor-to-label map is many-to-one: with hundreds of vendors grouped in 12 classes. While a single memorised vendor is enough for the baseline to classify a small category correctly, a linear head requires more samples to learn a reliable linear decision boundary.

Focusing on the transformers' performance, the geometric properties of Section 3 are associated with classification performance, with SBERT on input B substantially surpassing DeBERTa mean and CLS, respectively, and input B being the best across models. Unintuitively, the structured input (C), which would help a human learner, doesn't improve the performance of a language model, due to the artificially increased within-corpus similarity described in Section 3.2. The full classification reports can be found in Tables A4-A8. The classification layer trained on DeBERTa CLS embeddings ignores ($\sim$0% accuracy) some of the smaller categories, consistent with the collapse highlighted in Section 3.5. SBERT on input B achieves F1 scores around 0.8 on all categories but one, regardless of the category size, showing that there is a genuine signal extractable from the embedding space, but not linearly accessible.

4.2 Fine Tuning

We fine-tune both language models by training the last 2, 4 or 6 layers with the classification head, both with and without adding the business-name prefix, e.g. 'Fish and Chip restaurant' or 'Logistics and Transport', to the inputs (+ b.n.). See A.1 for full implementation details. Results for input B are reported in Table 6 (Figure A4 shows the confusion matrix), and full classification reports for all inputs in Table A9-A13.

Fine-tuning substantially improves the performance over the head-only baselines, closing the gap and surpassing the vendor baseline. Fine-tuning SBERT substantially improves performance in the *Director's loan account (DLA)* category. *DLA* and *Purchases* are financially different categories for which invoices may be identical. Loosely, the difference does not lie in the goods or services bought, but in whether the director or the business paid, and whether the purchase is in furtherance of the business. The per-class F1 reaches 0.53 (6 layers no b.n.) and 0.71 adding the business name prefix (2 layers + b.n.), an improvement of 27 percentage points over vendor baseline. Similarly, fine-tuned SBERT improves by 10% on the vendor baseline on the *Stationery & Postage* category. In Section 4.1, we showed that vendor name does account for a large part of the variation in invoices, but the rich text description is essential on the long tail of 'hard invoices', which is significant in a domain with inherent high-class imbalance.

Despite being far larger, the zero-shot LLM (Qwen3-4B) baseline underperformed SBERT both with frozen encoder and especially after fine-tuning ($+0.55$ M F1). This is consistent with the literature (Bucher and Martini, 2024), and corroborates our hypothesis that an in-house fine-tuned SLM is preferable to an off-the-shelf LLM in this computationally constrained setting.

DeBERTa is less robust, fine-tuning successfully only at depth 2. This is most likely due to the large number of DeBERTa parameters relative to our sample size. Similarly to the geometric properties, DeBERTa with mean pooling shows the same qualitative pattern as DeBERTa CLS but less extreme, and it still underperforms SBERT.

Figure 3 shows the first three principal components of the SBERT embeddings after fine-tuning

	DeBERTa CLS			DeBERTa mean			SBERT			Vendor	LLM
	A	B	C	A	B	C	A	B	C	baseline	baseline
Acc.	$0.39_{\pm 0.02}$	$0.62_{\pm 0.02}$	$0.45_{\pm 0.04}$	$0.51_{\pm 0.02}$	$0.63_{\pm 0.02}$	$0.62_{\pm 0.02}$	$0.70_{\pm 0.04}$	$\underline{0.87}_{\pm 0.01}$	$0.83_{\pm 0.02}$	$\mathbf{0.89}_{\pm 0.01}$	$0.52_{\pm 0.02}$
M F1	$0.18_{\pm 0.08}$	$0.55_{\pm 0.02}$	$0.33_{\pm 0.06}$	$0.39_{\pm 0.03}$	$0.53_{\pm 0.04}$	$0.49_{\pm 0.02}$	$0.67_{\pm 0.06}$	$\underline{0.80}_{\pm 0.01}$	$0.77_{\pm 0.02}$	$\mathbf{0.87}_{\pm 0.01}$	$0.39_{\pm 0.02}$
W F1	$0.31_{\pm 0.05}$	$0.62_{\pm 0.02}$	$0.44_{\pm 0.04}$	$0.49_{\pm 0.01}$	$0.63_{\pm 0.01}$	$0.62_{\pm 0.02}$	$0.69_{\pm 0.04}$	$\underline{0.85}_{\pm 0.01}$	$0.82_{\pm 0.02}$	$\mathbf{0.89}_{\pm 0.01}$	$0.52_{\pm 0.02}$

Table 5: **Frozen Transformers** accuracy, macro F1 and weighted F1 for classification with frozen transformers for each model and input type. The LLM baseline is zero-shot, and the transformers use a trained linear head. We report the mean±s.d. over 5 splits. The best performance is in **bold**, the second best is underlined.

Model	Acc.	M F1	W F1
SBERT 2l	$0.945_{\pm 0.00}$	$0.915_{\pm 0.01}$	$0.944_{\pm 0.00}$
SBERT 4l	$0.950_{\pm 0.01}$	$0.924_{\pm 0.02}$	$0.950_{\pm 0.01}$
SBERT 6l	$0.950_{\pm 0.01}$	$0.926_{\pm 0.01}$	$0.949_{\pm 0.01}$
SBERT 2l b.n.	$\mathbf{0.961}_{\pm 0.00}$	$\mathbf{0.939}_{\pm 0.01}$	$0.960_{\pm 0.00}$
SBERT 4l b.n.	$\mathbf{0.961}_{\pm 0.00}$	$0.938_{\pm 0.02}$	$\mathbf{0.961}_{\pm 0.01}$
SBERT 6l b.n.	$\mathbf{0.961}_{\pm 0.01}$	$0.936_{\pm 0.02}$	$0.960_{\pm 0.01}$
DeB. CLS 2l	$0.901_{\pm 0.02}$	$0.855_{\pm 0.04}$	$0.901_{\pm 0.02}$
DeB. CLS 4l	$0.381_{\pm 0.05}$	$0.149_{\pm 0.04}$	$0.323_{\pm 0.04}$
DeB. CLS 6l	$0.352_{\pm 0.10}$	$0.131_{\pm 0.13}$	$0.207_{\pm 0.04}$
DeB. CLS 2l b.n.	$0.913_{\pm 0.02}$	$0.870_{\pm 0.03}$	$0.915_{\pm 0.02}$
DeB. CLS 4l b.n.	$0.457_{\pm 0.19}$	$0.210_{\pm 0.25}$	$0.382_{\pm 0.24}$
DeB. CLS 6l b.n.	$0.344_{\pm 0.03}$	$0.077_{\pm 0.04}$	$0.256_{\pm 0.14}$
DeB. m 2l	$0.930_{\pm 0.01}$	$0.894_{\pm 0.02}$	$0.930_{\pm 0.01}$
DeB. m 4l	$0.779_{\pm 0.10}$	$0.618_{\pm 0.19}$	$0.757_{\pm 0.13}$
DeB. m 6l	$0.366_{\pm 0.06}$	$0.192_{\pm 0.14}$	$0.302_{\pm 0.14}$
DeB. m 2l b.n.	$0.950_{\pm 0.01}$	$0.916_{\pm 0.02}$	$0.950_{\pm 0.01}$
DeB. m 4l b.n.	$0.818_{\pm 0.04}$	$0.673_{\pm 0.11}$	$0.807_{\pm 0.05}$
DeB. m 6l b.n.	$0.480_{\pm 0.14}$	$0.245_{\pm 0.04}$	$0.389_{\pm 0.12}$

Table 6: **Post fine-tuning** accuracy, macro F1 and weighted F1 across fine-tuning configurations and number of unfrozen final layers (l) for input type B with/without business-nature prefix (+b.n.) reporting mean±s.d. over 5 splits. The best performance is in **bold**.

the last two layers. Compared to the X shaped manifold of Figure 2, fine-tuning has 'unravelled' the manifold that now qualitatively resembles a simplex with vertices at *Petrol & Oil* (orange), *Purchases* (green), *Motor Repairs and Servicing* (blue), and the 'recurring bills' which includes for example *Accountancy Fee*, *Subscriptions* and *Telephone, Fax & Internet*. *DLA*, which is semantically close to *Purchases*, is now much better separated from it, consistent with the large per-class F1 improvement. A simplex geometry is considered to be 'optimal' for classification (Weinan and Wojtowytsch, 2022) because the classes fall on the vertices and irregularities pull points towards the centre, opening a pathway to uncertainty calibration (Kull et al., 2019), which we leave for future work.

4.3 Generalise to new clients

In Section 4.2, we stratified the training and testing data per category due to high class imbalance, so a given client's data likely appeared in both sets. We now investigate performance for a previously unseen client. For practitioners, this simulates onboarding a new client.

We leave one client, the 'test client', out of our dataset at a time and split the remaining data 80/20 into training and validation sets, respectively. Then we repeat the experiment, injecting a fraction ϵ of the 'test client' data in the train/val set (and removing it from the test set). We fine-tune from scratch the best configurations of SBERT and DeBERTa from Section 4.2 for $\epsilon \in [0, 0.8]$.

Figure 4 reports the average macro F1 score on the 'testing client' as a function of ϵ. For a new client, the vendor baseline, which was competitive in-distribution, severely underperforms both models, since performance depends entirely on the overlap between the new client's suppliers and those seen in training. Between SBERT and DeBERTa, the former reaches higher F1 across all values of ϵ: at $\epsilon = 0$, corresponding to a fully unseen client, SBERT achieves an F1 10% higher than DeBERTa. This shows that the value of learned representations relies on their generalisation abilities: line-item text carries semantic signal that transfers to unseen vendors. The gap between SBERT and DeBERTa highlights the extent to which that signal is more easily recoverable from semantically meaningful representations with better geometry.

Both models need a warm-up period before reaching reliable performance; to achieve an F1 of 0.9 or higher, SBERT needs, on average, 100 client-specific data points, whereas DeBERTa needs ∼75% more client-specific data (174 data points). Beyond the warm-up region, both SBERT and DeBERTa continue to improve with ϵ and converge towards their fine-tuned baselines obtained in Section 4.2 (dashed lines in Figure 4).

This pattern is consistent with the geometric analysis and the overall findings of this paper. SBERT's higher-dimensional, less anisotropic space utilises more independent directions where a client's invoices can be separated from a few labelled examples. The generalisation gap at $\epsilon = 0$

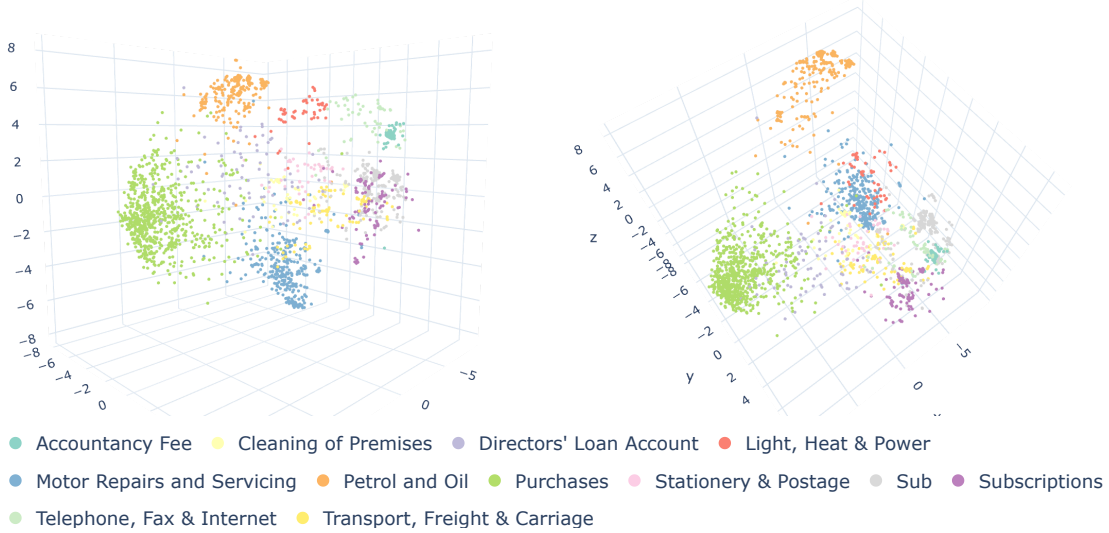

Figure 3: **Post fine-tuning** visualisation of the first three principal components of the representation space of SBERT with input B. Compared to Figure 2, note that classes appear more distributed over the sphere, with Purchases (green) in the same direction but separated from Directors Loan Accounts (light purple).

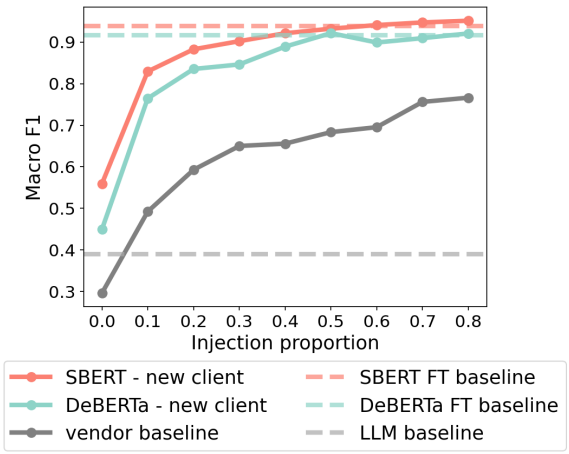

Figure 4: **Left-out client experiment**, Macro F1 for fine-tuned SBERT and DeBERTa as a function of the proportion of 'new client' data injected into the training set (solid line). The baselines (dashed lines) are from Section 4.2.

and the faster warm-up confirm that SBERT is the more robust choice in our setting, corroborating that pre-trained embedding geometry is linked to fine-tuned accuracy and robustness.

5 Conclusion

Whilst this experiment uses a small dataset covering a subset of all existing categories, it is a promising result for an in-house deployment where cost, data confidentiality, and interpretability matter. We explored geometric properties which demonstrated that previous token-level findings, that embeddings are globally anisotropic with locally isotropic clusters, apply to the sentence embedding of invoices.

While exploiting vendor identity alone reaches respectable performance, SLM became truly useful once fine-tuned, especially on smaller categories and for newly onboarded clients. This improvement when using full invoice content implies that 'line item' text has been at least partially mapped to the label through semantic meaning. Moreover, whilst both models achieved similar high accuracy under naive comparison, sentence embedding produced more robust embedding that generalised better than token embeddings, consistent with the hypothesis that geometry relates to both performance and robustness.

Fine-tuned performance on the structured Input C is comparable to Input B, which is particularly relevant for practitioners. Since the structured input remains uninformative after fine-tuning, we conclude that omitting labelling of fields in the free text OCR is sufficient. This removes an additional human-in-the-loop step, resulting in a reduction in cost and complexity. Input B requires only a single human-in-the-loop at the end, which is irreducible to comply with regulations.

These results suggest a promising direction for future research towards an uncertainty-aware model to identify the data that is confidently assigned with interpretable reasons, focusing expert attention where it is most needed.

Limitations

Our study is based on a single confidential dataset from one UK accountancy firm, comprising 2,168 invoices across 12 categories from seven clients. Real, labelled invoice data of this kind is difficult to obtain, and this corpus reflects the access we were able to secure; we acknowledge that the sample is modest. Our results should be seen as evidence that an in-house SLM pipeline is viable in this setting, but they may not transfer directly to firms operating with different accounting conventions, in different domains or countries.

Our one-shot LLM was also limited in scale and larger models may perform differently, and fine-tuning an open-weight LLM is likely to be performant given enough data, though this may require additionally training it as a sentence classifier. We have focussed here on SLM performance as the most promising locally-implementable method, but baselines including fuzzy matching of vendors and traditional bag-of-words models should be quantified before deployment.

Our geometric analysis, and its connection to classification performance, is based on a comparison of two model families. To further explore this connection and make an explanatory claim, more model types and additional experiments should be analysed. We used a single set of standard training parameters for both SBERT and DeBERTa for a consistent comparison; however, this study doesn't include an exhaustive search of the hyperparameter space. Whilst we observed CLS collapse on this dataset, this phenomenon should be investigated with more experiments with controlled ablations and bigger, more varied datasets, to confirm it as a general phenomenon in sentence transformers. We also omitted a formal model testing procedure as this would not address the more fundamental data limitations.

Ethical considerations

The data analysed in this paper is confidential: it is commercially sensitive and may contain identifying information about businesses and their suppliers. The data has been used under the firm's data-handling agreements, and it will not be released. Where representations are visualised, vendor identities and legends are removed for data privacy.

We do not advocate for a fully automated pipeline, and we stress the importance of humans-in-the-loop with professional accounting skills. This material is intended for academic and educational purposes only, and the authors and their institutions disclaim liability for any outcomes arising from its use.

We partially release the code, removing any identifying information and preserving part of the code to protect the firm's IP.

References

Mira Ait-Saada and Mohamed Nadif. 2023. Is anisotropy truly harmful? a case study on text clustering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1194–1203, Toronto, Canada. Association for Computational Linguistics.

Aluffi Pietro Alessandro, Brandi Jess, Marya Bazzi, Kate Kennedy, Matt Arderne, Daniel Rodrigues, and Martin Lotz. 2025. Categorising SME bank transactions with machine learning and synthetic data generation. *arXiv preprint arXiv:2508.05425*.

Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. 2019. A convergence theory for deep learning via over-parameterization. In *International conference on machine learning*, pages 242–252. PMLR.

Usman Anwar, Abulhair Saparov, Javier Rando, Daniel Paleka, Miles Turpin, Peter Hase, Ekdeep Singh Lubana, Erik Jenner, Stephen Casper, Oliver Sourbut, Benjamin L. Edelman, Zhaowei Zhang, Mario Günther, Anton Korinek, Jose Hernandez-Orallo, Lewis Hammond, Eric J Bigelow, Alexander Pan, Lauro Langosco, and 23 others. 2024. Foundational challenges in assuring alignment and safety of large language models. *Transactions on Machine Learning Research*. Survey Certification, Expert Certification.

Chiara Bardelli, Alessandro Rondinelli, Ruggero Vecchio, and Silvia Figini. 2020. Automatic electronic invoice classification using machine learning models. *Machine Learning and Knowledge Extraction*, 2(4):617–629.

Peter Belcak, Greg Heinrich, Shizhe Diao, Yonggan Fu, Xin Dong, Saurav Muralidharan, Yingyan Celine Lin, and Pavlo Molchanov. 2025. Small language models are the future of agentic ai. *arXiv preprint arXiv:2506.02153*.

David Berghaus, Armin Berger, Lars Hillebrand, Kostadin Cvejoski, and Rafet Sifa. 2025. Multimodal vision vs. text-based parsing: Benchmarking llm strategies for invoice processing. pages 2519–2526.

Martin Juan José Bucher and Marco Martini. 2024. Fine-tuned'small'LLMs (still) significantly outperform zero-shot generative AI models in text classification. *arXiv preprint arXiv:2406.08660*.

Xingyu Cai, Jiaji Huang, Yuchen Bian, and Kenneth Church. 2021. Isotropy in the contextual embedding space: Clusters and manifolds. In *International conference on learning representations*.

Yasheng Chen, Xian Huang, and Jing Cao. 2025. A framework for applying machine learning and natural language processing methods to accounting recognition. *Accounting Horizons*, 39(4):45–55.

Wei Wen Chi, Tiong Yew Tang, Narishah Mohamed Salleh, Muaadh Mukred, Hussain AlSalman, and Muhammad Zohaib. 2024. Data augmentation with semantic enrichment for deep learning invoice text classification. *IEEE Access*, 12:57326–57344.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL workshop BlackboxNLP: analyzing and interpreting neural networks for NLP*, pages 276–286.

Badhan Chandra Das, M Hadi Amini, and Yanzhao Wu. 2025. Security and privacy challenges of large language models: A survey. *ACM Computing Surveys*, 57(6):1–39.

Finale Doshi-Velez and Been Kim. 2017. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? comparing the geometry of BERT, ELMo, and GPT-2 embeddings. *arXiv preprint arXiv:1909.00512*.

Jun Gao, Di He, Xu Tan, Tao Qin, Liwei Wang, and Tieyan Liu. 2019. Representation degeneration problem in training natural language generation models. In *International Conference on Learning Representations*.

Silvia Garcia-Mendez, Milagros Fernandez-Gavilanes, Jonathan Juncal-Martinez, Francisco Javier González-Castaño, and Óscar Barba Seara. 2020. Identifying banking transaction descriptions via support vector machine short-text classification based on a specialized labelled corpus. *Ieee Access*, 8:61642–61655.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. 2020. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673.

Katharina Hämmerl, Alina Fastowski, Jindřich Libovickỳ, and Alexander Fraser. 2023. Exploring anisotropy and outliers in multilingual language models for cross-lingual semantic sentence similarity. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 7023–7037.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. DeBERTaV3: Improving DeBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. *Preprint*, arXiv:2111.09543.

John Hewitt and Christopher D Manning. 2019. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4129–4138.

HM Revenue and Customs. 2026. Measuring tax gaps 2026 edition: tax gap estimates for 2024 to 2025.

Rasmus Kær Jørgensen and Christian Igel. 2021. Machine learning for financial transaction classification across companies using character-level word embeddings of text fields. *Intelligent Systems in Accounting, Finance and Management*, 28(3):159–172.

Diego Santos Kieckbusch, Geraldo Pereira Rocha Filho, Vinicius Di Oliveira, and Li Weigang. 2020. Towards intelligent processing of electronic invoices: The general framework and case study of short text deep learning in brazil. In *International Conference on Web Information Systems and Technologies*, pages 74–92. Springer.

Pakorn Kotepuchai and Yachai Limpiyakorn. 2024. Tree-based classifiers for smart general ledger code suggestion. In *Proceedings of the 2024 13th International Conference on Informatics, Environment, Energy and Applications*, pages 17–22.

Meelis Kull, Miquel Perello-Nieto, Markus Kängsepp, Telmo Silva Filho, Hao Song, and Peter Flach. 2019. *Beyond temperature scaling: obtaining well-calibrated multiclass probabilities with Dirichlet calibration*. Curran Associates Inc., Red Hook, NY, USA.

Yuxin Liang, Rui Cao, Jie Zheng, Jie Ren, and Ling Gao. 2021. Learning to remove: Towards isotropic pre-trained BERT embedding. In *International conference on artificial neural networks*, pages 448–459. Springer.

Ziyi Liang, Yanfei Zhou, and Matteo Sesia. 2023. Conformal inference is (almost) free for neural networks trained with early stopping. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Juan Liu, Lei Pei, Ying Sun, Heather Simpson, Jocelyn Lu, and Nhung Ho. 2021. Categorization of financial transactions in quickbooks. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 3299–3307.

Haoyan Luo and Lucia Specia. 2024. From understanding to utilization: A survey on explainability for large language models. *arXiv preprint arXiv:2401.12874*.

Timothee Mickus, Raul Vazquez, and Joseph Attieh. 2024. I have an attention bridge to sell you: Generalization capabilities of modular translation architectures. In *Proceedings of the Fifth Workshop on Insights from Negative Results in NLP*, pages 34–40, Mexico City, Mexico. Association for Computational Linguistics.

Justin Munoz, Mahdi Jalili, and Laleh Tafakori. 2022. Hierarchical classification for account code suggestion. *Knowledge-Based Systems*, 251:109302.

Vivi Nastase and Paola Merlo. 2025. Testing the assumptions about the geometry of sentence embedding spaces: the cosine measure need not apply. *arXiv preprint arXiv:2509.01606*.

Sander Noels, Sébastien Viaene, and Tijl De Bie. 2024. TopoledgerBERT: Topological learning of ledger description embeddings using siamese BERT-networks. In *Proceedings of the Eighth Financial Technology and Natural Language Processing and the 1st Agent AI for Scenario Planning*, pages 63–72.

Qwen Team. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Sara Rajaee and Mohammad Taher Pilehvar. 2021. How does fine-tuning affect the geometry of embedding space: A case study on isotropy. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3042–3049.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in BERTology: What we know about how BERT works. *Transactions of the association for computational linguistics*, 8:842–866.

Cynthia Rudin. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215.

Andrew M. Saxe, James L. McClelland, and Surya Ganguli. 2019. A mathematical theory of semantic development in deep neural networks. *Proceedings of the National Academy of Sciences*, 116(23):11537–11546.

Ahmad S Tarawneh, Ahmad B Hassanat, Dmitry Chetverikov, Imre Lendak, and Chaman Verma. 2019. Invoice classification using deep features and machine learning techniques. In *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, pages 855–859. IEEE.

Shang Wang, Tianqing Zhu, Bo Liu, Ming Ding, Dayong Ye, Wanlei Zhou, and Philip Yu. 2025. Unique security and privacy threats of large language models: A comprehensive survey. *ACM Computing Surveys*, 58(4):1–36.

E Weinan and Stephan Wojtowytsch. 2022. On the emergence of simplex symmetry in the final and penultimate layers of neural network classifiers. In *Mathematical and Scientific Machine Learning*, pages 270–290. PMLR.

A Appendix

A.1 Experimental details.

Frozen transformers.

For each model and input type, we freeze the encoder and train a linear classification head on top of the sentence embedding. We use a cross-entropy loss weighted by inverse class frequency, the AdamW optimiser with $5e{-}5$ learning rate and linear rate, batch size 16, and up to 60 epochs with early stopping on validation weighted F1.The dataset is split 60/20/20 into train, validation and test sets with stratified sampling. For DeBERTa CLS, we use the model's standard sequence-classification head; for both SBERT and DeBERTa, we use standard mean pooling.

LLM baseline. We implement a baseline for zero-shot classification using Qwen3-4B (Qwen Team, 2025), a small model matching the in-house deployment computation resources constraint. Due to data confidentiality we excluded API-hosted LLMs, which could improve performance. We classify each invoice independently with the following prompt: *You are an expert UK bookkeeper assigning invoices to General Ledger categories. Choose exactly one category from the list. Definitions: {}. Invoice: {}. Only answer with the name of the category, do not answer with any other text*. The category definitions were provided by the firm (see Table A1), and the invoice text is as in input B. We disable thinking mode and use greedy decoding for reproducibility. We used a deterministic parser to map the model outputs to a canonical label and count as error outputs that couldn't be parsed to a category.

Fine tuning We fine-tune both language models by training the last 2, 4 or 6 layers with the classification head. We use a smaller learning rate of $1e{-}5$ for the transformer layers than for the classification head ($5e{-}5$), and optimise the two parameter groups jointly with AdamW and add 10% linear warm-up which helps DeBERTa to avoid early divergence. Each configuration is run with and without the business-name prefix (+ b.n.). The rest of the parameters are the same.

Computing resources. Fine-tuning was performed on an NVIDIA RTX PRO 500 Black with 6 GB VRAM, using CUDA 13.2 and driver version 595.71.05. Finetuning SBERT (2 layers + b.n.) took about 10 minutes for 46 epochs, and DeBERTa (2 layers + b.n.) took approximately 15 minutes for 31 epochs. An inference pass on the 434 test invoices for Qwen3-4B took approximately 5 minutes.

A.2 Supplementary Figures

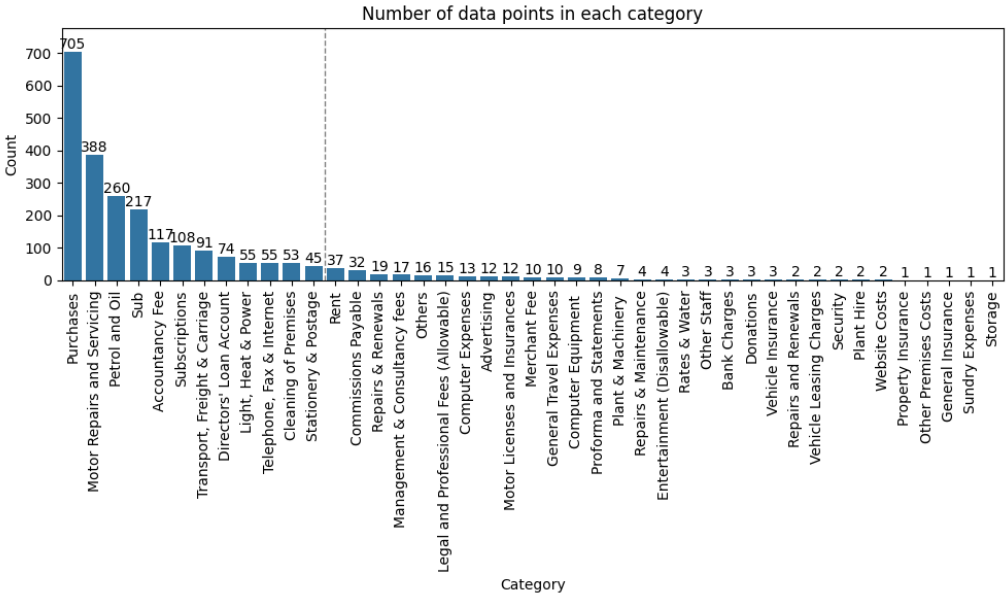

Figure A1: Distribution of invoices across the 42 accounting categories in the full dataset; the categories before the dashed line are included in the analysis.

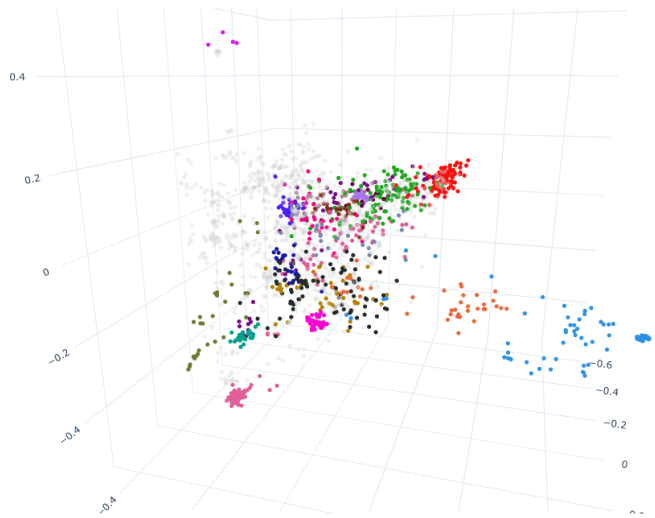

Figure A2: Visualisation of the first three principal components of the representation space of pre-trained SBERT with input B. Points coloured by Vendor name for the 20 most popular vendors, grey otherwise. Legend removed for data privacy.

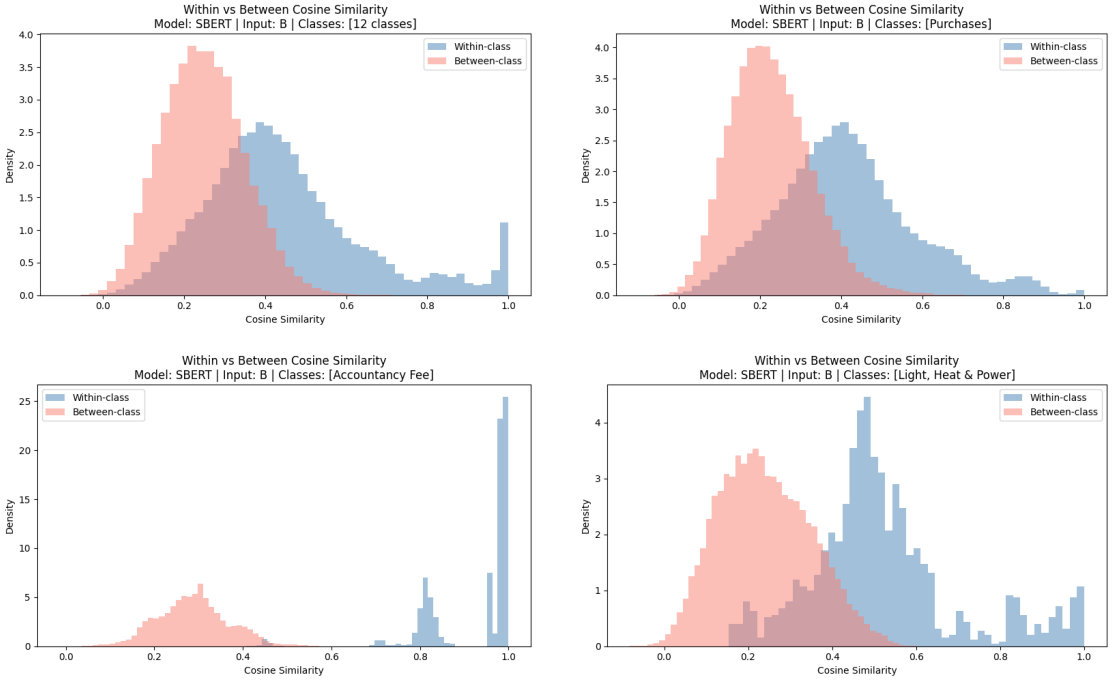

Figure A3: Within and between cosine similarity using SBERT Frozen transformer on input B, showing all-class average, Purchases, Accountancy Fee and Light, Heat & Power. With the exception of Accountancy Fee, they all overlap showing that the direction of the angular variation in embedding space is important for successful classification.

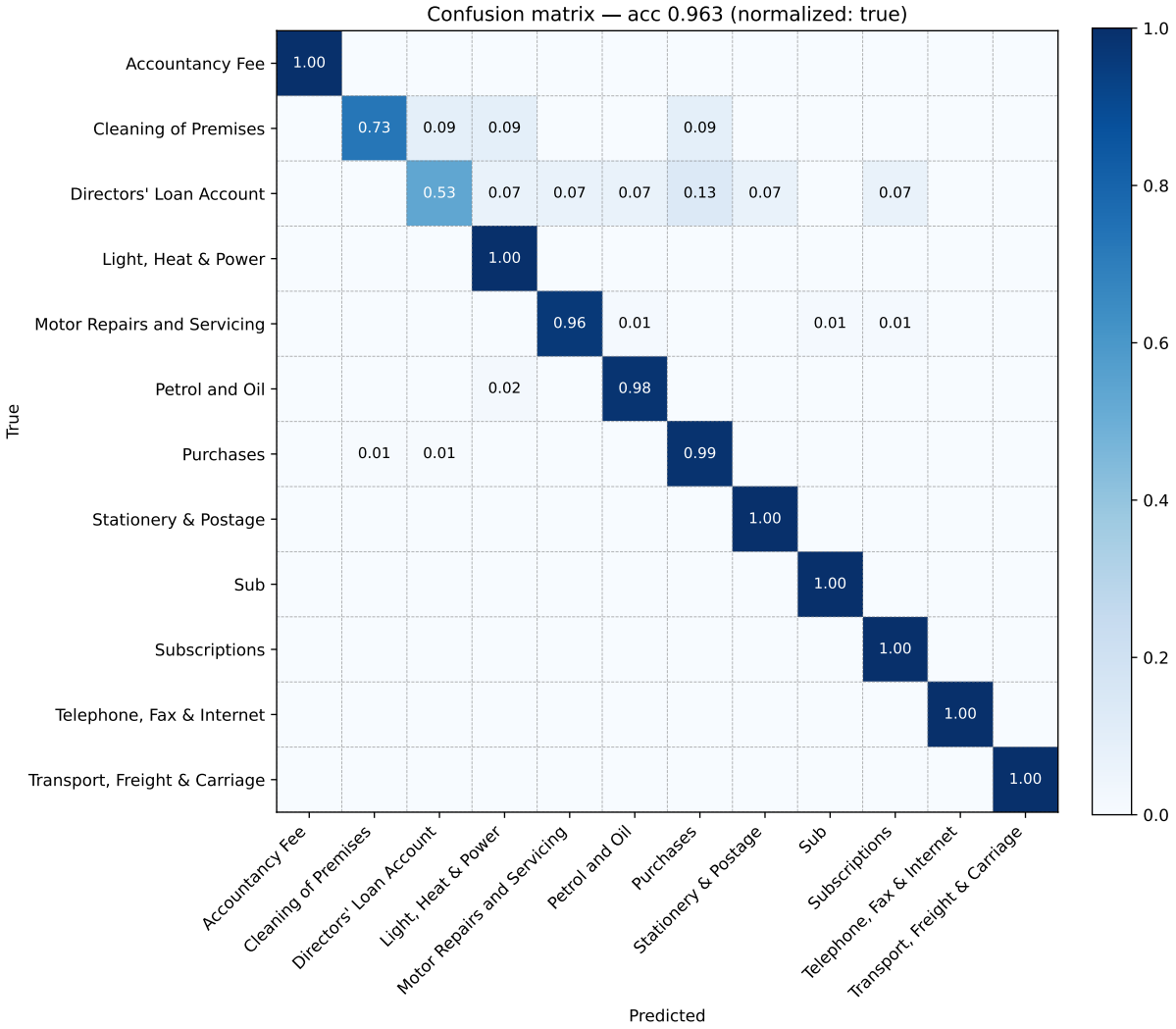

Figure A4: Confusion matrix for SBERT fine-tuned with 2 layers and business name/nature prefix.

A.3 Supplementary Tables

Account Category	Definition
Purchases	Cost of goods, raw materials or components bought for resale or for consumption in production during the period. A profit-and-loss item forming part of cost of sales; it records the buying flow, not stock held. Excludes capital assets and overhead services.
Directors' Loan Account	Related-party control account for amounts a director personally lends to, or withdraws from, the company other than salary, dividend or reimbursed expenses. May be in credit (company owes the director — a liability) or overdrawn (director owes the company — an asset); overdrawn balances attract a s455 CTA 2010 charge and beneficial-loan implications.
Petrol and Oil	Fuel and lubricant costs incurred in operating business motor vehicles.
Sub	Abbreviation for Subcontractor. Payments for subcontractor or subcontract labour directly attributable to revenue-generating work.
Motor Repairs and Servicing	Maintenance, servicing and repair of company vehicles.
Subscriptions	Recurring professional-body memberships, journals and software or service subscriptions of an administrative nature.
Stationery & Postage	Office consumables, printing materials and postal or courier charges.
Cleaning of Premises	Janitorial costs of keeping premises clean — cleaning staff or contractors, services and cleaning consumables. Excludes repair and upkeep of premises, which is maintenance (see Repairs & Renewals).
Light, Heat & Power	Utility costs — electricity, gas, heating and power — consumed in operating business premises.
Telephone, Fax & Internet	Communication costs: telephone, broadband, internet access and related charges
Transport, Freight & Carriage	Inwards - Cost of transporting purchased goods or materials into the business. Added to the cost of purchases and therefore part of cost of sales. Outwards - Cost of delivering finished goods out to customers. A distribution/selling overhead, not cost of sales.
Accountancy Fee	Fees for preparation of statutory and/or management accounts, bookkeeping, tax compliance and related non-audit advisory services. Excludes statutory audit.

Table A1: Definitions for accounting categories used in the dataset. There is no standardised UK nominal ledger and nominal ledger code numbers are a convention of the different accounting software. The UK firm that provided our data uses a custom accounting categorisation and proprietary software. When an invoice contains line items spanning multiple categories, the invoice is assigned to the category with the largest total value, as is standard practice in this UK firm.

Table A2: Average within-class and between-class cosine similarity, using the true labels.

(a) Uncentered

Type	Model	Within	Between
A	SBERT	0.503	0.348
A	DeBERTa CLS	0.999	0.999
A	DeBERTa mean	0.977	0.972
B	SBERT	0.478	0.249
B	DeBERTa CLS	0.995	0.993
B	DeBERTa mean	0.938	0.930
C	SBERT	0.641	0.485
C	DeBERTa CLS	0.997	0.995
C	DeBERTa mean	0.956	0.948

(b) Centred

Type	Model	Within	Between
A	SBERT	-0.005	0.000
A	DeBERTa CLS	0.000	0.000
A	DeBERTa mean	0.000	0.002
B	SBERT	0.003	0.000
B	DeBERTa CLS	0.024	0.016
B	DeBERTa mean	0.013	0.009
C	SBERT	0.003	0.000
C	DeBERTa CLS	0.041	0.029
C	DeBERTa mean	0.019	0.011

Table A3: Average within-cluster and between-cluster cosine similarity.

(a) Uncentered

Type	Model	Within	Between
A	SBERT	0.820	0.341
A	DeBERTa CLS	0.999	0.999
A	DeBERTa mean	0.991	0.970
B	SBERT	0.708	0.265
B	DeBERTa CLS	0.997	0.981
B	DeBERTa mean	0.942	0.841
C	SBERT	0.823	0.476
C	DeBERTa CLS	0.998	0.986
C	DeBERTa mean	0.953	0.864

(b) Centred

Type	Model	Within	Between
A	SBERT	-0.005	0.000
A	DeBERTa CLS	0.000	0.000
A	DeBERTa mean	0.000	0.002
B	SBERT	0.003	0.000
B	DeBERTa CLS	0.024	0.016
B	DeBERTa mean	0.013	0.009
C	SBERT	0.003	0.000
C	DeBERTa CLS	0.041	0.029
C	DeBERTa mean	0.019	0.011

Table A4: Classification Report (precision, recall, F1) - SBERT. We report the mean±s.d. over 5 splits.

	Input A			**Input B**			**Input C**		
Class	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
Accountancy Fee	$0.91_{\pm 0.02}$	$0.99_{\pm 0.02}$	$0.95_{\pm 0.01}$	$0.79_{\pm 0.06}$	$0.99_{\pm 0.02}$	$0.88_{\pm 0.04}$	$0.64_{\pm 0.06}$	$0.99_{\pm 0.02}$	$0.78_{\pm 0.05}$
Cleaning of Premises	$0.74_{\pm 0.09}$	$0.55_{\pm 0.14}$	$0.63_{\pm 0.13}$	$0.92_{\pm 0.07}$	$0.65_{\pm 0.08}$	$0.76_{\pm 0.08}$	$0.93_{\pm 0.10}$	$0.58_{\pm 0.15}$	$0.71_{\pm 0.14}$
Directors' Loan Account	$0.48_{\pm 0.38}$	$0.11_{\pm 0.08}$	$0.17_{\pm 0.13}$	$0.53_{\pm 0.36}$	$0.07_{\pm 0.05}$	$0.12_{\pm 0.08}$	$0.50_{\pm 0.50}$	$0.00_{\pm 0.06}$	$0.10_{\pm 0.10}$
Light, Heat & Power	$0.86_{\pm 0.13}$	$0.71_{\pm 0.24}$	$0.76_{\pm 0.20}$	$0.94_{\pm 0.07}$	$0.96_{\pm 0.05}$	$0.95_{\pm 0.02}$	$0.95_{\pm 0.07}$	$0.95_{\pm 0.08}$	$0.95_{\pm 0.04}$
Motor Repairs and Servicing	$0.77_{\pm 0.04}$	$0.87_{\pm 0.04}$	$0.82_{\pm 0.04}$	$0.90_{\pm 0.06}$	$0.91_{\pm 0.03}$	$0.90_{\pm 0.03}$	$0.92_{\pm 0.02}$	$0.87_{\pm 0.02}$	$0.90_{\pm 0.02}$
Petrol and Oil	$0.47_{\pm 0.02}$	$0.81_{\pm 0.04}$	$0.60_{\pm 0.03}$	$0.81_{\pm 0.07}$	$0.94_{\pm 0.03}$	$0.87_{\pm 0.04}$	$0.74_{\pm 0.06}$	$0.92_{\pm 0.02}$	$0.82_{\pm 0.04}$
Purchases	$0.86_{\pm 0.04}$	$0.53_{\pm 0.04}$	$0.66_{\pm 0.04}$	$0.90_{\pm 0.02}$	$0.89_{\pm 0.03}$	$0.89_{\pm 0.02}$	$0.90_{\pm 0.03}$	$0.83_{\pm 0.04}$	$0.86_{\pm 0.03}$
Stationery & Postage	$0.81_{\pm 0.14}$	$0.69_{\pm 0.09}$	$0.74_{\pm 0.10}$	$0.91_{\pm 0.13}$	$0.82_{\pm 0.10}$	$0.85_{\pm 0.06}$	$0.94_{\pm 0.13}$	$0.71_{\pm 0.10}$	$0.80_{\pm 0.08}$
Sub	$0.80_{\pm 0.04}$	$0.70_{\pm 0.07}$	$0.75_{\pm 0.05}$	$0.93_{\pm 0.01}$	$0.87_{\pm 0.05}$	$0.90_{\pm 0.03}$	$0.73_{\pm 0.07}$	$0.94_{\pm 0.03}$	$0.82_{\pm 0.03}$
Subscriptions	$0.59_{\pm 0.06}$	$0.85_{\pm 0.11}$	$0.70_{\pm 0.08}$	$0.81_{\pm 0.03}$	$0.82_{\pm 0.09}$	$0.81_{\pm 0.04}$	$0.90_{\pm 0.04}$	$0.75_{\pm 0.09}$	$0.81_{\pm 0.07}$
Telephone, Fax & Internet	$0.37_{\pm 0.11}$	$0.73_{\pm 0.17}$	$0.49_{\pm 0.13}$	$0.76_{\pm 0.11}$	$1.00_{\pm 0.00}$	$0.86_{\pm 0.07}$	$0.75_{\pm 0.08}$	$0.96_{\pm 0.05}$	$0.84_{\pm 0.05}$
Transport, Freight & Carriage	$0.60_{\pm 0.11}$	$0.90_{\pm 0.10}$	$0.72_{\pm 0.11}$	$0.83_{\pm 0.07}$	$0.89_{\pm 0.09}$	$0.86_{\pm 0.06}$	$0.94_{\pm 0.08}$	$0.77_{\pm 0.11}$	$0.84_{\pm 0.07}$

Table A5: Classification Report (precision, recall, F1) - DeBERTa CLS. We report the mean±s.d. over 5 splits.

	Input A			**Input B**			**Input C**		
Class	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
Accountancy Fee	$0.15_{\pm 0.15}$	$0.59_{\pm 0.54}$	$0.23_{\pm 0.23}$	$0.81_{\pm 0.08}$	$0.97_{\pm 0.02}$	$0.88_{\pm 0.05}$	$0.61_{\pm 0.24}$	$0.83_{\pm 0.04}$	$0.68_{\pm 0.17}$
Cleaning of Premises	$0.00_{\pm 0.00}$	$0.00_{\pm 0.00}$	$0.00_{\pm 0.00}$	$0.56_{\pm 0.27}$	$0.80_{\pm 0.04}$	$0.62_{\pm 0.17}$	$0.31_{\pm 0.19}$	$0.71_{\pm 0.22}$	$0.41_{\pm 0.19}$
Directors' Loan Account	$0.01_{\pm 0.02}$	$0.03_{\pm 0.06}$	$0.01_{\pm 0.03}$	$0.15_{\pm 0.29}$	$0.04_{\pm 0.06}$	$0.06_{\pm 0.10}$	$0.20_{\pm 0.45}$	$0.01_{\pm 0.03}$	$0.03_{\pm 0.06}$
Light, Heat & Power	$0.22_{\pm 0.44}$	$0.16_{\pm 0.28}$	$0.10_{\pm 0.14}$	$0.57_{\pm 0.21}$	$0.65_{\pm 0.10}$	$0.60_{\pm 0.15}$	$0.10_{\pm 0.19}$	$0.20_{\pm 0.28}$	$0.13_{\pm 0.21}$
Motor Repairs and Servicing	$0.52_{\pm 0.09}$	$0.65_{\pm 0.14}$	$0.56_{\pm 0.05}$	$0.78_{\pm 0.05}$	$0.61_{\pm 0.04}$	$0.68_{\pm 0.03}$	$0.51_{\pm 0.04}$	$0.55_{\pm 0.06}$	$0.53_{\pm 0.03}$
Petrol and Oil	$0.05_{\pm 0.11}$	$0.06_{\pm 0.14}$	$0.05_{\pm 0.12}$	$0.52_{\pm 0.14}$	$0.83_{\pm 0.08}$	$0.62_{\pm 0.06}$	$0.44_{\pm 0.10}$	$0.67_{\pm 0.17}$	$0.51_{\pm 0.07}$
Purchases	$0.52_{\pm 0.14}$	$0.55_{\pm 0.21}$	$0.49_{\pm 0.06}$	$0.86_{\pm 0.03}$	$0.55_{\pm 0.09}$	$0.67_{\pm 0.06}$	$0.84_{\pm 0.06}$	$0.43_{\pm 0.07}$	$0.57_{\pm 0.06}$
Stationery & Postage	$0.16_{\pm 0.23}$	$0.13_{\pm 0.20}$	$0.13_{\pm 0.18}$	$0.44_{\pm 0.12}$	$0.29_{\pm 0.13}$	$0.33_{\pm 0.11}$	$0.05_{\pm 0.07}$	$0.07_{\pm 0.10}$	$0.06_{\pm 0.08}$
Sub	$0.29_{\pm 0.29}$	$0.30_{\pm 0.28}$	$0.29_{\pm 0.27}$	$0.60_{\pm 0.07}$	$0.80_{\pm 0.07}$	$0.68_{\pm 0.05}$	$0.53_{\pm 0.10}$	$0.70_{\pm 0.14}$	$0.59_{\pm 0.05}$
Subscriptions	$0.08_{\pm 0.12}$	$0.07_{\pm 0.14}$	$0.07_{\pm 0.13}$	$0.44_{\pm 0.10}$	$0.55_{\pm 0.23}$	$0.46_{\pm 0.10}$	$0.37_{\pm 0.18}$	$0.25_{\pm 0.22}$	$0.26_{\pm 0.16}$
Telephone, Fax & Internet	$0.42_{\pm 0.53}$	$0.42_{\pm 0.40}$	$0.34_{\pm 0.38}$	$0.69_{\pm 0.20}$	$0.67_{\pm 0.05}$	$0.66_{\pm 0.11}$	$0.33_{\pm 0.24}$	$0.27_{\pm 0.22}$	$0.26_{\pm 0.16}$
Transport, Freight & Carriage	$0.00_{\pm 0.00}$	$0.00_{\pm 0.00}$	$0.00_{\pm 0.00}$	$0.35_{\pm 0.02}$	$0.40_{\pm 0.18}$	$0.36_{\pm 0.10}$	$0.21_{\pm 0.19}$	$0.12_{\pm 0.08}$	$0.14_{\pm 0.09}$

Table A6: Classification Report (precision, recall, F1) - DeBERTa mean. We report the mean±s.d. over 5 splits.

	Input A			**Input B**			**Input C**		
Class	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
Accountancy Fee	$0.46_{\pm 0.07}$	$1.00_{\pm 0.00}$	$0.63_{\pm 0.07}$	$0.34_{\pm 0.05}$	$0.99_{\pm 0.02}$	$0.51_{\pm 0.05}$	$0.37_{\pm 0.06}$	$0.92_{\pm 0.10}$	$0.52_{\pm 0.07}$
Cleaning of Premises	$0.00_{\pm 0.00}$	$0.00_{\pm 0.00}$	$0.00_{\pm 0.00}$	$0.74_{\pm 0.06}$	$0.36_{\pm 0.09}$	$0.48_{\pm 0.09}$	$0.85_{\pm 0.22}$	$0.29_{\pm 0.08}$	$0.43_{\pm 0.11}$
Directors' Loan Account	$0.22_{\pm 0.22}$	$0.04_{\pm 0.04}$	$0.07_{\pm 0.06}$	$0.23_{\pm 0.20}$	$0.08_{\pm 0.06}$	$0.11_{\pm 0.08}$	$0.07_{\pm 0.09}$	$0.03_{\pm 0.04}$	$0.04_{\pm 0.05}$
Light, Heat & Power	$0.11_{\pm 0.06}$	$0.18_{\pm 0.11}$	$0.12_{\pm 0.05}$	$0.55_{\pm 0.17}$	$0.47_{\pm 0.24}$	$0.50_{\pm 0.21}$	$0.43_{\pm 0.08}$	$0.58_{\pm 0.22}$	$0.49_{\pm 0.13}$
Motor Repairs and Servicing	$0.62_{\pm 0.07}$	$0.72_{\pm 0.08}$	$0.66_{\pm 0.03}$	$0.73_{\pm 0.02}$	$0.69_{\pm 0.07}$	$0.71_{\pm 0.04}$	$0.69_{\pm 0.05}$	$0.68_{\pm 0.07}$	$0.68_{\pm 0.03}$
Petrol and Oil	$0.36_{\pm 0.04}$	$0.46_{\pm 0.05}$	$0.40_{\pm 0.04}$	$0.65_{\pm 0.06}$	$0.77_{\pm 0.05}$	$0.70_{\pm 0.03}$	$0.72_{\pm 0.05}$	$0.74_{\pm 0.05}$	$0.73_{\pm 0.05}$
Purchases	$0.60_{\pm 0.06}$	$0.51_{\pm 0.08}$	$0.55_{\pm 0.04}$	$0.81_{\pm 0.04}$	$0.64_{\pm 0.05}$	$0.71_{\pm 0.04}$	$0.80_{\pm 0.02}$	$0.65_{\pm 0.05}$	$0.72_{\pm 0.03}$
Stationery & Postage	$0.80_{\pm 0.45}$	$0.20_{\pm 0.16}$	$0.31_{\pm 0.23}$	$0.78_{\pm 0.22}$	$0.33_{\pm 0.21}$	$0.43_{\pm 0.19}$	$0.56_{\pm 0.29}$	$0.20_{\pm 0.09}$	$0.29_{\pm 0.14}$
Sub	$0.74_{\pm 0.08}$	$0.43_{\pm 0.05}$	$0.54_{\pm 0.03}$	$0.85_{\pm 0.04}$	$0.73_{\pm 0.06}$	$0.78_{\pm 0.03}$	$0.56_{\pm 0.04}$	$0.76_{\pm 0.05}$	$0.64_{\pm 0.04}$
Subscriptions	$0.40_{\pm 0.11}$	$0.49_{\pm 0.10}$	$0.43_{\pm 0.06}$	$0.41_{\pm 0.09}$	$0.55_{\pm 0.06}$	$0.46_{\pm 0.08}$	$0.55_{\pm 0.09}$	$0.49_{\pm 0.09}$	$0.52_{\pm 0.08}$
Telephone, Fax & Internet	$0.58_{\pm 0.26}$	$0.53_{\pm 0.15}$	$0.52_{\pm 0.14}$	$0.67_{\pm 0.16}$	$0.67_{\pm 0.21}$	$0.66_{\pm 0.16}$	$0.63_{\pm 0.18}$	$0.64_{\pm 0.21}$	$0.63_{\pm 0.18}$
Transport, Freight & Carriage	$0.59_{\pm 0.14}$	$0.28_{\pm 0.08}$	$0.37_{\pm 0.09}$	$0.31_{\pm 0.16}$	$0.26_{\pm 0.12}$	$0.28_{\pm 0.14}$	$0.26_{\pm 0.10}$	$0.21_{\pm 0.09}$	$0.23_{\pm 0.09}$

	Precision	Recall	F1-score
Accountancy Fee	$0.22_{\pm 0.05}$	$0.23_{\pm 0.05}$	$0.23_{\pm 0.05}$
Cleaning of Premises	$0.68_{\pm 0.22}$	$0.33_{\pm 0.08}$	$0.43_{\pm 0.10}$
Directors' Loan Account	$0.00_{\pm 0.00}$	$0.00_{\pm 0.00}$	$0.00_{\pm 0.00}$
Light, Heat & Power	$0.29_{\pm 0.02}$	$1.00_{\pm 0.00}$	$0.45_{\pm 0.03}$
Motor Repairs and Servicing	$0.58_{\pm 0.08}$	$0.36_{\pm 0.05}$	$0.44_{\pm 0.06}$
Petrol and Oil	$0.93_{\pm 0.04}$	$0.62_{\pm 0.05}$	$0.75_{\pm 0.04}$
Purchases	$0.79_{\pm 0.01}$	$0.85_{\pm 0.03}$	$0.82_{\pm 0.02}$
Stationery & Postage	$0.67_{\pm 0.20}$	$0.29_{\pm 0.17}$	$0.37_{\pm 0.17}$
Sub	$0.47_{\pm 0.45}$	$0.03_{\pm 0.03}$	$0.05_{\pm 0.05}$
Subscriptions	$0.17_{\pm 0.02}$	$0.46_{\pm 0.09}$	$0.25_{\pm 0.04}$
Telephone, Fax & Internet	$0.82_{\pm 0.17}$	$0.35_{\pm 0.04}$	$0.48_{\pm 0.06}$
Transport, Freight & Carriage	$0.30_{\pm 0.02}$	$0.50_{\pm 0.07}$	$0.37_{\pm 0.03}$

Table A7: Classification report for the LLM baseline, mean±s.d. over 5 splits.

	Precision	Recall	F1-score
Accountancy Fee	$1.00_{\pm 0.00}$	$0.99_{\pm 0.02}$	$1.00_{\pm 0.01}$
Cleaning of Premises	$0.88_{\pm 0.16}$	$0.75_{\pm 0.08}$	$0.80_{\pm 0.09}$
Directors' Loan Account	$0.96_{\pm 0.09}$	$0.29_{\pm 0.12}$	$0.44_{\pm 0.15}$
Light, Heat & Power	$0.95_{\pm 0.05}$	$0.98_{\pm 0.04}$	$0.96_{\pm 0.02}$
Motor Repairs and Servicing	$0.99_{\pm 0.01}$	$0.92_{\pm 0.02}$	$0.95_{\pm 0.01}$
Petrol and Oil	$0.77_{\pm 0.04}$	$0.88_{\pm 0.03}$	$0.82_{\pm 0.03}$
Purchases	$0.82_{\pm 0.02}$	$0.93_{\pm 0.02}$	$0.87_{\pm 0.01}$
Stationery & Postage	$0.93_{\pm 0.06}$	$0.82_{\pm 0.17}$	$0.86_{\pm 0.10}$
Sub	$1.00_{\pm 0.00}$	$0.90_{\pm 0.04}$	$0.95_{\pm 0.02}$
Subscriptions	$0.98_{\pm 0.03}$	$0.91_{\pm 0.09}$	$0.94_{\pm 0.05}$
Telephone, Fax & Internet	$1.00_{\pm 0.00}$	$0.91_{\pm 0.09}$	$0.95_{\pm 0.05}$
Transport, Freight & Carriage	$1.00_{\pm 0.00}$	$0.99_{\pm 0.02}$	$0.99_{\pm 0.01}$

Table A8: Classification report for the vendor baseline. We report the mean±s.d. over 5 splits.

Model	Acc.	M F1	W F1
SBERT - 2 l	0.880	0.845	0.882
SBERT - 4 l	0.876	0.845	0.874
SBERT - 6 l	0.917	**0.875**	0.914
SBERT - 2 l + b.n.	0.903	0.839	0.904
SBERT - 4 l + b.n.	**0.931**	0.874	0.929
SBERT - 6 l + b.n.	0.929	0.873	**0.927**
DeBERTa - 2 l	0.846	0.774	0.842
DeBERTa - 4 l	0.664	0.471	0.651
DeBERTa - 6 l	0.433	0.309	0.411
DeBERTa - 2 l + b.n.	0.839	0.766	0.843
DeBERTa - 4 l + b.n.	0.406	0.108	0.302
DeBERTa - 6 l + b.n.	0.325	0.041	0.159
DeBERTa mean - 2 l	0.869	0.801	0.863
DeBERTa mean - 4 l	0.634	0.442	0.601
DeBERTa mean - 6 l	0.325	0.041	0.159
DeBERTa mean - 2 l + b.n.	0.922	0.845	0.915
DeBERTa mean - 4 l + b.n.	0.684	0.436	0.664
DeBERTa mean - 6 l + b.n.	0.442	0.143	0.396

Table A9: Accuracy, macro F1 and weighted F1 across fine-tuning configurations and number of unfrozen final layers (l) for input type A with and without business- nature prefix (+b.n.), for a single split. The best performance is in **bold**.

Model	Acc.	M F1	W F1
SBERT - 2 l	0.942	0.892	0.939
SBERT - 4 l	0.945	0.894	0.938
SBERT - 6 l	0.938	0.898	0.934
SBERT - 2 l + b.n.	0.940	0.895	0.937
SBERT - 4 l + b.n.	**0.959**	0.923	**0.956**
SBERT - 6 l + b.n.	**0.959**	**0.931**	**0.956**
DeBERTa - 2 l	0.878	0.776	0.872
DeBERTa - 4 l	0.535	0.247	0.494
DeBERTa - 6 l	0.325	0.041	0.159
DeBERTa - 2 l + b.n.	0.887	0.812	0.884
DeBERTa - 4 l + b.n.	0.537	0.181	0.448
DeBERTa - 6 l + b.n.	0.325	0.041	0.159
DeBERTa mean - 2 l	0.917	0.861	0.914
DeBERTa mean - 4 l	0.781	0.671	0.783
DeBERTa mean - 6 l	0.353	0.073	0.214
DeBERTa mean - 2 l + b.n.	0.933	0.886	0.930
DeBERTa mean - 4 l + b.n.	0.793	0.551	0.772
DeBERTa mean - 6 l + b.n.	0.385	0.155	0.261

Table A10: Accuracy, macro F1 and weighted F1 across fine-tuning configurations and number of unfrozen final layers (l) for input type C with and without business- nature prefix (+b.n.), for a single split. The best performance is in **bold**.

Table A11: Classification Report (precision, recall, F1) - SBERT fine-tuned with 2 unfrozen layers + business name prefix. We report the mean±s.d. over 5 splits.

	Precision	**Recall**	**F1**
Accountancy Fee	$1.00_{\pm 0.00}$	$0.99_{\pm 0.02}$	$1.00_{\pm 0.01}$
Cleaning of Premises	$0.83_{\pm 0.09}$	$0.91_{\pm 0.11}$	$0.86_{\pm 0.08}$
Directors' Loan Account	$0.86_{\pm 0.05}$	$0.61_{\pm 0.12}$	$0.71_{\pm 0.07}$
Light, Heat & Power	$0.92_{\pm 0.09}$	$0.98_{\pm 0.04}$	$0.95_{\pm 0.04}$
Motor Repairs and Servicing	$0.97_{\pm 0.02}$	$0.96_{\pm 0.01}$	$0.97_{\pm 0.01}$
Petrol and Oil	$0.96_{\pm 0.02}$	$0.97_{\pm 0.02}$	$0.96_{\pm 0.01}$
Purchases	$0.97_{\pm 0.00}$	$0.98_{\pm 0.01}$	$0.98_{\pm 0.01}$
Stationery & Postage	$0.94_{\pm 0.08}$	$0.98_{\pm 0.05}$	$0.96_{\pm 0.04}$
Sub	$0.96_{\pm 0.02}$	$0.98_{\pm 0.02}$	$0.97_{\pm 0.02}$
Subscriptions	$0.97_{\pm 0.04}$	$0.98_{\pm 0.02}$	$0.98_{\pm 0.02}$
Telephone, Fax & Internet	$0.95_{\pm 0.07}$	$1.00_{\pm 0.00}$	$0.97_{\pm 0.04}$
Transport, Freight & Carriage	$0.96_{\pm 0.07}$	$0.99_{\pm 0.02}$	$0.97_{\pm 0.05}$

Table A12: Classification Report (precision, recall, F1) - SBERT fine-tuned with 6 unfrozen layers. We report the mean±s.d. over 5 splits.

	Precision	**Recall**	**F1**
Accountancy Fee	$1.00_{\pm 0.00}$	$0.99_{\pm 0.02}$	$1.00_{\pm 0.01}$
Cleaning of Premises	$0.83_{\pm 0.13}$	$0.91_{\pm 0.11}$	$0.86_{\pm 0.08}$
Directors' Loan Account	$0.71_{\pm 0.11}$	$0.44_{\pm 0.14}$	$0.53_{\pm 0.08}$
Light, Heat & Power	$0.94_{\pm 0.07}$	$1.00_{\pm 0.00}$	$0.97_{\pm 0.04}$
Motor Repairs and Servicing	$0.99_{\pm 0.01}$	$0.95_{\pm 0.00}$	$0.97_{\pm 0.01}$
Petrol and Oil	$0.96_{\pm 0.01}$	$0.97_{\pm 0.02}$	$0.97_{\pm 0.01}$
Purchases	$0.94_{\pm 0.02}$	$0.97_{\pm 0.03}$	$0.95_{\pm 0.01}$
Stationery & Postage	$0.94_{\pm 0.08}$	$0.98_{\pm 0.05}$	$0.96_{\pm 0.04}$
Sub	$0.96_{\pm 0.01}$	$0.99_{\pm 0.01}$	$0.97_{\pm 0.01}$
Subscriptions	$0.98_{\pm 0.02}$	$0.96_{\pm 0.04}$	$0.97_{\pm 0.03}$
Telephone, Fax & Internet	$0.97_{\pm 0.05}$	$1.00_{\pm 0.00}$	$0.98_{\pm 0.02}$
Transport, Freight & Carriage	$0.96_{\pm 0.04}$	$0.99_{\pm 0.02}$	$0.97_{\pm 0.02}$

Table A13: Classification Report (precision, recall, F1) - DeBERTa fine-tuned with 2 unfrozen layers + business name prefix. We report the mean±s.d. over 5 splits.

	Precision	**Recall**	**F1**
Accountancy Fee	$0.98_{\pm 0.04}$	$0.99_{\pm 0.02}$	$0.99_{\pm 0.02}$
Cleaning of Premises	$0.98_{\pm 0.04}$	$0.99_{\pm 0.02}$	$0.99_{\pm 0.02}$
Directors' Loan Account	$0.77_{\pm 0.06}$	$0.64_{\pm 0.2}$	$0.68_{\pm 0.09}$
Light, Heat & Power	$1.00_{\pm 0.00}$	$0.98_{\pm 0.04}$	$0.99_{\pm 0.02}$
Motor Repairs and Servicing	$0.98_{\pm 0.01}$	$0.95_{\pm 0.02}$	$0.97_{\pm 0.01}$
Petrol and Oil	$0.98_{\pm 0.03}$	$0.97_{\pm 0.02}$	$0.98_{\pm 0.02}$
Purchases	$0.96_{\pm 0.01}$	$0.99_{\pm 0.01}$	$0.97_{\pm 0.01}$
Stationery & Postage	$0.86_{\pm 0.10}$	$0.80_{\pm 0.18}$	$0.82_{\pm 0.13}$
Sub	$0.96_{\pm 0.05}$	$0.95_{\pm 0.03}$	$0.95_{\pm 0.02}$
Subscriptions	$0.93_{\pm 0.04}$	$0.95_{\pm 0.05}$	$0.94_{\pm 0.02}$
Telephone, Fax & Internet	$0.96_{\pm 0.10}$	$0.96_{\pm 0.05}$	$0.96_{\pm 0.05}$
Transport, Freight & Carriage	$0.94_{\pm 0.08}$	$1.00_{\pm 0.00}$	$0.97_{\pm 0.05}$