Linking Industry Sectors and Financial Statements: A Hybrid Approach for Company Classification

Guy Stephane Waffo Dzuyo[1,2], **Gaël Guibon**[2,3], **Christophe Cerisara**[2], **Luis Belmar-Letelier**[1]
[1]Forvis Mazars
[2] LORIA, CNRS, Université de Lorraine
[3] Université Sorbonne Paris Nord, CNRS, Laboratoire d'Informatique de Paris Nord, LIPN, F-93430 Villetaneuse, France
guy.stephane.waffo@forvismazars.com, gael.guibon@lipn.fr,
christophe.cerisara@loria.fr, luis.belmar-letelier@forvismazars.com

Abstract

The identification of the financial characteristics of industry sectors has a large importance in accounting audit, allowing auditors to prioritize the most important area during audit. Existing company classification standards such as the Standard Industry Classification (SIC) code allow to map a company to a category based on its activity and products. In this paper, we explore the potential of machine learning algorithms and language models to analyze the relationship between those categories and companies' financial statements. We propose a supervised company classification methodology and analyze several types of representations for financial statements. Existing works address this task using solely numerical information in financial records. Our findings show that beyond numbers, textual information occurring in financial records can be leveraged by language models to match the performance of dedicated decision tree-based classifiers, while providing better explainability and more generic accounting representations. We think this work can serve as a preliminary work towards semi-automatic auditing.

Code — https://github.com/WaguyMz/company_classification

1 Introduction

Accounting is a key area of finance focused on recording, analyzing, and reporting a company's financial transactions. It provides critical insights into a company's financial health, supporting informed decision-making and strategic planning. An accounting audit involves a meticulous examination of a company's financial reports to verify their compliance with accounting norms and regulations. The auditing process is labor-intensive and time-consuming, often requiring significant resources and expertise. Recent advancements in Artificial Intelligence (AI) and automation have shown promise in transforming the auditing process. Prior to the emergence of neural networks, there was a considerable focus on Robotic Process Automation (RPA), aiming to automate manual tasks in accounting and auditing processes (Gotthardt et al. 2020). RPA is process-driven, as it mainly consists in the automation of rule-based tasks including data collection and invoicing.

To further enhance the efficiency and accuracy of auditing processes, the field has seen a shift from RPA to a more sophisticated family of approaches called Intelligent Process Automation (IPA) (Nunes, Leite, and Pedrosa 2020). IPA integrates AI technologies to handle more complex processes, including decision-making and analysis, mimicking human cognitive functions. Machine Learning offers powerful tools for identifying patterns and anomalies in financial data. These capabilities are crucial in auditing, where detecting irregularities early can prevent fraud and ensure compliance with financial regulations (Lokanan, Tran, and Vuong 2019). Machine Learning models can analyze historical financial data to predict potential risks (Cheng et al. 2023), prioritize areas of focus for auditors, and even suggest corrective actions based on past trends.

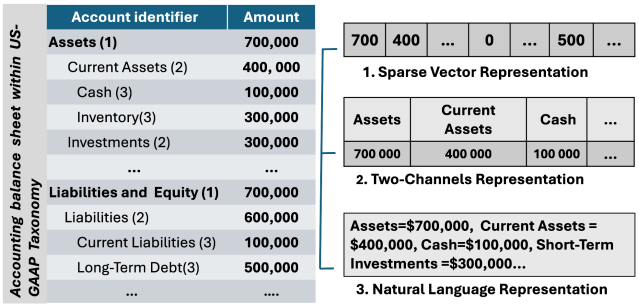

Figure 1: Balance sheet representations for Machine Learning Algorithms. Within US-GAAP taxonomy, each financial account is mapped to an identifier with taxonomy tree which help to organize financial information

Financial statements are a collection of financial reports that present the financial status of a company at a specific point in time. Balance sheets and income statements are two common types of financial statements, generally published at the end of a fiscal year. A balance sheet is a snapshot of a company's financial position at a specific point in time, detailing its assets, liabilities, and equity. The income statement, on the other hand, provides a summary of the company's financial performance over a period, highlighting its revenues, expenses, and profits or losses.

As visible in Figure 1's left part, a financial statement typically consists of a two-column table, where the first column contains the identifier of an account, and the second column displays the corresponding amount in a given cur-

rency. The identifier can be a bank account number or a short textual label describing the account's purpose. In the context of Machine Learning applications in finance, data representation plays a critical role. One approach is to disregard the label information and represent the statement as a sparse feature vector (Van Der Heijden 2022; Bakumenko and Elragal 2022). Alternatively, the label and the numeric amount can be combined into a unified textual representation, enabling the use of Language Models (Li and Vasarhelyi 2024). Another approach involves the usage of knowledge graphs (Hou 2022) which preserve the inherent hierarchical structure of financial data. We propose a novel method which combines embedding vectors of identifiers with scalars representing numerical amounts.

In this work, we use these various representations for the task of company classification, where companies are classified by their industry sector. These representations are displayed in Figure 1's right part. This task transforms financial patterns recognition into a supervised learning task. Decision tree-based methods and other algorithms have shown effective performance on this task by leveraging the inherent structure of financial statements to predict a company's classification code (Leen and Cely 2017; Van Der Heijden 2022). Through this task, we assess the effectiveness of different representations of financial reports and the underlying patterns they reveal.

Our main contributions can be summarized as follows:

1. We introduce an enhanced version of the company classification task, improving data composition and exploring different architectural approaches.
2. We demonstrate that a unified data representation across different channels yields the best results, highlighting the crucial role of data representation in tasks related to financial reports.
3. We leverage the capabilities of large language models to generate explanations for the decisions made by the classification model, showcasing its ability to reason over financial records.

Finally, we think this work can serve as a preliminary step towards semi-automatic auditing. An important task to help relieve the tedious efforts from human auditors.

2 Background

Company classification is the process of categorizing companies based on their business activities, products, and other relevant factors. Various standards (Hrazdil, Trottier, and Zhang 2013) are used to group companies with similar operating characteristics. These standards play a crucial role in research and regulatory activities. The **Global Industry Classification Standard (GICS)**, developed in 1999 by Standard & Poor's (S&P) and Morgan Stanley Capital International (MSCI), is one of the most used industry classification systems. It categorizes companies into sectors, industry groups, industries, and sub-industries based on their primary business activities. The **Standard Industrial Classification (SIC)**, developed in the 1930s to standardize industry definitions in the United States uses a four-digit code and is historically the first globally adopted classification standard. The **North American Industry Classification System (NAICS)** was introduced in 1997 to replace the SIC system. NAICS is used primarily in the United States by federal statistical agencies. Despite being largely replaced by NAICS, the SIC is still in use for historical data analysis and regulatory purposes. Notably, the public dataset used in our study relies on the SIC system.

3 Related Work

Numerical Approaches for Company Classification. Regarding the usage of numerical amounts for company classification, few notable prior works can be identified. Leen and Cely (2017) proposed a Neural Network-based approach to identify banks from their balance sheets. Using a 2-layer perceptron, that approach achieved an interesting performance on a dataset of 21 Colombian banks.

Van Der Heijden (2022) explored the effectiveness of linear discriminant analysis (LDA) and random forest classifier (Louppe 2015) for predicting companies' NAICS codes using 25-dimensional feature vectors, including 15 common size percentages and 12 financial ratios. Common size percentages (NerdWallet Staff 2023) normalize balance sheet accounts by total assets, allowing for comparison across companies, while financial ratios (Institute 2023) offer insights into financial performance. In their study, the random forest classifier achieved an F1-Score of 89%, outperforming LDA's 66%. However, a key limitation was that the train-test split did not account for company identities, risking the model learning company-specific patterns rather than industry sector patterns. Our methodology addresses this by ensuring that companies in the training, validation, and test sets are strictly disjoint.

Adaptation of Language Models to Finance. Language Models have shown remarkable performance across various domains, including specialized finance applications. Araci (2019) adapted the BERT encoder (Devlin et al. 2019) for financial sentiment analysis, resulting in FinBERT, which classifies the sentiment (positive, negative, or neutral) of stock-related news articles. Building on FinBERT, Loukas et al. (2022) developed SEC-BERT for Named Entity Recognition (NER), using a dataset of 1.1 million sentences from annual financial reports, containing 139 XBRL tags (Grosu et al. 2010).

In the accounting field, Vamvourellis et al. (2023) utilized embeddings from SEC 10-K filings (Staff 2023) to classify companies by GICS code. They fine-tuned a pre-trained Sentence BERT model (Reimers and Gurevych 2019) on a dataset of 3,000 publicly traded U.S. companies from 2022, achieving high accuracy in reproducing GICS sectors. Their approach also identified companies with similar financial profiles, showing higher return correlations compared to traditional GICS groupings, demonstrating the model's ability to capture financial similarities.

Zero-shot classification has been explored in accounting as well. Rizinski et al. (2023) introduced a method using the DistilBART model (Lewis et al. 2020) to classify companies using their business descriptions. It leverages natu-

ral language descriptions to achieve an accurate prediction of the GICS code. However, to our knowledge, no existing methods exclusively use financial records for this purpose. Our novel approach aims to leverage Language Models to classify companies based solely on financial records.

4 Financial Statements Representation for Company Classification

In this section, we explore various approaches to represent financial statements to address the company classification task. We focus on three primary methodologies namely sparse vector representation, multichannel sequence representation, and natural language representation.

Sparse Vector Representation. This representation assumes that account labels carry no semantic information. Based on this assumption, a financial statement is represented as a vector, where each dimension corresponds to a unique financial account in the dataset. The vector's length equals the total number of unique tags, with each position representing a specific account. The sparsity arises because the number of tags reported by a company in a statement is relatively small compared to the total number of unique tags.

Multichannel Sequence Representation. We propose to incorporate the semantic information from the account number in the model. The initial input financial statement is converted into an unordered sequence of N steps, N being the number of tags in the financial statement. Each step has 2 dimensions, the first being the text of an account identifier and the second being the corresponding amount. This representation avoids the curse of dimensionality and enables the model to capture an intermediate representation of the identifiers' meaning.

Natural Language Representation. The structure of a financial statement is converted into a sentence format using the template `Tag = Amount`. We propose two templates: (1) **Raw values**: Values are reported as raw signed numbers, rounded to one decimal place, with a currency symbol sign as prefix; (2) **Relative values**: Values are normalized by the total assets, typically the largest amount in the statement, and then converted into percentages, rounded to two decimal places. The use of relative values is inspired by the success of common size percentages in Van Der Heijden (2022).

The (tag, value) pairs are organized linearly in the text according to a predefined depth-first traversal of the taxonomy tree, ensuring that higher-level aggregate tags precede their subcategories. More details on data representation are provided in the appendix.

5 Methods for Company Classification

On top of each of the representations above, we propose an end-to-end supervised classification pipeline which aims to predict a company's sector, based on its quarterly financial statements. The company classification task is used here as a proxy to illustrate how machine learning models can capture and generalize common characteristics across groups of companies within specific industry sectors.

5.1 Decision-Tree Based Algorithm

We begin with a decision-tree algorithm, chosen for its ability to capture patterns in numerical financial data. Decision trees align well with the rule-based logic often employed in audits, making them a natural candidate for this task. Using a sparse vector representation, we apply a LightGBM classifier (Ke et al. 2017) to predict the sector classification (`SIC`) of the company. We investigate three representation variants for the amounts:

- **Raw Amount Representation**: Directly uses signed amounts from financial statements.
- **Relative Values Representation**: Represents each amount relative to the total `Assets`, reducing bias from varying company sizes (NerdWallet Staff 2023).
- **Binary Representation**: Uses 1 for the presence and 0 for the absence of a financial statement tag, disregarding actual amounts to assess the impact of account presence on sector classification.

5.2 LLM-clf: LLM With Classification Head

LLM-clf pipeline uses a standard supervised classification approach with a pretrained LLM followed by a linear layer and a softmax activation for classification. The input to this classifier is the latent representation from the [EOS] token, which encapsulates the entire input sequence. This method leverages the LLM's natural language understanding to transform company classification into a supervised text classification task.

5.3 LLM-gen: LLM With Generation Head

LLM-gen is a classification pipeline that uses the generative abilities of LLMs to predict a company's industry sector through autoregressive sentence completion. Financial statements are converted into a natural language sentence, augmented with an instruction prompt detailing the task and providing candidate labels. The LLM is fine-tuned under a Causal Language Modelling objective to maximize the conditional probability of tokens. Instead of requiring exact label matching, we check whether any of the eight candidate labels appears in the generated text.

5.4 Text-Numeric Transformer

We propose a two-stage architecture to leverage both textual representations of accounts' identifiers and corresponding amounts. The input is a sequence of N (identifier, amount) pairs (t_i, x_i). The first stage of this architecture is a pretrained Sentence-BERT (Reimers and Gurevych 2019) encoder τ_0 that computes an embedding representation $\hat{e_i}$ of each identifier t_i. Then, a **fusion module** combines $\hat{e_i}$ with amount t_i into a single vector c_1. This is done by multiplying $\hat{e_i}$ by $sign(x_i) \log |x_i|$, where x_i is the account amount. Through this transformation, we scale embeddings to reflect relative importance of each account, enhancing the model's ability to prioritize information effectively. The logarithm keeps the scaling factor small for numerical stability. The embedding's scaling is followed by a linear layer and GeLU activation, producing a new embedding vector c_i, that we call the **fused embedding**.

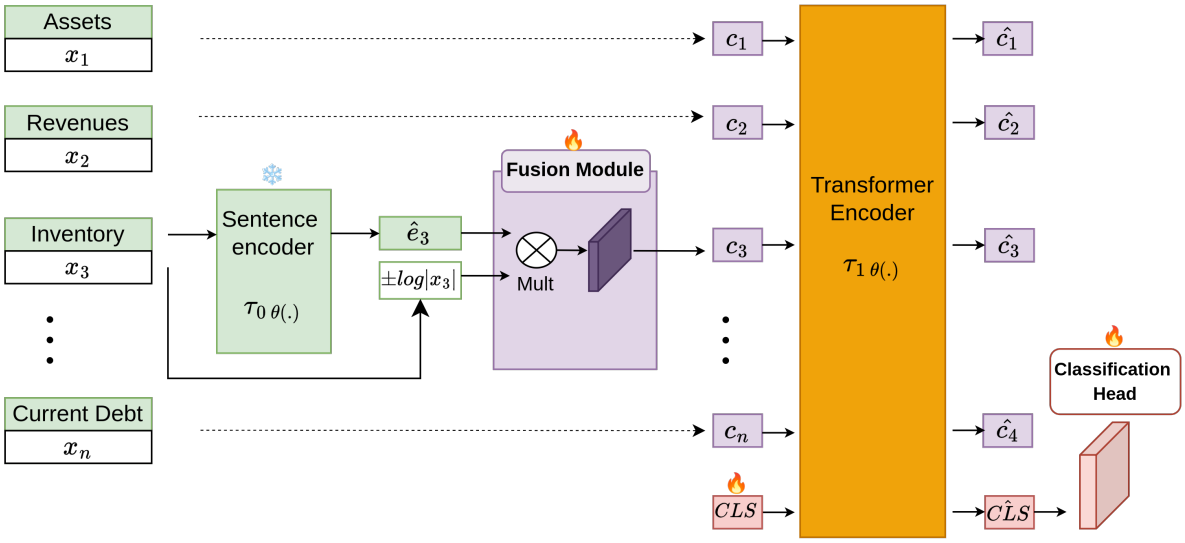

Figure 2: Text-Numeric network. Given a sequence of n (tags, amounts) pairs (t_i, x_i). A sentence encoder computes an embedding $\mathbf{e_i}$ of each identifier $\mathbf{t_i}$. A fusion module produces a fused embedding $\mathbf{c_i}$ by multiplying $\mathbf{e_i}$ with $\log|x_i|$. A CLS vector is mapped to a final classification head

The network's next component is an **Encoder Transformer** τ_1, which processes the sequence of fused embedding vectors c_i. Like BERT (Devlin et al. 2019), this sequence is prepended with a learnable $[CLS]$ vector, initialized from a gaussian distribution. The transformer's attention mechanism allows the output vector $[C\hat{L}S]$ to capture information from all fused embeddings. Finally $[C\hat{L}S]$ is fed to a single linear layer followed by a softmax activation function which outputs the log-probabilities of the target classes. Figure 2 shows an overview of the network.

6 Dataset

We leverage publicly available U.S. company filings from the SEC's EDGAR database (U.S. Securities and Exchange Commission 2024), which includes quarterly financial statements from U.S. companies spanning from January 2008 to March 2024. That dataset provides detailed financial information, such as balance sheets, income statements, cash flow statements, and statements of equity, all reported according to the US-GAAP taxonomy(Van Der Meulen, Gaeremynck, and Willekens 2007). US-GAAP categorizes financial data into sections such as assets, liabilities, equity, income, and cash flow. Each category within the taxonomy is represented by a unique identifier called a **tag**.

The dataset comprises **358,460** quarterly submissions from **19,329** companies, with industry sectors classified by 4-digit SIC codes. From the initial dataset, we extract a subset consisting of data from the fiscal year 2023 only. This subset includes **26,907** submissions from **7,013** companies. The rationale behind focusing on this period is to prevent our model from learning seasonal patterns rather than industry sector patterns. For instance, specific periods like the COVID-19 pandemic from 2019 to 2020 could induce certain patterns in company financial reports.

We apply a filtering process followed by data imputation stage. The filtering process includes the following steps:

1. We apply a cutoff of **100 samples per industry sector**. As a result, submissions from companies in the **Agriculture** sector are excluded because the sector contains only **78 samples**.
2. In each submission, any *tag* that does not come from the balance sheet or income statement is removed.
3. Submissions containing less than 30 balance sheet tags are discarded. We set this threshold to 30, which corresponds to the 90th percentile of the number of tags per submission distribution.

Data imputation involves calculating the values of missing tags by leveraging the tree structure of the US GAAP taxonomy. In average, this process adds **28** new tags per submission. Finally, we filter tags based on their depth in the US GAAP taxonomy, retaining only those within the first five levels. This step reduces bias from highly specific, industry-specific tags, ensuring the data remains general and comparable across sectors. After all these steps, we obtain a dataset of **9,582** samples including **2,861** companies. The target labels used for the classification are 8 official industry sectors derived from the first two digits of the SIC Code (siccode.com 2024). Figure 3 shows the industry sectors and their distribution in the final dataset. Appendix provides more details on the dataset and the preprocessing pipeline. Each sample consists of 2-columns as presented in Figure 1. The tags have a size varying between 4 and 169 characters.

7 Experiments and Results

Considering the representations and methods presented, we conduct different experiments to verify their quality and impact on the proxy task of company classification.

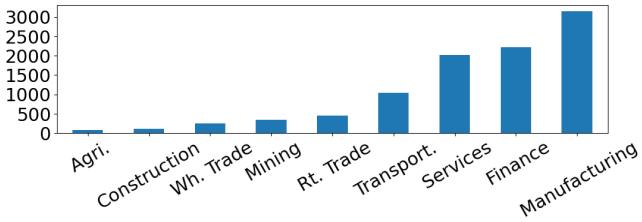

Figure 3: Distribution of industry sectors in the SEC financial statements dataset

7.1 Dataset Splitting

The dataset includes submissions from the four quarters of the year 2023, with each company contributing an average of four submissions. To avoid bias from unique company-specific patterns, we propose a more robust dataset splitting strategy that ensures that the companies represented in the training, validation, and test sets are distinct. This is different from the random splitting proposed by Van Der Heijden (2022). We partition the dataset based on identities of the companies. The training set includes submissions from companies that make up **70%** of the total dataset. The validation set comprises submissions from other companies that account for **10%** of the dataset, while the test set includes submissions from the remaining **20%**. This approach ensures that no company appears in more than one subset, enabling evaluation of generalization to unknown companies.

7.2 High Scores With LightGBM

We use the LightGBM classifier from the official library (Ke et al. 2017), with key hyper-parameters **number of estimators=100**, **maximum depth=600**, and **number of leaves=60** tuned using the Hyperopt framework and Bayesian optimization (Kegl 2023).

We conduct experiments using the three representation variants described in Section 5.1: raw, percentage, and binary. Three key metrics are measured: A weighted **F1 score**, **Pearson-Matthews correlation (MCC)** (Stoica and Babu 2023), and **Mean Reciprocal Rank (MRR)** (Craswell 2009). MCC is particularly robust in imbalanced settings, while MRR provides insights into how well the misclassified samples are ranked relative to the ground truth. Table 1 presents the performance of the LightGBM classifier on these metrics. The best performance is obtained when using the relative values representation, which leads to a MCC of 0.62 on the validation set and 0.69 on the test set. The raw values yield a lower performance, highlighting the importance of normalizing financial data to account for company size variations. Unsurprisingly, the binary representation performs the worst with a MCC of 0.56 on the test but this performance still indicates a high contribution of the presence of specific accounts in the characterization of industry sectors.

LightGBM Predictions Explanation. The LightGBM model generates up to 800 decision trees, making individual predictions difficult to interpret. Instead, we calculate global feature importance scores based on the total number of splits each feature contributes to. These scores identify `Inventory Net of Allowances Customer Advances and Progress Billings` as the most important tag, representing the company's net inventory value after deductions. See Appendix 3 for details on LightGBM's feature importance.

Data	Representation	Dim	MCC	F1	MRR
Val.	Raw	368	0.57	0.67	0.79
	Rel	368	**0.62**	**0.70**	**0.81**
	Binary	368	0.51	0.58	0.74
Test	Raw	368	0.66	0.74	0.83
	Rel	368	**0.69**	**0.76**	**0.85**
	Binary	368	0.56	0.64	0.78

Table 1: Performance of LightGBM classifier on company classification. Rel= Relative Values . Val= Validation set.

7.3 Features From LLM-clf Underperform

For this approach, we utilize LLaMA3.1 8B (Touvron et al. 2023a). Instead of fine-tuning the entire model, we employ LoRA adapter (Hu et al. 2021) to train a small number of added parameters. We also conduct experiments using Finance-LLaMA3 (Cheng et al. 2024), a fine-tuned version of LLaMA 3 on a Financial Corpora (FinLLaMA3 8B). The experimental setup of this model is detailed in the appendix. Each model is trained for 15 epochs using AdamW optimizer (Loshchilov and Hutter 2017) with an initial learning rate of $1e{-}4$ decayed using a `ReduceLrOnPlateau` (Defazio et al. 2023) strategy.

Data	Rep.	Model	MCC	F1	MRR
Val	Rel.	**LLaMA3.1**	**0.59**	**0.67**	**0.80**
	Rel.	FinLLaMA3	0.59	0.67	0.79
	Raw	LLaMA3.1	0.53	0.63	0.76
	No.A	LLaMA3.1	0.02	0.12	0.53
Test	Rel.	LLaMA3.1	0.64	0.70	0.82
	Rel.	**FinLLaMA3**	**0.64**	**0.71**	**0.82**
	Raw	LLaMA3.1	0.58	0.67	0.79
	No.A	LLaMA3.1	0.06	0.15	0.55

Table 2: Performance of LLM-clf in company classification. Rel. corresponds to a relative values representation, No.A is representation where accounts amounts are not mentioned. All models are 8B versions. Best results are in bold face

Both FinLLaMA3 and LLaMA3 achieve similar MCC scores of 0.64 on the test set using relative value representation. The lower performance observed with raw values is likely due to the LLM's limitations in handling large numbers. Future work could explore improved numerical representation techniques, such as those proposed Schwartz et al. (2024) and Yang et al. (2022) where special tokens are introduced to represent the structure of numbers. We notice that the usage of only account tags (No.A) without amounts leads to significantly poor performance with LLM-clf, emphasizing the importance of account amounts in the characterization of industry sectors.

7.4 LLM-gen And Explainable Predictions

We again utilize LLaMA3.1 8B (Touvron et al. 2023a), but instead of the pretrained version of that model, we use LLaMA3.1 8B-instruct, a fine-tuned version of LLaMA3.1 8B using instruction tuning (Zhang et al. 2024). We also use FinLlaMA3, which was already fine-tuned using a corpus of instructions (Cheng et al. 2024). Once again, we apply LoRA adaptation for training, following the same configuration as LLM-clf. Next-token prediction finetuning is applied only to the completion part of the prompt because the model is not expected to generate financial statements. The prompt template used for LLM-gen is provided in the appendix.

Table 3 presents the results of LLM-gen. With relative values representation, LLaMa3.1 8B-Instruct and FinLLaMA3 8B achieve comparable performances on the validation set with a MCC score of 0.59. On the test set, FinLLaMA3 8B performs slightly better than LlaMA3.1 8B-Instruct with a MCC score of $0.66 > 0.64$.

LlaMA3.1 8B is also trained using raw values. It achieves an MCC of 0.60 on validation and 0.65 on the test set. While other approaches, notably LightGBM and LLM-clf, struggle with raw values representation, LLM-gen performs comparably well whether using raw or relative values representation. We hypothesize that the straightforward classification pipeline in LLM-gen allows the LLM to fully exploit its reasoning capabilities, including comparative analysis of accounts. In contrast, LLM-clf relies on an intermediary representation (the EOS token encoding), which may constrain its reasoning abilities and put greater emphasis on representational capacity rather than reasoning through the data. When trained without any amount, the MCC on validation and test sets are lowers, at 0.56 and 0.58, respectively.

Data	Rep.	Model	MCC	F1
Val	Rel.	LLaMA3.1	0.59	0.67
	Rel.	FinLLaMA3	0.59	0.67
	Raw	**LLaMA3.1**	**0.60**	**0.68**
	No.A	LLaMA3.1	0.56	0.62
Test	Rel.	LLaMA3.1	0.64	0.70
	Rel.	**FinLLaMA3**	**0.66**	**0.71**
	Raw	LLaMA3.1	0.65	0.70
	No.A	LLaMA3.1	0.58	0.64
Zero shot	Rel.	LLaMA3.1	0.30	0.42
	Rel.	FinLLaMA3	0.29	0.40

Table 3: Performance of the **LLM-gen** in company classification. Rel. corresponds to a relative values representation, No.A is representation where accounts amounts are not mentioned. Best results are in bold face.

Zero-Shot Classification. We evaluate the intrinsic ability of pretrained LLMs on the test set to classify companies based solely on financial statements, without any task-specific fine-tuning. The performance is notably weak, with LLaMA3.1 8B and FinLLaMA3 8B achieving MCC scores of only 0.30 and 0.29, respectively, highlighting the challenges faced when applying LLMs to highly specialized tasks without task-specific training. The model's predictions are stuck between two classes : **Manufacturing** (57% of predicted labels) and **Services** (25% of predicted labels).

Explanations of LLM-gen's predictions. With LLM-gen, we benefit from the LLM's generation capabilities to obtain predictions explanations in natural language. Conversely, the usage of a classification head in LLM-clf annihilates its generation capabilities. Considering the entire test set (2,596 samples) and the output labels predicted by the LLM, we prompt it again to generate an explanation of the predicted answer. The prompt template used for generating the explanation is included in the appendix.

An example explanation provided by the LLM for a correct prediction on the test set is:

> ***Predicted label****: Transportation & Public Utilities*
> ***Generated explanation****: The presence of Asset Retirement Obligations Noncurrent (18.41% of total assets) and Regulatory Assets Current and Noncurrent (totaling 30.23% of total assets) suggests that the company is in the utility industry, likely in the electric, gas, or water sector.*

We conduct a qualitative evaluation of 10 generated explanations (See Appendix 5), and we found them coherent; for instance, the regulatory assets mentioned in this example are indeed often related to public utilities activities [1].

Low Impact of Ordering in Natural Language Representation. With both LLM-gen and LLM-clf, we apply a depth-first traversal of the US-GAAP structure to build the input sequence of the LLM. We conducted additional experiments using a breadth-first traversal strategy in the representation and obtained similar results. This finding suggests that the specific ordering of account tags, has a minimal impact on model performance.

7.5 Text-Numeric Transformer Yields the Best Performance

We explore two sentence encoders in the Text-Numeric network:

- **BGE Base** (Xiao et al. 2023): A state-of-the-art 768-dimensional model trained on a massive text dataset including Wikipedia pages, StackExchange, and Reddit.
- **FinLang** (Investopedia 2024): A version of BGE Base fine-tuned on a financial corpus (FinLang 2023).

We keep all the parameters of the sentence encoders frozen. The fusion modules' input layer has 768 dimensions, matching the sentence encoder's embedding dimensions, and its output layer has 64 dimensions, corresponding to the embedding dimension of the transformer τ_1. Positional encoding is not applied on the input sequence of τ_1 as the order of the tags in the initial sequence does not account.

τ_1 consists of 4 transformer encoder layers with the feed-forward layer in each transformer block having 256 hidden dimensions.

[1] see https://www.investopedia.com/terms/r/regulatoryasset.asp

We conduct experiments with batch size of 128 using the AdamW optimizer (Loshchilov and Hutter 2017) with an initial learning rate of 1e-4 decayed using a `ReduceLrOnPlateau` (Defazio et al. 2023) strategy. Each variant is trained for 60 epochs.

Data	Model	MCC	F1	MRR
Val	BGE Base	**0.63**	**0.70**	**0.82**
	FinLang	**0.63**	**0.70**	**0.82**
	FinLang(No Amounts)	0.56	0.62	0.78
Test	BGE Base	0.68	0.73	0.84
	FinLang	**0.71**	**0.76**	**0.86**
	FinLang(No Amounts)	0.60	0.65	0.80

Table 4: Performance of Text-Numeric in company classification. Comparison between BGE Base and FinLang models on validation and test sets. No Amounts is an experiment without including amount. Val = Validation set.

The results of experiments with the Text-Numeric model on the Test and validation sets are presented in Table 4.

The FinLang-base network provides the best results with a MCC score of **0.71** on the test set. However, we note that FinLang and BGE performs comparably on the validation set. Therefore, it is hard to conclude on the superiority of FinLang over BGE. We conduct an additional experiment without amounts, by replacing all input financial statement values with 10 ($\log_{10}(10) = 1$), effectively turning the multiplication gate of the fusion module into an identity function for the text embeddings. The performance is low, demonstrating the interest of merging account amounts with the semantic representation of accounts. Although its performance are low, this pure-text model outperforms the LightGBM binary representation without amounts, highlighting the benefit of using embeddings for tags' representation.

Interpretability of Text-Numeric Predictions We analyze the contribution of related accounts to industry sector predictions using the local relevance score method from Chefer, Gur, and Wolf (2021). This method calculates relative scores for each transformer layer via Deep Taylor Decomposition and propagates them to assign relevance scores to input tokens. We compute relevance scores for the fused embedding vectors c_i to directly assess the relevance of paired tags and amounts. Figure 4 shows relevance scores for a single prediction. We qualitatively evaluate 10 samples where the network correctly predicts the target label. Appendix provides additional details on this analysis.

8 Limitations

This study investigates methods to effectively capture industry-specific patterns in financial statements using numerical and language model-based approaches. While promising results are achieved in explainability and classification accuracy, several limitations remain.

Imbalanced Dataset: the dataset's imbalance posed challenges for minority classes, and attempts to oversample minority classes reduced overall performance. Future work will explore techniques like GANs and Diffusion Models for better handling class imbalance.

Assets Current	$37,318,996
Assets Noncurrent	$110,419,072
Common Stock Value	$-420,000
Construction In Progress Gross	$3,720,002
Debt Current	$-1,278,000
Deferred Income Tax Liabilities Net	$-605,000
Finite Lived Intangible Assets Net	$69,000
Income Loss From Continuing Operations Including Portion Attributable To Noncontrolling Interest	$-6,271,002
Indefinite Lived Intangible Assets Excluding Goodwill	$1,267,999
Intangible Assets Net Excluding Goodwill	$1,336,999
Intangible Assets Net Including Goodwill	$1,336,999

Figure 4: Heatmap of relevance scores for a Text-Numeric's prediction. Darker rows indicate higher relevance.

Evaluation of LLM-gen's Explanations: Our analysis indicates that the LLM-gen explanations are generally coherent. However, due to the lack of a grounded reference for the specific characteristics of industry sectors, we were unable to conduct a quantitative evaluation. In future studies, we plan to develop a dataset containing a set of sector-specific rules to enable a robust and systematic quantitative assessment of those explanations.

Computations capabilities for larger LLMs: due to GPU limitations, we did not conduct any experiment using larger LLMs such as LLaMA 405B (Touvron et al. 2023b) for LLM-gen and Voyage-Finance Embedding (AI 2024) for Text-Numeric Network. We hypothesize that their more advanced capabilities could lead to a better understanding of patterns in financial statements.

9 Conclusion and Perspectives

This study explores the application of machine learning techniques in accounting audit, focusing on company classification as a proxy task. The results highlight the capability of machine learning methods to learn industry-specific patterns within financial reports. We investigated various data representations, including sparse vectors, textual sequences, hybrid models, and introduced a novel Text-Numeric Transformer that achieved the highest classification performance.

Our experiments demonstrate that decision-tree-based methods like LightGBM establish solid baselines. However, more sophisticated techniques using language models provide not only strong performance but also enhanced explainability. In particular, the LLM-gen approach, leveraging generative capabilities, delivers explanations that can serve as valuable insights for auditors.

Our proposed Text-Numeric architecture effectively integrates numerical data with semantic representations of account identifiers, leading to superior classification accuracy. Future work will extend this approach to other accounting tasks, such as anomaly detection and risk scoring. We also aim to enhance explainability and further refine data representations, particularly for more complex and fine-grained company classification. Ultimately, our research contributes to the development of semi-automated auditing systems that closely mimic human reasoning in financial analysis.

Ethical Statement

It is crucial to acknowledge the ethical implications associated with using AI systems and especially Large Language Models (LLMs) in auditing. These implications include the question of legal responsibility for potential errors induced by the LLM, as well as inherent biases in the model's foundational knowledge. We firmly believe that, despite their remarkable capabilities, LLMs should never replace human auditors. Instead, they should serve as assistants, supporting auditors in exhausting and time-consuming tasks.

Acknowledgments

Experiments presented in this paper were carried out using the Grid'5000 testbed, supported by a scientific interest group hosted by Inria and including CNRS. RENATER and several Universities as well as other organizations (see https://www.grid5000.fr).

References

AI, V. 2024. Domain-Specific Embeddings: Finance Edition (Voyage Finance 2). https://blog.voyageai.com/2024/06/03/domain-specific-embeddings-finance-edition-voyage-finance-2/. Accessed: August 15, 2024.

Araci, D. 2019. FinBERT: Financial Sentiment Analysis with Pre-trained Language Models. arXiv:1908.10063.

Bakumenko, A.; and Elragal, A. 2022. Detecting Anomalies in Financial Data Using Machine Learning Algorithms. 10(5): 130.

Chefer, H.; Gur, S.; and Wolf, L. 2021. Transformer Interpretability Beyond Attention Visualization. arXiv:2012.09838.

Cheng, D.; Gu, Y.; Huang, S.; Bi, J.; Huang, M.; and Wei, F. 2024. Instruction Pre-Training: Language Models are Supervised Multitask Learners. arXiv:2406.14491.

Cheng, D.; Niu, Z.; Zhang, J.; Zhang, Y.; and Jiang, C. 2023. Critical firms prediction for stemming contagion risk in networked-loans through graph-based deep reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 14205–14213.

Craswell, N. 2009. *Mean Reciprocal Rank*, 1703–1703. Boston, MA: Springer US. ISBN 978-0-387-39940-9.

Defazio, A.; Cutkosky, A.; Mehta, H.; and Mishchenko, K. 2023. When, Why and How Much ? Adaptive Learning Rate Scheduling by Refinement. arXiv:2310.07831.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Burstein, J.; Doran, C.; and Solorio, T., eds., *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)*, 4171–4186. Minneapolis, Minnesota: Association for Computational Linguistics.

FinLang. 2023. Investopedia Embedding Dataset. https://huggingface.co/datasets/FinLang/investopedia-embedding-dataset. Accessed: 2024-07-26.

Gotthardt, M.; Koivulaakso, D.; Paksoy, O.; Saramo, C.; Martikainen, M.; and Lehner, O. 2020. Current State and Challenges in the Implementation of Smart Robotic Process Automation in Accounting and Auditing. 9(1): 90–102.

Grosu, V.; Hlaciuc, E.; Iancu, E.; Petris, R.; and Socoliuc, M. 2010. The Role of the XBRL Standard in Optimizing the Financial Reporting. arXiv:1002.3997.

Hou, X. 2022. Design and Application of Intelligent Financial Accounting Model Based on Knowledge Graph. 2022: 1–9.

Hrazdil, K.; Trottier, K.; and Zhang, R. 2013. A comparison of industry classification schemes: A large sample study. 118(1): 77–80.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; and Chen, W. 2021. LoRA: Low-Rank Adaptation of Large Language Models. *CoRR*, abs/2106.09685.

Institute, C. F. 2023. Financial Ratios. https://corporatefinanceinstitute.com/resources/accounting/financial-ratios/. Accessed: 2024-08-13.

Investopedia. 2024. Investopedia Embedding for Finance Applications. https://huggingface.co/FinLang/finance-embeddings-investopedia. Accessed: 2024-07-26.

Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; and Liu, T.-Y. 2017. LightGBM: A Highly Efficient Gradient Boosting Decision Tree.

Kegl, B. 2023. A systematic study comparing hyperparameter optimization engines on tabular data. arXiv:2311.15854.

Leen, C.; and Cely, J. 2017. Whose Balance Sheet Is This ? Neural Networks for Banks' Pattern Recognition.

Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; and Zettlemoyer, L. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In Jurafsky, D.; Chai, J.; Schluter, N.; and Tetreault, J., eds., *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7871–7880. Online: Association for Computational Linguistics.

Li, H.; and Vasarhelyi, M. A. 2024. Applying Large Language Models in Accounting: A Comparative Analysis of Different Methodologies and Off-the-Shelf Examples.

Lokanan, M.; Tran, V.; and Vuong, N. H. 2019. Detecting anomalies in financial statements using machine learning algorithm: The case of Vietnamese listed firms. 4(2): 181–201.

Loshchilov, I.; and Hutter, F. 2017. Fixing Weight Decay Regularization in Adam. *CoRR*, abs/1711.05101.

Loukas, L.; Fergadiotis, M.; Chalkidis, I.; Spyropoulou, E.; Malakasiotis, P.; Androutsopoulos, I.; and Paliouras, G. 2022. FiNER: Financial Numeric Entity Recognition for XBRL Tagging. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics.

Louppe, G. 2015. Understanding Random Forests: From Theory to Practice. arXiv:1407.7502.

NerdWallet Staff. 2023. Common Size Analysis of Financial Statements. https://www.nerdwallet.com/article/small-business/common-size-analysis. Accessed: 2024-08-11.

Nunes, T.; Leite, J.; and Pedrosa, I. 2020. Intelligent Process Automation: An Overview over the Future of Auditing. In *2020 15th Iberian Conference on Information Systems and Technologies (CISTI)*, 1–5. IEEE. ISBN 978-989-54-6590-3.

Reimers, N.; and Gurevych, I. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In Inui, K.; Jiang, J.; Ng, V.; and Wan, X., eds., *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 3982–3992. Hong Kong, China: Association for Computational Linguistics.

Rizinski, M.; Jankov, A.; Sankaradas, V.; Pinsky, E.; Miskovski, I.; and Trajanov, D. 2023. Company classification using zero-shot learning. arXiv:2305.01028:2305.01028 [cs].

Schwartz, E.; Choshen, L.; Shtok, J.; Doveh, S.; Karlinsky, L.; and Arbelle, A. 2024. NumeroLogic: Number Encoding for Enhanced LLMs' Numerical Reasoning. arXiv:2404.00459:2404.00459 [cs].

siccode.com. 2024. Standard Industrial Classification (SIC) Code Lookup. https://siccode.com/. Accessed: 2024-08-15.

Staff, I. 2023. Annual Report vs. 10-K: What's the Difference ? https://www.investopedia.com/ask/answers/102714/what-are-differences-between-10k-report-and-firms-own-annual-report.asp. Accessed: 2024-08-11.

Stoica, P.; and Babu, P. 2023. Pearson-Matthews correlation coefficients for binary and multinary classification and hypothesis testing. arXiv:2305.05974.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; Rodriguez, A.; Joulin, A.; Grave, E.; and Lample, G. 2023a. LLaMA: Open and Efficient Foundation Language Models. arXiv:2302.13971.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; Rodriguez, A.; Joulin, A.; Grave, E.; and Lample, G. 2023b. LLaMA: Open and Efficient Foundation Language Models. arXiv:2302.13971.

U.S. Securities and Exchange Commission. 2024. Financial Statement Data Sets. https://www.sec.gov/dera/data/financial-statement-data-sets. Accessed: 2024-08-14.

Vamvourellis, D.; Toth, M.; Bhagat, S.; Desai, D.; Mehta, D.; and Pasquali, S. 2023. Company Similarity using Large Language Models. arXiv:2308.08031:2308.08031 [q-fin, stat].

Van Der Heijden, H. 2022. Predicting industry sectors from financial statements: An illustration of machine learning in accounting research. 54(5): 101096.

Van Der Meulen, S.; Gaeremynck, A.; and Willekens, M. 2007. Attribute differences between U.S. GAAP and IFRS earnings: An exploratory study. 42(2): 123–142.

Xiao, S.; Liu, Z.; Zhang, P.; and Muennighoff, N. 2023. C-Pack: Packaged Resources To Advance General Chinese Embedding. arXiv:2309.07597.

Yang, L.; Li, J.; Dong, R.; Zhang, Y.; and Smyth, B. 2022. NumHTML: Numeric-Oriented Hierarchical Transformer Model for Multi-Task Financial Forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10): 11604–11612.

Zhang, S.; Dong, L.; Li, X.; Zhang, S.; Sun, X.; Wang, S.; Li, J.; Hu, R.; Zhang, T.; Wu, F.; and Wang, G. 2024. Instruction Tuning for Large Language Models: A Survey. arXiv:2308.10792:2308.10792 [cs].