IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 29 January 2024, accepted 22 April 2024, date of publication 25 April 2024, date of current version 3 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3393830

APPLIED RESEARCH

Automated Assessment of Capital Allowances

JAVAD GHOLIZADEH[1,2], KWANG-SUNG CHUN[1], CLIVE CURD[1], NOLAN MASTERS[1], DAVID GIBSON[1], AND YONGMIN LI[2], (Senior Member, IEEE)

[1]Veritas Advisory Ltd., WC1A 2SL London, U.K.
[2]Department of Computer Science, Brunel University London, UB8 3PH Uxbridge, U.K.

Corresponding author: Yongmin Li (yongmin.li@brunel.ac.uk)

This work was supported by the Innovate U.K. Knowledge Transfer Partnership Program under Agreement 94121.

ABSTRACT Capital allowances play a crucial role in enabling businesses to claim tax relief on specific capital expenditures, reducing their taxable profits and overall tax burden. However, the current manual process for managing capital allowance claims is time-consuming and complex, particularly for small and medium enterprises (SMEs) that often lack access to expert consultation. Furthermore, the distinct nature of construction expenditure on buildings adds to the complexity, with unique costs and data for each property and project. These challenges underscore the necessity for the development of automated technologies and systems for capital allowance assessment. To address these challenges, we present the development of an automated capital allowance assessment system comprising three key components: a capital allowance expert system, a tax coding system, and an integrated web-based application. The capital allowance expert system covers the entire process of capital allowance assessment, leveraging rules and procedures extracted from standardised processes and expertise. The tax coding system automatically classifies textual costing items into corresponding tax codes, addressing the complexity of capital allowance rules and frequent legislative changes. The integrated web-based application offers an interactive experience for data gathering, analysis, coding, and report generation, providing a comprehensive solution for efficient and accurate capital allowance assessment. This automated system addresses the complexities and inefficiencies associated with manual capital allowance assessment. It potentially benefits tax authorities in standardising and streamlining allowance assessment processes while fostering economic growth through accessible services for SMEs and promoting environmental sustainability by encouraging energy-efficient practices.

INDEX TERMS Capital allowance, tax relief, expert systems, artificial intelligence, natural language processing, automation.

I. INTRODUCTION

Capital allowances refer to the tax relief that businesses can claim on certain types of capital expenditure, such as equipment, machinery, or business vehicles [10]. These allowances enable a business to deduct the cost of these assets from their taxable profits, reducing the amount of tax they need to pay [6]. Capital allowances serve as a mechanism through which businesses can obtain tax relief to account for the gradual deterioration or depreciation of these assets over time [26].

A typical process for a capital expenditure assessment in the UK, according to [9], can be described as follows.

The associate editor coordinating the review of this manuscript and approving it for publication was Shadi Alawneh.

Information gathered from client contains two main sets of information: capital expenditure ledger and contractors' final account. A capital expenditure ledger is a record-keeping tool used by businesses to track and manage capital expenditures. It serves as a detailed log or record of all significant capital expenditures made by a company. Capital expenditures are investments made in assets that provide long-term benefits and are not intended for immediate resale. These might include machinery, equipment, land, buildings, or significant improvements that will generate value for the business over an extended period. A contractor's final account refers to the detailed financial summary of a construction project once it reaches completion. It is a comprehensive statement that outlines the total cost of the project, covering all the expenses and revenues related to the construction work undertaken by the contractor. This final account is created at the conclusion of the project and typically involves several key components. After the initial consultation, the information gathered is further analysed by the consultants and are categorized to several capital allowance pools. The rules and methods for calculating capital allowance for each pool depends on several conditions such as date of the expenditure and rules imposed by government.

The assessment of capital allowances has predominantly relied on manual processes, heavily dependent on the expertise and skills of individual tax consultants. Depending on the project's scale, it can take days or even weeks for an experienced tax consultant to complete a claim from start to finish, making it an exceedingly time-consuming task.

Also, the complexity of the capital allowance rules and the frequent changes in legislation further complicates the issue. Typically, corporate clients receive services from the big accounting firms and a few large quantity surveying (QS) firms, leaving small and medium enterprises (SMEs) excluded from these tax benefits. This exclusion is mainly due to the prohibitive consultation fees charged by smaller consultancies or accounting firms that often lack expertise in calculating capital allowances. The distinct nature of construction expenditure on buildings further complicates matters, with each property and project having unique costs and data. While guidelines exist for large projects, the absence of standardized reporting for small refurbishments exacerbates challenges for SMEs, accountants, and government tax agencies, hindering their ability to obtain accurate data for tax claims.

These challenges underscore the necessity for the development of automated technologies and systems for capital allowance assessment.

In this paper, we report the development of an automated capital allowance assessment system to address the aforementioned problem.[1] The system has three key components: (1) a capital allowance expert system covering the whole process of capital allowance assessment, (2) a tax coding system automatically classifying textual costing items into corresponding tax codes, and (3) an integrated web-based application to offer users the interactive experience of data gathering, analysis, coding and report generation.

The rest of the paper is organized as follows: In Section II we review the previous studies on the relevant areas of tax assessment, expert systems, natural language analysis and web-based application development, as well as the overview of the proposed system for automated capital allowance assessment. The rule-based expert system is introduced in Section III. Details of the tax code classification are discussed in Section IV. Section V provides the design of the web application, key modules of the software, database structure, deployment of the system on cloud, and UI/UX design. Finally, conclusions are drawn in Section VI with benefits for businesses, research gaps and insights into potential enhancements in the future.

[1]The system was the outcome of a collaborated project between Brunel University London and Veritas Advisory. To our knowledge, it is the first commercial AI-based captital allowance assessment system in the UK.

II. BACKGROUND

A. TAX RELIEF AND CAPITAL ALLOWANCE

Capital allowance and tax relief are crucial components of the financial framework for businesses, with far-reaching implications for investment, economic performance, and financial decision-making [6], [15], [35].

The impact of tax relief on public finance has been explored, emphasizing its influence on individual and corporate financial behaviour and its potential positive or negative effects on economic and social factors [5]. Additionally, tax-induced trading around tax relief acts has been studied, demonstrating the significant implications of tax relief policies on investor behaviour and market dynamics [3]. Tax relief awareness and the importance of its utilisation has been investigated in the context of its financial consequences to small businesses [2].

Capital allowance is a relief given to individuals or businesses who have acquired qualifying capital expenditure for business or trade purposes [22]. This incentive has been shown to have a positive effect on the performance of businesses [22]. Additionally, the impact of corporation tax on investment in the UK has been studied, highlighting the importance of capital allowances in influencing investment decisions [6]. Furthermore, the strategic impact of business intelligence utilisation has been linked to economic performance, with broader implications of financial incentives [23].

The importance of these financial incentives is further underscored by their relevance in the context of global economic shocks and crises. For example, during the COVID-19 pandemic, small businesses in US sought funding through relief programs such as the Coronavirus Aid, Relief, and Economic Security (CARES) Act, highlighting the critical role of government-initiated relief measures during times of crisis [4]. Furthermore, the impact of tax relief laws on investment asset allocation in South Korea has been examined, providing insights into the tangible effects of tax relief policies on investment decisions and asset allocation strategies [8].

B. ARTIFICIAL INTELLIGENCE AND EXPERT SYSTEMS

Expert systems and artificial intelligence (AI) have been increasingly integrated into various domains, including finance and taxation.

In the context of tax assessment, AI has shown potential addressing tax noncompliance [17]. The use of AI in finance and supply chain networks, has been explored in [25]. The role of AI in combating tax evasion and its impact on financing constraints for non-state-owned enterprises in emerging markets has been investigated [32]. In the context of taxation, the application of AI and machine learning in the Indian taxation system has been studied [29]. The impact of tax systems on tax evasion has been investigated, emphasizing the moderating role of social norms in influencing tax compliance [17]. A graph neural network model named Eagle was introduced to enhance tax evasion detection within a heterogeneous graph [33]. Additionally, solutions combining Q-learning and recent advances in Deep Reinforcement Learning were proposed assess the tax evasion behaviour of risk-averse firms [16].

The development and application of expert systems has been a subject of extensive research. Expert systems have been elevated for their performance in various domains, such as medical diagnosis [18], [27], [38]. Furthermore, the expert systems in evaluating students' curricula based on competencies has been demonstrated, indicating their applicability beyond traditional problem-solving domains [36]. The development and validation of expert systems have been crucial in enhancing their reliability and effectiveness in decision support [28], [38].

Expert systems have also been applied to tax assessment. Early studies include the development of an expert system addressing income and transfer tax planning for individuals [24], and an expert system integrated into the corporate tax accrual and planning function of an accounting firm's audit and tax practices [34]. The integration of expert systems and AI in finance has been a topic of interest, with studies highlighting the challenges, techniques, and opportunities associated with their implementation [7]. Moreover, the influence of AI on the financial industry, in particular on the need for transparency and risk management in the evolution of finance, has been examined [40]. Leveraging reimbursement strategies to guide the value-based adoption of medical AI has also been explored, reflecting the intersection of AI, finance, and healthcare [37].

C. SYSTEM OVERVIEW

The system consists of three main components which are capital allowance expert system, automated tax code classification and web-based application. Figure 1 presents the three components and their relation and interaction with each other.

1) The capital allowance expert system contains the rules and procedures extracted from the standardised process and expertise of the company [9]. It is implemented on the back-end using Python Flask API, Mongodb database, web development languages (CSS, HTML and Javascript) and Microsoft Azure cloud platform. Rules of capital allowances are implemented in the back-end which are applied based on the project types and apportion them on some of the categories like fees, preliminaries and builders work.
2) The tax coding system classifies the costing items in the ledgers and final accounts into pre-determined tax categories. In accordance with the current UK legislation, these categories belong to five different pools Plant and Machinery, Special rate Pool, Revenue Deduction, Structure and Building Allowances and Non Qualifying items. These pools are reflected in the final report as final claim items with details and breakdown of each of them. There are also categories like Fees, Preliminaries, Builders Work (BWIC) which are apportioned to the main pools mentioned in the table. Including all categories, there are 31 overall categories which are used for classification problem. Table 1 list all the costing categories, their descriptions and the tax codes.
3) The web-based application provides user interface and functions for the clients to obtain the information required for a standard capital allowance project. These information are provided in the interface by filling out forms and uploading documents. These information about the project are necessary to find out more about the project and apply the right rules to get the most accurate results. Having NOSQL database (Mongodb) enables users to retrieve their projects, make modifications and upload new data or track their project progress.

In the next sections, we will discuss each of these components in details.

III. CAPITAL ALLOWANCE EXPERT SYSTEM

The flowchart depicted in Figure 2 outlines the sequential steps involved in generating AI-driven reports, starting from the user's input of information to the automatic generation of reports in Excel format.

Upon user submission of data in the system, the software initiates the analysis of the data. It then proceeds to automatically code items in the ledger and final account using pre-trained machine learning models embedded in the software package. The system searches for lump sum values based on the description and individual item values, subsequently generating benchmark charts specific to project and property types. The next step involves comparing these benchmark values with AI-generated values and deriving a score on a scale of 0 to 100.

The calculation of the score is based on a set of pre-defined benchmark rules compiled from tax consultants and adjusted from thousands of past projects conducted by the company. If the obtained score surpasses the predefined threshold of 85% (as determined by company experts), the system prompts users to download their report. Conversely, if the score falls below this threshold, the system notifies users of potential issues in the files or information provided, based on how well it matches against the benchmark rules stated above. It recommends revisiting and re-uploading new files or overwriting potentially misclassified items through the application interface.

After the application of coding and breaking down packages, the updated and coded files are saved, rendering them ready for report generation. Within the application package, several modules implement capital allowances rules. These modules create pivot tables from each of the coded files and

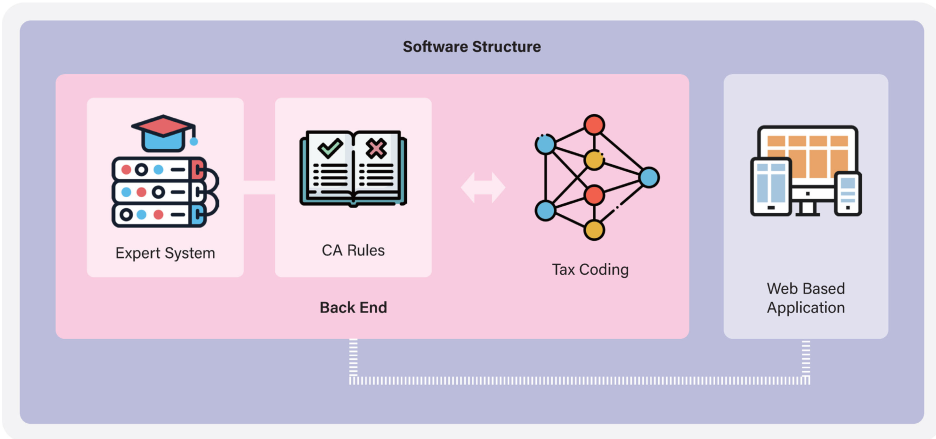


FIGURE 1. Overall structure of the capital allowances system, where a rule-based expert system encodes the standardised process of assessment, a text coding system automatically classifies the costing items into pre-defined tax categories, and a web-based application facilitates the interaction with users.

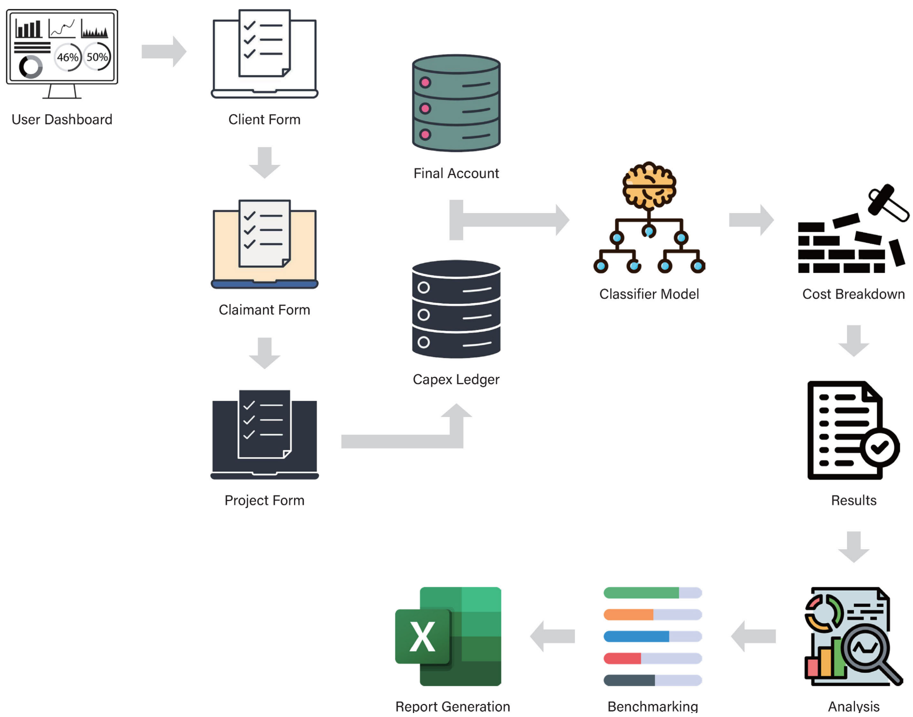


FIGURE 2. The complete process of capital allowance assessment, from data gathering, analysis of ledger and final account, automated tax coding, cost breakdown, benchmark analysis and report generation.

then apply various rules to calculate the totals for each item, placing them into their corresponding capital allowances pools. These pools encompass Revenue, Integral Features, Plant and Machinery, Structures and Building Allowances, and Non-Qualifying Expenditures, as shown in TABLE 1. The pools play a crucial role in the final report. Another step involves splitting the final values based on the year ends and producing the final report according to the year of expenditure, information provided in the ledger file.

A. GENERATE SUMMARY PAGE

The primary function of the application is to automate the coding of items using machine learning models. These models assign categories to each item, which are then utilized to generate a final report based on capital allowances rules. The complexity of the rules arises from the treatment of items depending on project and property types. The final report comprises various categories of expenditure items, combining ledger and final account details. Certain categories, such as Preliminaries, Fees, and Builders Work in connection, are apportioned over other categories. Fees can be service-related or from direct expenditure (ledger), and their apportionment differs.

When installing a qualifying capital allowance item in a property, costs directly associated with the provision of that item also qualify (on costs). Construction project items are costed based on key elements like labor, plant, and materials. For instance, the installation of lighting involves the electrician, scaffolding, light fitting, and wiring. The main contractor provides builders works related to services (creating openings in walls for wiring), as well as preliminaries (project security, safety, welfare facilities, cleaning, and management of materials and plant). In design and build contracts, the main contractor hires a designer, services engineer, and project manager, and all associated fees directly relate to the installation cost. Clients may engage external advisors in the initial design or scope, considering how overall services integrate into the property. Hence, it is permissible to incorporate or distribute these items over the main plant and machinery and other pools.

The initial stage in report generation involves distributing the values of builders work in connection (BWIC) across items associated with that category. These items encompass hot water, cold water, air conditioning, fire alarms, electrical systems, computer and telecommunication, burglar and access control systems, and gas installations, totaling 13 categories. The amounts allocated for BWIC in both the ledger and final account are then apportioned among these specified items.

In the second phase, the process involves distributing preliminary (PR) amounts from both the ledger and final account across all remaining categories. This entails aggregating the PR amounts from both files and calculating their respective percentages. Subsequently, for each category within the main pools (plant-machinery and integral features), these percentages are added to the overall PR percentage in the project. When computing the PR percentages, the amount corresponding to Fees is deducted from the overall percentage. Thus, the percentage is derived by dividing PR values by the overall value of the final account, minus the total Fees value. This resulting percentage is then applied to each value within the main pools.

Subsequently, Fees and service-related fees in the final account undergo an apportionment process over the main pools. These values are then added to the total derived from the PR apportionment stage. While the formulas for Fees and SFees (service-related fees) differ slightly, they are calculated separately, yet their methodologies align closely with the preceding steps.

This procedure is iterated for fees and service-related fees in the ledger, also conducted separately. Notably, the values obtained from direct fees are multiplied by the values obtained from the last step (final account fees). However, for direct service fees, the values are acquired by multiplying the percentage with the initial values obtained from the pivot table at the outset.

The ultimate outcome is encapsulated in a table referred to as the summary table. This table encompasses all the computations from the preceding steps, with the final value for each item representing the aggregate of values obtained at each stage. These cumulative values provide the conclusive figure for the main pools, namely Plant-Machinery and Integral Features. Refer to Figure 3 for a comprehensive breakdown of the steps involved in generating the summary page and its corresponding columns.

The computation of values for the other pools, SBA and REV, follows a distinct approach and is carried out independently. Two analogous tables are generated after the apportionment of fees, PR, and BWIC over these pools.

To derive the SBA value, only Fees in the final account and ledger undergo apportionment over it. Unlike in the main pools, BWIC and SFees are not allocated over SBA items, leading to a variation in the formulas used.

For the determination of values related to revenue items, the calculations for REV (revenue) and SREV (service-related revenue) are conducted separately. Unlike BWIC items, which are not apportioned over revenue items, the other calculations mirror those executed for the main pools. The outcome is a table akin to the main pools table, summarizing the results of the apportionment process for SBA and REV.

In cases where there is a difference between the final account value and the main contractor value in the ledger, these differences are handled differently, primarily being split as Retention or Outstanding Value. Certain categories necessitate apportionment to other capital allowance pools to calculate the overall summary of final values. The system calculates values for each pool by incorporating these apportionment items, such as Fees, Preliminaries, and Builders Work. The algorithms in the system then generate summary results for each of the pools, with specific rules applied according to the nature of each pool. For instance, apportionment for Structures and Buildings differs from the main pool apportionment.

B. RECONCILIATION OF FINAL ACCOUNT AND LEDGER

The subsequent step involves calculating and reconciling the values with the final account and total expenditure to ensure accuracy and alignment. The system then addresses differences between the final account total and the main contractor payment in the ledger, considering client preferences. Typically, if the final account value is within a 10% difference with the ledger main contractor payment and the ledger value is less than the final account, the outstanding is split as retention. However, if the difference is more than the final account value, it is split as unanalyzed expenditure.

C. SPLITTING YEAR ENDS

The final step is to distribute the calculated pools based on the year of expenditure for accounting purposes. The user-provided year-end information in the interface is utilized to split values and provide details based on the year of expenditure. The year ends of the report are obtained from ledger depending on what date the construction or main contractor payments are made through company accounts.

The final report contains different sections covering the pool values divided to year ends, the breakdown of each pool and reconciliation of ledger or direct expenditures.

Expenditure Category	Original Value	Builders Work	Preliminaries	Contract Fees	Contract Service Fees	Contract Total	Direct Fees	Direct Service Fees	Direct Expenditure	Total Ledger	Retention	Total Expenditure
Structures & Building Allowances	54,290	0	18,124	0	0	72,414	10,356	0	0	82,770	0	**82,770**
Air Conditioning, Heating & Ventilation Systems	31,250	822	10,707	0	0	42,779	6,118	0	0	48,897	0	**48,897**
Cold Water Systems	6,250	164	2,141	0	0	8,556	1,224	0	0	9,779	0	**9,779**
Computer, Telecommunication & Surveillance Systems	8,750	230	2,998	0	0	11,978	1,713	0	0	13,691	0	**13,691**
Electrical Systems	40,750	1,072	13,962	0	0	55,784	7,978	0	0	63,762	0	**63,762**
Fire Alarm Systems	500	13	171	0	0	684	98	0	0	782	0	**782**
Furniture, Furnishings & General Equipment	57,147	0	19,077	0	0	76,225	10,901	0	0	87,126	0	**87,126**
Hot Water Systems	3,750	99	1,285	0	0	5,133	734	0	0	5,868	0	**5,868**
Machinery	750	0	250	0	0	1,000	143	0	0	1,143	0	**1,143**
Sanitaryware	3,750	99	1,285	0	0	5,133	734	0	0	5,868	0	**5,868**
Non Qualifying	0	0	0	0	0	0	0	0	6,000	6,000	0	**6,000**
Total	**207,188**	**2,500**	**70,000**	**0**	**0**	**279,688**	**40,000**	**0**	**6,000**	**325,688**	**0**	**325,688**

Ledger: £ 325,688 | **Final Account:** £ 279,688

FIGURE 3. Generating the summary table of aggregated values of each expenditure categories.

IV. AUTOMATED TAX CODE CLASSIFICATION

The automated tax code classification within the system is the result of training on company data from past projects, validated using the deep learning methods. In this section, details of the machine learning models are provided to assess its performance. The models are designed to be self-learning, continuously improving its capabilities over time by assimilating new coded data uploaded to the database.

Natural Language Processing (NLP) is characterized as technology facilitating communication between humans and computers [14], analyzing and representing naturally occurring texts at various linguistic levels, in order to achieve human-like language processing for a diverse array of language-related tasks [12]. In the last decade, deep learning approaches on textual data have emerged as a potent technology, widely applied in fields such as accounting and finance. Notable examples include ChatGPT and large language models, serving as impressive demonstrations of the practical use of deep learning methods [39]. NLP and deep learning, specifically Recurrent Neural Networks, are the foundational methods employed to construct text classifiers for the textual data in capital allowances. According to [39], deep learning methods are more consistent and perform exceptionally well for textual data in the realms of accounting and finance.

Modern data science techniques, encompassing raw data collection, cleaning, preprocessing, model training, validation, and testing, are systematically implemented and evaluated in this application. The aim is to build reliable, efficient, and robust models suitable for deployment in production. The application, along with ML model files, is deployed on Microsoft Azure cloud services as software, serving users during project generation and document analysis.

A. DATA COLLECTION

The initial step in any machine learning project involves identifying the business problem and devising a solution by acquiring data tailored to the problem's requirements. Text analysis and classification, being a supervised machine learning problem, necessitate labeled data. Fortunately, the company had compiled files and projects containing ledger and final account items with descriptions, all labeled in their Excel template files. The significant challenge lay in identifying methods and algorithms to automatically extract all these items from the numerous Excel report files. A software interface was developed to automatically collect data from over 400 projects, successfully retrieving 28,000 items with their corresponding labels.

In addition to project data, there are standard industrial price books for capital allowances [1] providing valuable information on items with descriptions and related labels. Collecting information and automatically labeling them proved beneficial, resulting in an additional 7,000 items with labels, subsequently justified and revised by company consultants. These were then incorporated into the overall dataset, totaling 35,000 description items and their tax code labels. Collecting data from past projects and employing text mining to extract information from reference price books constituted the primary effort in amassing valuable data for the models.

Given that the data was imbalanced, with a majority of items falling into only four categories (SBA, FFE, ES, and PR) accounting for nearly 70% of the total data, efforts were made to address this imbalance. Various techniques, including cutting and oversampling, were implemented to balance the data. After these adjustments, the most frequent category comprised just 25% of the total data.

Even after trimming the most frequent items, the data remains imbalanced, underscoring the significance of certain items over others. The collected data encompasses a total of 31 categories, and this diverse set was utilized for training the model. This decision was driven by two primary considerations: firstly, it provides the model with a broader vocabulary to operate with, and secondly, it accurately mirrors the typical distribution of data observed in each project.

The collected data needs preprocessing to be prepared for training tax coding models. The methods used for preprocessing data for text classification are generally similar across various Natural Language Processing (NLP) tasks. The data typically includes descriptions of items along with their categories, and these need to be cleaned and processed for training.

To clean the data, several essential steps are performed using Keras tokenizer. These steps involve filtering special characters, converting all words to lowercase, lemmatization, and creating a vocabulary of words in the corpus. The tokenized words are then converted into numerical vectors for training. Subsequent steps in preparing the data for training involve converting text to sequences and padding, which are implemented using Keras' Preprocessing library. For more detailed information, it is recommended to refer to the documentation on the Keras Preprocessing library [19].

B. THE MODELS

Three models, LSTM (Long Short Term Memory) [31], BERT [11] and DistilBERT [30] have been developed and included in the system. We will detail the training and evaluation of these models as follows.

1) LSTM

The LSTM model [31] is a type of recurrent neural network that overcomes several constraints of traditional RNNs, such as vanishing gradients and difficulty learning long-term dependencies. It achieves this by using special multiplicative units called memory blocks and gates to control information flow and store the network's state in memory cells with self-connections. This creates a hidden state set that includes sequential details about the sentence's word order. Finally, the last hidden vector is used as the input for the output layer, which represents the phrase for this baseline.

The model built for training the LSTM contains an embedding layer with a dimension of 512, an LSTM layer with a dimension of 128, two hidden layers, and an output layer with an activation function set to softmax, catering to 31 categories for estimation. Categorical CrossEntropy serves as the loss function during training, with the Adam optimizer utilized for model optimization. A batch size of 32 is employed, and a maximum sequence length of 250 is used for sequencing the data.

2) BERT

Bidirectional Encoder Representations from Transformers is known as BERT [11]. Bidirectional describes how BERT learns about both the left and the right sides of a token's context during the training phase. It is recommended to pre-train deep bidirectional representations from an unlabeled text by concurrently conditioning the left and correct context. The pre-trained BERT model may be enhanced to produce cutting-edge models for several NLP applications with only one additional output layer. The meaning of each word in this statement will be revealed as it is said, one at a time. BERT has already received training on an extensive corpus of unlabelled text, such as the Book Corpus and Wikipedia (which include 800 million words).

In essence, BERT is a transformer design that uses an encoder stack. An encoder-decoder network using a transformer design employs self-attention on the encoder side and attention on the decoder side. BERT utilises the masked language modelling technique to prevent the word in focus from "seeing itself" or having a fixed meaning irrespective of its context. Both BERT model sizes feature many encoder layers called transformer blocks. Instead of having a predetermined identity, their context defines words in BERT [21].

Four layers were utilised to develop the model after the first two input layers, which defined the input ids and attention masks of the tokenisation were built. The first is the BERT embedding layer, which represents each word using input ids and attention masks. The second layer is the dense layer with 512 neurons and a SoftMax activation function. The dropout layer, which is the next layer and has a unit of 0.3, aids in keeping the model from overfitting. Each class has 31 output values in the output layer's last layer.

To implement BERT (base case), we use Keras Transformers. The learning rate of the model was set to 1e-5 using a Categorical Cross entropy loss function and Nadam optimiser, and the training set included 15 epochs. Therefore, the model was evaluated when the training ended.

3) DISTILBERT

DistilBERT [30] is a distilled version of BERT, allowing performance to be maintained using fewer parameters. Utilising a compression technique called distillation, DistillBERT trains a tiny model to behave in a way that is like a larger model. DistilBERT offers some analysis of the BERT architecture and advises specific actions to reduce the number of parameters, such as removing the token-type embeddings and the pooler and reducing the number of layers by a factor of two. DistilBERT and BERT have the same overall architecture. By removing the token-type embeddings and pooler, the number of layers is reduced by 2. In modern linear algebra frameworks, most of the operations utilised in the

Transformer design (linear layer and layer normalisation) are highly optimised. According to the research, differences in the hidden size dimension, the tensor's final dimension, have a more insignificant impact on computing efficiency than variations in other variables, such as the number of layers, given a set parameter budget [20].

The model structure is similar to the BERT base. In our system, a lighter version called the DistillBERT base case is used. Four layers were utilised to develop the model after the first two input layers, which defined the input ids and attention masks of the tokenisation were built. The first is the Bert embedding layer, which represents each word using input ids and attention masks. The second layer is the dense layer, which has 512 neurons and a SoftMax activation function. The dropout layer, which is the next layer and has a unit of 0.3, aids in keeping the model from overfitting. Each class has 31 output values in the output layer's last layer.

To implement DistilBERT, we use Keras Transformers. The learning rate of the model was set to 1E-5 using a Categorical Cross entropy loss function and Nadam optimiser, and the training set included 15 epochs.

C. EXPERIMENTS AND RESULTS

The data is split into two main parts: the training data and the test set. The training data constitutes 85% of the total data, leaving the remaining 15% for testing. This implies that the model undergoes training on 85% of the data and is then tested on the remaining 15%. The test data is not involved in the training process; rather, it is reserved for the final evaluation of the model's performance.

Moreover, the training data is further subdivided into two segments: 15% for evaluation and 85% for the actual model fitting. This division is implemented to ensure thorough model evaluation and to guard against overfitting. Overfitting can occur when the model is excessively tailored to the training data, hindering its performance on new data. Cross Validation techniques also are used to ensure that the performance of the model is similar along all the training dataset.

Once the model is trained and validated, it is deployed on the application platform to function as a classifier for the documents provided by users. The model files, saved in TensorFlow h5 format, are stored in a folder within the application package. These files are loaded and utilized for inference. The entire application is deployed on the Microsoft Azure cloud to serve clients.

The application undergoes testing on various live projects, and the results consistently demonstrate an average difference of around 20 percent for each capital allowances pool. This validation showcases the software's effectiveness, enabling consultants to significantly reduce time spent by correcting the approximately 20 percent of misclassified codes. Consultants can swiftly issue reports based on the developed algorithms for the calculation of different pools. This section delves further into the evaluation of the application, comparing its results with the reports generated by consultants.

On testing the tax code classification models, the results exhibit robustness and consistency, with an accuracy score of 72% on the test data, representing 15% of the total dataset. It has been observed that past data contains noisy and miscoded items, attributable to human error. The accuracy level is expected to improve over time as the system continues to learn from new items introduced when new projects are loaded onto the system. Despite these challenges, the system consistently generates reliable and robust reports, particularly when the values align with benchmark pool values based on the project.

The second notable consideration is that the input data may not provide sufficient information for the models to accurately classify them. This aligns with the software development adage "garbage-in, garbage-out," and can be addressed by guiding users to provide more accurate information or offering more precise benchmarks for their use.

There are two sets of tax code classifiers deployed in to the system which individually classify data from ledger and final account respectively. Based on the data collected from company projects and reference books for capital allowances to feed in to the models, the accuracy score obtained on test data has been around 72% for both ledger and final account files. The classification report for each category is obtained for each of the models and it shows that model is robust and able to distinguish between different categories.

Table 1 also presents the precision and recall scores for each category, demonstrating that the trained model can identify each category with a similar level of accuracy. The categories with lower precision and recall scores are those that may be subject to interpretation based on project types or user preferences. Additionally, due to data skewness, less frequent and less important items exhibit lower scores, as there is insufficient support for them to be accurately trained and classified by the classifier. Items such as Manufacturing Processing and Storage Equipment (MPS), Hoists (HO), or Combined Heat and Power (CHP), which rarely appear in projects, have minimal impact on the overall performance of the model, so those rare items are excluded from the Table 1.

D. DISCUSSIONS

These models can undergo continuous improvement over time as more data is fed into the system, leading to enhanced model performance in the future. To address this, two solutions have been identified to improve the accuracy and efficiency of the system for end users.

The first solution involves presenting the top three probabilities extracted from the classifier model as the top predictions of the model. This allows users to choose the correct code from the recommended labels based on their knowledge. While the accuracy on the test data, considering

TABLE 1. Pools, expenditure categories and their descriptions, and the classificatioin performance by precision, recall and F1-score from the BERT model.

Pool	Category	Description	Precision	Recall	F1-Score	Support
Plant & Machinery	FFE	Furniture, Furnishings and General Equipment	0.83	0.78	0.80	608
	FA	Fire Alarm Systems	0.77	0.76	0.77	114
	BWA	Burglar, Welfare, Access Control Systems	0.77	0.80	0.79	94
	CTS	Computer, Telecommunication, Surveillance Systems	0.75	0.74	0.74	182
	MA	Machinery	0.89	0.80	0.84	103
	DA	Decorative Assets	1.00	0.12	0.22	38
	GS	Gas , Sewerage Installations	0.50	0.45	0.47	59
	SA	Sanitaryware	0.82	0.85	0.84	124
	SD	Signs, Displays , Similar Assets	0.76	0.93	0.84	86
Special Rate	HVAC	Air Conditioning, Heating and Ventilation Systems	0.78	0.72	0.75	335
	ES	Electrical Systems	0.87	0.85	0.86	539
	CW	Cold Water Systems	0.79	0.86	0.82	86
	HW	Hot Water Systems	0.92	0.85	0.88	119
	LEM	Lifts, Escalators and Moving Walkways	0.71	0.77	0.74	63
	TI	Thermal Insulation	0.89	0.53	0.67	52
Revenue	REV	Revenue Deductions	0.64	0.56	0.60	258
Structures & Buildings	SBA	Structures and Building Allowances	0.75	0.77	0.76	875
	SSBA	Service Related SBA	0.75	0.38	0.50	41
Preliminaries	PR	Preliminaries	0.74	0.83	0.78	257
Fees	FEES	Fee	0.75	0.79	0.77	100
	SFEES	Service Related Fees	1.00	0.44	0.61	16
Builders Work	BWIC	Builders Work In Connection	0.61	0.78	0.68	65
Non Qualifying	NQ	Non - Qualifying	0.67	0.39	0.50	101
		Accuracy			0.74	4365
		Macro avg	0.71	0.67	0.68	4365
		Weighted avg	0.74	0.74	0.74	4365

TABLE 2. Test results for three models LSTM, BERT and distilbert for first and three top choices. The top three choices are provided to the users for easy manual adjustment as shown in Figure 6.

	Top Accuracy	Top 3 Accuracy	Loss
LSTM	72%	95%	1.65
BERT	74%	97%	1.20
DistilBERT	73%	95%	1.46

only the top probability, is 72%, considering the three options boosts the accuracy to 95%. This significantly addresses the issue of inaccuracy and renders the machine learning models more robust and reliable.

The second solution is to incorporate a confidence level, representing the highest probability assigned by the machine learning model for each item. This approach is widely used in various AI products. Users can access the confidence of the model for each item, presented as a probability ranging from 0 to 1, scaled to 0 to 100 where 0 implies no confidence and 100 signifies full confidence. Users have the option to overwrite the codes based on confidence scores. Figure 6 illustrates the data table provided for users, including the three highest options and confidence levels.

Table 2 displays the results for each model trained on the data, with test scores provided based on 15% of the total data. The trained model exhibits robust scores, making it suitable for use in production. While the two BERT models offer slightly better classification accuracy, the LSTM model is able to inference without the need for GPU computation and requiring lower storage. It demonstrates faster performance even with a CPU and incurs lower costs than BERT Transformers when deployed on cloud platforms. The choice of different tax coding models can be switched through system configuration.

While there is ample room for improvement in model scores as more data is collected over time through system usage in production, the current model performance is sufficiently suitable to enhance report efficiency, expedite the consultancy process, and be deployed in production. The data used for model training is constrained by company resources, but the model is expected to improve through its self-learning process over time.

V. WEB-BASED APPLICATION

The back-end of the application is created using the Python Flask API, while the front-end is developed with a combination of JavaScript, jQuery, Bootstrap, HTML, and CSS. The front-end design prioritizes meeting user needs at the highest standards, incorporating both internal and external user feedback to enhance the user-friendliness of the application. The back-end is structured with various components, including Authentication, Capital Allowances rules, project creation, editing, sharing, PDF analysis, result monitoring, and plotting. Each of these modules is further divided into different sub-modules.

Numerous web pages and functionalities have been designed and developed to offer users an efficient and user-friendly platform. Major functionalities and modules of the application are presented in Figure 4. It illustrates the back-end structure, showcasing modules and their interconnections with the front end.

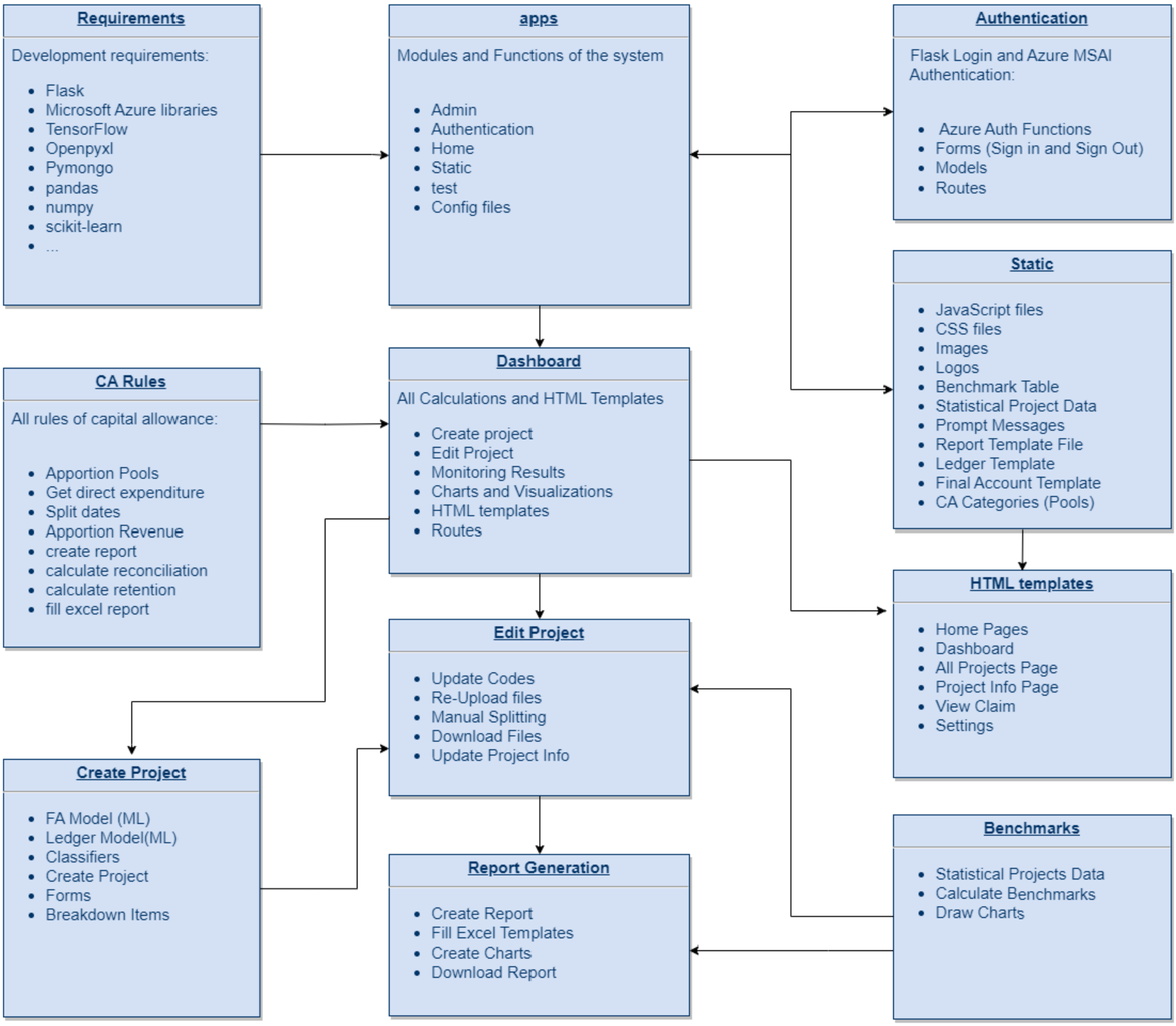


FIGURE 4. Block diagram of major modules and functions in the system.

A. USER DASHBOARD

User dashboard encompasses all the forms and tables pertaining to clients and claimant entities, which users interact and create their project. The client information, representing the highest level of details, is initially filled out by users and includes information about the client company. Subsequently, users provide details about associated claimant entities, which are entities related to the client company. For instance, if a user has a client company with multiple claimant entities (companies under the client's name), they must furnish details about these associated claimant companies relevant to the project they wish to undertake. The third level involves inputting project-specific information crucial for generating accurate reports. Once users complete all the forms, they can swiftly generate their reports, gaining immediate access to the results. Figure 5 visually depicts the interface and elements of the dashboard, facilitating users in initiating and producing their claims.

B. CREATING PROJECT

One of the major modules of the application is Create Project module. All the functionalities related to this module are encapsulated in it and its function is to analyse the input data from the client, analyze the information and document in sub modules and generate initial report by using deployed ML models.

The primary processes within this module involve obtaining data from the client, performing validation, utilizing machine learning models to classify items based on their descriptions, and employing breakdown models to split costs.

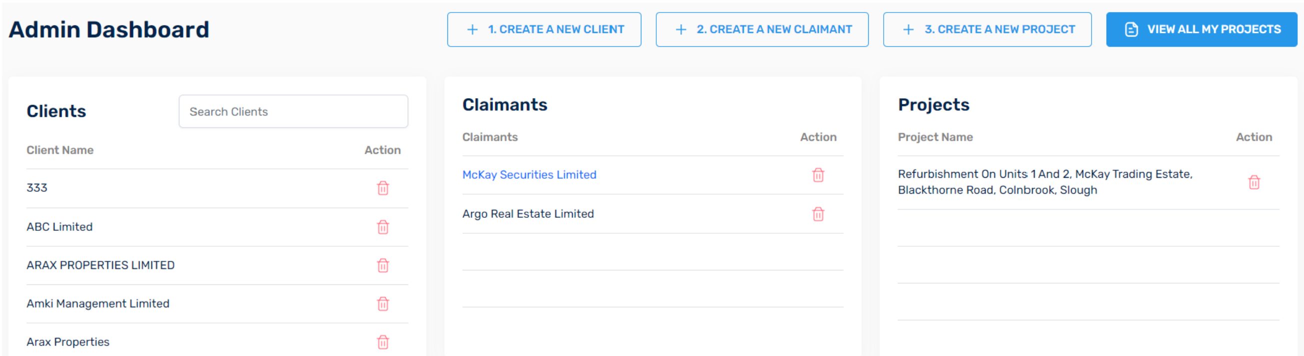


FIGURE 5. Dashboard containing the generated clients, claimants and projects.

The final step directs users to the relevant project page based on the processed information.

1) REQUEST DATA

Users are required to upload documents in Excel format using templates provided within the application. These templates serve the purpose of guiding users to include essential headers in each file. For instance, the ledger file must include four mandatory headers: Date, Description, Supplier, and Value. Similarly, the final account file should contain different headers, specifically Description and Amount. Adhering to these header requirements ensures the inclusion of necessary information in the respective files.

2) EMPLOY ML MODEL

Upon requesting and validating user data to ensure the presence of required columns, the system deploys classifiers to categorize item descriptions in both ledger and final account files. Two distinct machine learning models are employed for text classification, as the ledger data introduces an additional category.

In the ledger categorization process, an extra step is incorporated in the classify ledger function. This step involves searching for the contractor name in the supplier column of the ledger. If a match is found, the system overwrites the codes with "MC" (main contractor) to ensure accurate identification of main contractor values in the ledger. This information is crucial for subsequent calculations and the generation of accurate reports. The "MC" values, representing amounts paid to the contractor, ideally align with the values in the final account file.

During the coding of files, when calculating the maximum probability of the activation function in the neural network, the top three probabilities are selected and saved in tables. This provides users with options to choose one of the three probabilities through the interface, contributing to the improvement of claim accuracy. The highest probability is also stored as the confidence level of the model, offering users insight into the model's confidence for each item. Further details on this process will be discussed in subsequent sections.

3) BREAKDOWN PACKAGES

Following the rapid generation of coded ledger and final account, the subsequent step involves identifying items that may need to be split into multiple breakdowns, referred to as "packages" in the context of capital allowances. Packages are items within the files that have the potential to be divided into various categories. These packages are split based on breakdown models established by company experts, and the results are saved as separate Excel files within the application package. Utilizing these models corresponding to the project type, lump sum (package) items are segmented into multiple codes. This process enhances project accuracy and automates the report generation process.

Specific algorithms within the application are dedicated to this task, operating under two conditions. Firstly, the algorithms search for keywords in the descriptions that could potentially indicate a specific package. Subsequently, the algorithms compare the value of the item with a threshold value and proceed to split the item if it satisfies both conditions.

C. EDITING PROJECT

The "Edit Project" module provides users with functionalities to modify their data inputs by allowing them to re-upload data, overwrite codes, and add new items if needed. Upon project creation, users are redirected to the project information page, housing three tables: specific job information, coded final account, and code ledger. These tables empower users to make adjustments and update their information at any stage of the project. Users have the flexibility to revisit and refine their data, ensuring the production of more accurate reports. Once content with their coded information, users can proceed to the "View Claim" page. Figure 6 presents the interface provided for the users to make changes and update the project data or overwrite the codes classified by AI model.

D. VIEWING CLAIMS

The "View Claim" module consolidates all analyses and results derived from the implemented algorithms for applying capital allowances rules and generating the final report.

Ref	Description	Amount	Code	Pool	Confidence	Option 1	Option 2	Option 3	Actions
1	Internal Walls and partitions	12,500	SBA	Structures and Buildings	86.0	☑ SBA	☐ NQ	☐ REV	⋮
2	Internal Door	2,000	SBA	Structures and Buildings	69.0	☑ SBA	☐ NQ	☐ MA	⋮
3	Internal Doors - ironmongery	750	MA	Plant & Machinery	82.0	☑ MA	☐ SBA	☐ FFE	⋮
4	Wall finishes	6,250	SBA	Structures and Buildings	66.0	☑ SBA	☐ FFE	☐ NQ	⋮
5	Floor Finishes	204	SBA	Structures and Buildings	65.0	☑ SBA	☐ FFE	☐ NQ	⋮
↳	Wall finishes	3,451	SBA	Structures and Buildings	100.0	☑ SBA	☐ SBA	☐ SBA	⋮
↳	Floor Finishes	12,423	SBA	Structures and Buildings	100.0	☑ SBA	☐ SBA	☐ SBA	⋮
↳	Entrance Matwell	414	FFE	Plant & Machinery	100.0	☑ FFE	☐ FFE	☐ FFE	⋮
↳	Ceiling finishes	6,212	SBA	Structures and Buildings	100.0	☑ SBA	☐ SBA	☐ SBA	⋮
6	Entrance Matwell	750	FFE	Plant & Machinery	99.0	☑ FFE	☐ SBA	☐ NQ	⋮

Total: £279,688

FIGURE 6. Interface provided for the user to update or edit. Items with red arrows to the left are results from the cost breakdown model. The automated coded item are shown in different colours corresponding to their respective categories, followed by the indicators of confidence from the models. The top three choices are listed as option 1 to 3, which offer an accuracy as high as 97%.

Section 1: Executive Summary

Reference	Total P&M	Total Expenditure	Retention	31-03-2022(100.0%)
Main Pool	108,611	108,611	0	108,611
Special Rate Pool	128,306	128,306	0	128,306

Section 2: Expenditure Reconciled To Accounting Periods

Reference	Total P&M	Total Expenditure	Retention	31-03-2022(100.0%)
Main Pool	108,611	108,611	0	108,611
Non Qualifying Expenditure	6,000	6,000	0	6,000
Notional Structures & Buildings Allowances*	82,770	82,770	0	82,770
Revenue Deductions	0	0	0	0
Special Rate Pool	128,306	128,306	0	128,306
Total Expenditure	**325,688**	**325,688**	**0**	**325,688**

FIGURE 7. The final report is generated automatically and calculated using the AI models and capital allowance rules.

It comprises three tiers of information: a summary of the claim report, an analysis of results, and a reconciliation of the ledger. Users can leverage various elements such as charts, accuracy scores, a summary table, ledger and final account reconciliation, and claim report details to validate the AI-generated results. The interface, as illustrated in Figure 7, visually represents how the results and claim reports are presented to the user.

1) GENERATE REPORT AND MONITORING

The Final Report generated by the combination of AI and expert rules undergoes thorough checks and validation in the back-end to minimize the risk of producing incorrect results. In the event of any discrepancies or errors, the system promptly informs the user through the interface of the application. Various prompts are provided to guide users when their data might lack sufficient information for the classifier. Figure 8 illustrates how the values generated by the AI classifiers are compared with benchmark values estimated by expert knowledge, allowing users to monitor the accuracy of their values.

Benchmark values, derived from the expertise of company professionals across more than 170 projects and property types, play a crucial role in this comparison. Project type refers to the nature of the work undertaken by the client on their property (e.g., fit-out, new building, refurbishment), while property type pertains to the business or use of the property (e.g., office, retail store, industrial).

The system is also designed to collect information on past project and property types, including descriptive statistics such as mean and standard deviation. These statistics are used in combination with expert knowledge. Benchmark

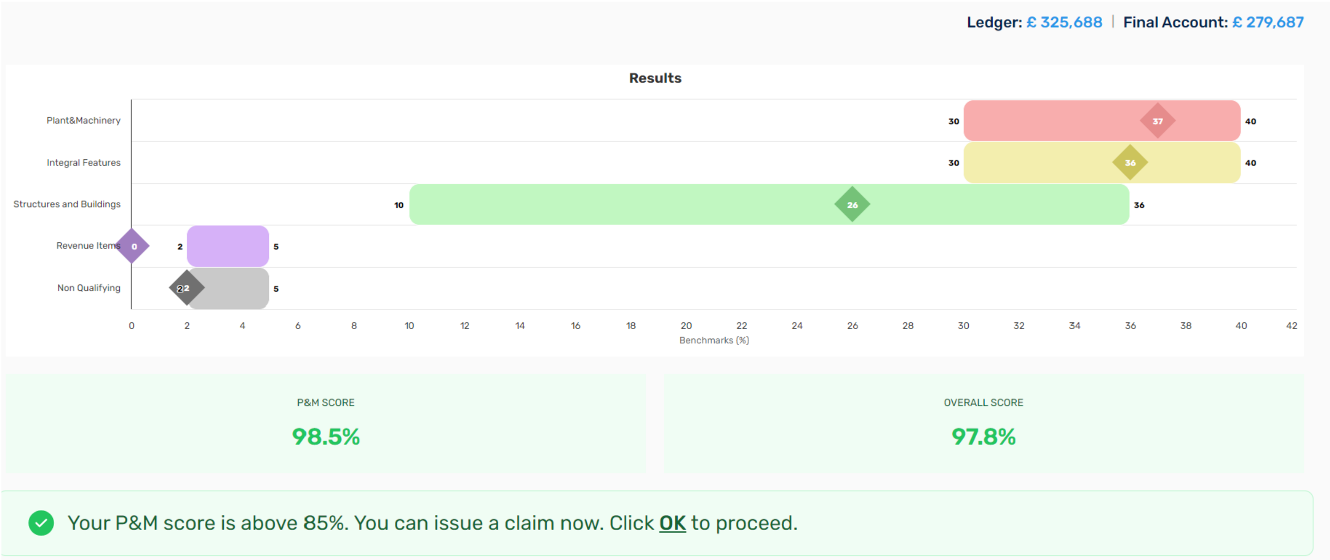


FIGURE 8. The benchmark charts to compare actual values against acceptable ranges from the benchmark models.

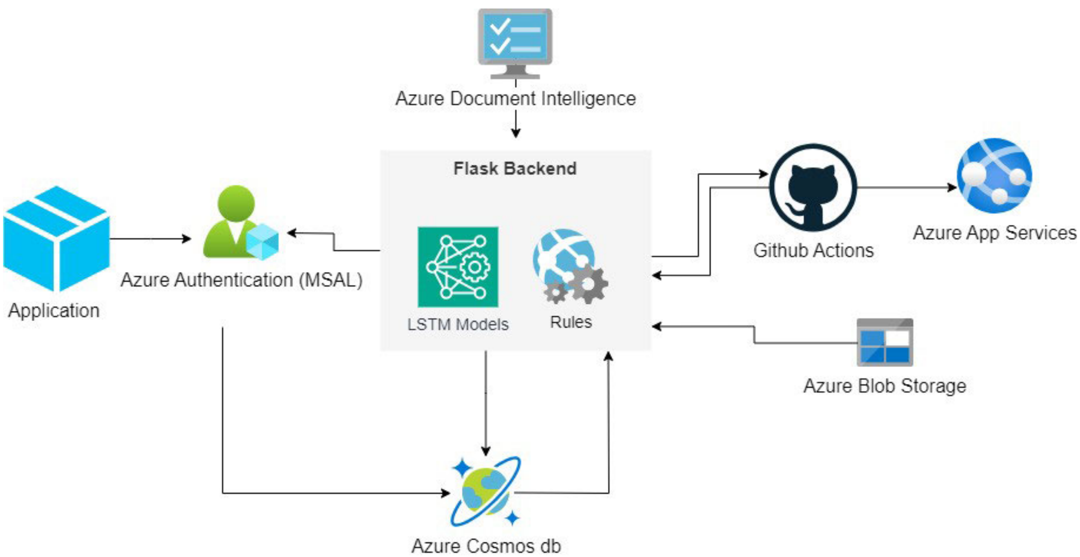


FIGURE 9. Microsoft azure services used for application and deployment.

values are determined by an initial combination of 10% from past projects and 90% from expert estimations. Over time, these weights gradually shift towards statistical data as more projects are added to the system database.

2) FILL UP EXCEL TEMPLATES

The concluding step involves populating an Excel template with various section values for each pool. A dedicated module within the application package is designed to fill each page of the report template, ultimately generating the final Excel report. This report template closely mirrors the templates crafted by experts, with the key distinction being its automated generation without requiring human input—an integral functionality of the expert system.

The algorithm responsible for populating the Excel template is written in the back-end using the Openpyxl [13] library. This library is chosen for its efficiency in generating reports with styling, contributing to increased efficiency and consistency in the presentation of reports. The final Excel report comprises eight pages, with each page or tab providing details about individual pools and reconciliation of ledger and final account information.

Benchmark charts offer valuable insights for users to gauge the accuracy of their claims. They serve as indicators, highlighting potential issues in the data that users may need to revisit and update for improved results. Figure 8 visually represents the benchmarks alongside actual values and accuracy scores, providing users with a comprehensive view of their performance.

The final stage of the user experience involves the options to either download the report or return to the project information page. Users may choose to revisit and modify their input data, or, if prompted by the application, they can proceed to download the generated report for official use.

E. DEPLOYMENT

Deploying the application is a crucial aspect of the development process, ensuring accessibility to the targeted users during the production stage. The deployment leverages cloud facilities, specifically utilizing the Microsoft Azure platform. The entire application package, comprising APIs, Python modules, package dependencies, rules, and static template files, is deployed on Azure app services as a unified package. The deployment process is structured using Continuous Integration/Continuous Deployment (CI/CD) methodology, and thorough testing and validation are conducted using GitHub Actions to assess new features and functionalities. Figure 9 illustrates the services employed on the Microsoft Azure cloud platform, depicting various deployment stages and services provided by Microsoft Azure.

VI. CONCLUSION

In this paper, we have presented a detailed account of the development of an automated capital allowance assessment system. The system is designed with three components, an captital allowance expert system, an automated tax coding system and a web-based online application, each contributing to the seamless and efficient execution of the capital allowance assessment process.

1) The capital allowance expert system incorporates a sophisticated set of rules and procedures extracted from the standardized processes and expertise of tax consultants. This expert system is implemented on the back-end, and covers the whole process from data gathering, analysis of ledger and final account, automated tax coding, cost breakdown, benchmark analysis and report generation.
2) The automated tax coding system plays a pivotal role in classifying textual costing items into corresponding tax codes. Aligned with the UK legislation, this system categorizes items into five distinct pools: Plant and Machinery, Special Rate Pool, Revenue Deduction, Structure and Building Allowances, and Non-Qualifying items. These categorizations are then reflected in the final report, providing a detailed breakdown of each claim item within the specified pools. Notably, there are also specific categories, such as Fees, Preliminaries, and Builders Work, which are intelligently apportioned to the main pools mentioned in the accompanying table.
3) A web-based application is developed to provide an intuitive user interface and diverse functions catering to the clients' needs in obtaining information crucial for a standard capital allowance project. Through this interface, users can effortlessly fill out forms, upload necessary documents, generate detailed tax codes, retrieve their projects, make modifications, upload new data, and track their project progress seamlessly.

The system is implemented using Python Flask API, MongoDB database, and popular web development languages including CSS, HTML, and Javascript. Additionally, the system leverages the robust capabilities of the Microsoft Azure cloud platform, ensuring a resilient and scalable architecture.

To the best of our knowledge, it is the first commercial AI-based capital allowance assessment system in the UK. It addresses the complexities and inefficiencies associated with manual capital allowance assessment, and will have significant impacts in the following aspects:

- Governance and Compliance: The project aids the tax authorities in standardising and streamlining the allowance assessment process, ensuring precision in claims, fostering transparency, maintaining consistency, and promoting compliance. This bolsters governance integrity and aligns with government strategies aimed at encouraging investment.
- Social: It brings accessible allowance assessment services to SMEs, fostering expansion, reducing taxable profits, and encouraging innovation. This contributes to job creation, economic resilience, and enhances SME competitiveness, thus facilitating inclusive growth for businesses and society.
- Environmental: In line with the regulations, it promotes environmental sustainability by encouraging energy-efficient practices, the adoption of renewable energy, and the introduction of green innovations.

Nevertheless, there are still limitations in the current system. Firstly, the capital allowance rules in the knowledge base are primarily predetermined by experts and lack self-learning capabilities. Secondly, while the bottleneck of the AI-based tax coding system is attributed to noisy data, there is potential to employ more advanced models that integrate domain knowledge. Future work to enhance the system will concentrate on improving knowledge base management and tax coding accuracy.

ACKNOWLEDGMENT

The authors would like to thank the Innovate U.K., for the funding.

REFERENCES

[1] *Spons's Architecture and Builders Price Book*, CRC Press, Boca Raton, FL, USA, 2023.
[2] S. Arhin, "Tax relief awareness: Its financial consequences to non-filers in Ghana," *SSRN Electron. J.*, Jan. 2019.
[3] B. C. Ayers, O. Z. Li, and J. R. Robinson, "Tax-induced trading around the taxpayer relief act of 1997," *J. Amer. Taxation Assoc.*, vol. 30, no. 1, pp. 77–100, Mar. 2008.
[4] A. W. Bartik, M. Bertrand, Z. Cullen, E. L. Glaeser, M. Luca, and C. Stanton, "The impact of COVID-19 on small business outcomes and expectations," *Proc. Nat. Acad. Sci. USA*, vol. 117, no. 30, pp. 17656–17666, Jul. 2020. i
[5] E. Bikas and L. Jurevičiūtė, "Impact of tax relief on public finance," *Econ. Culture*, vol. 13, no. 2, pp. 14–22, Dec. 2016.

[6] S. Bond, K. Denny, and M. Devereux, "Capital allowances and the impact of corporation tax on investment in the UK," *Fiscal Stud.*, vol. 14, no. 2, pp. 1–14, May 1993.
[7] L. Cao, "AI in finance: Challenges, techniques and opportunities," *SSRN Electron. J.*, Feb. 2021.
[8] H. Cho and S. Yoon, "Investment asset allocation in response to tax relief for mutual funds: The case of South Korea," *Investment Manage. Financial Innov.*, vol. 18, no. 3, pp. 347–358, Sep. 2021.
[9] C. Curd, N. Masters, and D. Gibson, "Handbook of capital allowance assessment," Veritas Advisory, London, U.K., Tech. Rep., 2021.
[10] Z. Dai, E. L. Maydew, H. H. Zhang, and D. A. Shackelford, "Capital gains taxes and asset prices: Capitalization or lock-in?" *J. Finance*, vol. 63, no. 2, pp. 709–742, Oct. 2008.
[11] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.
[12] E. D. Liddy, "Natural language processing," in *Encyclopedia of Library and Information Science*, 2nd ed. Boca Raton, FL, USA: CRC Press, 2001.
[13] E. Gazoni and C. Clark, "Openpyxl—A Python library to read/write excel 2010 xlsx/xlsm files," Tech. Rep., 2010. [Online]. Available: https://openpyxl.readthedocs.io/en/stable/
[14] I. E. Fisher, M. R. Garnsey, and M. E. Hughes, "Natural language processing in accounting, auditing and finance: A synthesis of the literature with a roadmap for future research," *Intell. Syst. Accounting, Finance Manage.*, vol. 23, no. 3, pp. 157–214, Jul. 2016.
[15] G. Yong, W. Xin, and W. Zhenlei, "Fault detection for a class of industrial processes based on recursive multiple models," *Neurocomputing*, vol. 169, pp. 430–438, Dec. 2015.
[16] N. D. Goumagias, D. Hristu-Varsakelis, and Y. M. Assael, "Using deep Q-learning to understand the tax evasion behavior of risk-averse firms," *Expert Syst. Appl.*, vol. 101, pp. 258–270, Jul. 2018.
[17] M. A. Górecki and N. Letki, "Social norms moderate the effect of tax system on tax evasion: Evidence from a large-scale survey experiment," *J. Bus. Ethics*, vol. 172, no. 4, pp. 727–746, Sep. 2021.
[18] E. Kentala, M. Juhola, Y. Auramo, and I. Pyykkö, "Comparison between diagnoses of human experts and a neurotologic expert system," *Ann. Otology, Rhinol. Laryngol.*, vol. 107, no. 2, pp. 135–140, Feb. 1998.
[19] *Open-Source Library for Artificial Neural Networks*, Keras, Mountain View, CA, USA, 2023.
[20] M. V. Koroteev, "BERT: A review of applications in natural language processing and understanding," 2021, *arXiv:2103.11943*.
[21] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," 2019, *arXiv:1910.01108*.
[22] J. N. Kuria, D. B. Omboi, and D. G. Achoki, "Effect of capital allowance incentive on the performance of EPZ firms in Kenya," *Int. J. Finance Accounting*, vol. 2, no. 5, pp. 1–14, Aug. 2017.
[23] Y. Y. Lim and A. P. Teoh, "Realizing the strategic impact of business intelligence utilization," *Strategic Direction*, vol. 36, no. 4, pp. 7–9, Mar. 2020.
[24] R. H. Michaelsen, "An expert system for federal tax planning," *Expert Syst.*, vol. 1, no. 2, pp. 149–167, Oct. 1984.
[25] F. Olan, E. O. Arakpogun, U. Jayawickrama, J. Suklan, and S. Liu, "Sustainable supply chain finance and supply networks: The role of artificial intelligence," *IEEE Trans. Eng. Manag.*, 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9669168
[26] J. Pointon, "Investment and risk: The effect of capital allowances," *Accounting Bus. Res.*, vol. 10, no. 40, pp. 432–439, Sep. 1980.
[27] S. Putri, A. Alamsyah, and I. Akhlis, "Implementation of expert system to diagnose pregnancy disorders using fuzzy expert system method," *J. Adv. Inf. Syst. Technol.*, vol. 3, no. 1, pp. 33–42, Apr. 2021.
[28] S. Quaglini, M. Stefanelli, G. Barosi, and A. Berzuini, "A performance evaluation of the expert system ANEMIA," *Comput. Biomed. Res.*, vol. 21, no. 4, pp. 307–323, Aug. 1988.
[29] A. Rathi, S. Sharma, G. Lodha, and M. Srivastava, "A study on application of artificial intelligence and machine learning in Indian taxation system," *Psychol. Educ. J.*, vol. 58, no. 2, pp. 1226–1233, Feb. 2021.
[30] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter," 2019, *arXiv:1910.01108*.
[31] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
[32] J. Shao, Z. Lou, C. Wang, J. Mao, and A. Ye, "The impact of artificial intelligence (AI) finance on financing constraints of non-SOE firms in emerging markets," *Int. J. Emerg. Markets*, vol. 17, no. 4, pp. 930–944, May 2022.
[33] B. Shi, B. Dong, Y. Xu, J. Wang, Y. Wang, and Q. Zheng, "An edge feature aware heterogeneous graph neural network model to support tax evasion detection," *Expert Syst. Appl.*, vol. 213, Mar. 2023, Art. no. 118903.
[34] D. Shpilberg, L. E. Graham, and H. Schatz, "ExperTAXsm: An expert system for corporate tax planing," *Expert Syst.*, vol. 3, no. 3, pp. 136–151, Jul. 1986.
[35] Y. Shynkevich, T. M. McGinnity, S. A. Coleman, A. Belatreche, and Y. Li, "Forecasting price movements using technical indicators: Investigating the impact of varying input window length," *Neurocomputing*, vol. 264, pp. 71–88, Nov. 2017.
[36] L. Sánchez, A. Santos-Olmo, E. Álvarez, M. Huerta, S. Camacho, and E. Fernández-Medina, "Development of an expert system for the evaluation of students' curricula on the basis of competencies," *Future Internet*, vol. 8, no. 4, p. 22, May 2016.
[37] K. P. Venkatesh, M. M. Raza, J. A. Diao, and J. C. Kvedar, "Leveraging reimbursement strategies to guide value-based adoption and utilization of medical AI," *NPJ Digit. Med.*, vol. 5, no. 1, Aug. 2022. [Online]. Available: https://www.nature.com/articles/s41746-022-00662-1
[38] A. Verdaguer, A. Patak, J. J. Sancho, C. Sierra, and F. Sanz, "Validation of the medical expert system PNEUMON-IA," *Comput. Biomed. Res.*, vol. 25, no. 6, pp. 511–526, Dec. 1992.
[39] N. Webersinke, "Natural language processing meets accounting and finance: Review and performance comparison of textual analysis approaches," *SSRN Electron. J.*, Jul. 2023. [Online]. Available: https://ssrn.com/abstract=4527724
[40] X. Xu, "The influence of artificial intelligence on the financial industry," in *Proc. 4th Int. Conf. Econ. Manage. Cultural Ind. (ICEMCI)*, Dec. 2022, pp. 393–400.

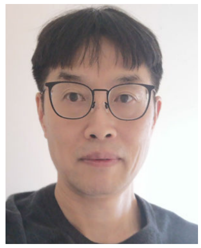
JAVAD GHOLIZADEH received the bachelor's degree in physics from the University of Tabriz, Iran, and the master's degree in mathematical finance from the University of Birmingham. With a strong background in mathematics and statistics, he has engaged in practical projects involving the analysis of financial time series data and applied machine learning to various real-world scenarios. In 2022, he joined Veritas Advisory Ltd., a tax consulting firm specializing in capital allowances and tax relief for real estate industry. He initiated a project to automate tax relief consultancy in collaboration with Brunel University of London, as part of a government-funded initiative. Employing his skills in Python programming, machine learning, and data science, he applied his expertise to this endeavor. His primary research interests include text data analysis through natural language processing (NLP), deep learning utilizing recurrent neural networks, and the application of large language models at the enterprise level.

KWANG-SUNG CHUN has trained in planning and real estate with The Bartlett of University College London (UCL). He started his career as a Golf Course Architect and an Urban Designer of the Professional Golfers' Association (PGA), where he received over ten years of experience in primary planning, architecture, developments, and investments for large-scale projects across the world. He is a Consultant with Veritas Advisory, specializing in East Asian market, where he was a Chartered Surveyor in taxation allowances, for over seven years. Recently, he is the Managing Director with the Black Buffalo Group. He has been advising some of the major institutional investors in South Korea, such as securities houses and corporate banks on asset management and new capital investments.

CLIVE CURD currently acts for a variety of property investors and corporate clients on all types of property from data centers and dairy factories, to football stadia and student accommodation, and the more typical commercial property. With over 25 years of experience, he has successfully created asset allocation guides for clients, to be used by both tax and non-tax specialist users to correctly depreciate property assets. His extensive experience negotiating with HMRC and understanding claims and their scrutiny by HMRC is key to maximizing claims on behalf of clients. He has also been involved with the tax depreciation systems employed by many countries around the world, liaising with local tax consultants to formulate policy for companies that operate in many countries. He also advises the Royal Institution of Chartered Surveyor on the guides to be becoming a Chartered Taxation Surveyor, and assessing and chairing the panel for the assessment of professional competence.

NOLAN MASTERS qualified as a Chartered Quantity Surveyor, before joining Deloitte's Real Estate Team and became tax trained, before specialized in Capital Allowances. He has over 20 years of experience advising clients on all aspects of Capital Allowances. Over that time, he has worked with a varied client base, responsible for delivering millions of pounds of tax relief across a full spectrum of projects, including advising on two of the largest shopping center transactions and providing advice across a £300m property portfolio, which in both cases led to significant tax savings being generated for both offshore vehicles. One of his key approaches is to emphasize the importance of considering early tax planning on both developments and acquisitions, often working with lawyers and clients on developing strategies to secure the best Capital Allowances position available.

DAVID GIBSON has been advising on Capital Allowances, since 2000, covering property developments and acquisitions across all sectors with a particular emphasis on medical, industrial, office, hotels, and portfolio reviews. Projects advised on range from £300m single asset acquisitions and £2bn portfolios on behalf of institutions, REITS, private equity firms, and overseas investors to £50,000 refurbishments on behalf of an individual investor. His specialism is creating innovative ways to improve efficiencies in the claim process to meet tight client deadlines and improve future capital allowances reporting to HMRC. He has developed the first Capital Allowances document in the country to fulfil the clients reporting requirements within an Authorized Contractual Scheme (ACS), taking into account all new property additions, disposals, and future capex spends on a daily basis for each unit holder. He regularly presents on Capital Allowances at RICS and in-house to lawyers, Jersey Trusts, investment agents, and clients. He is also in regular contact with HMRC on current and future legislation changes and issues.



YONGMIN LI (Senior Member, IEEE) received the B.Eng. and M.Eng. degrees from Tsinghua University, China, and the Ph.D. degree from the Queen Mary University of London. Before joining Brunel University London, he was a Research Scientist with the British Telecom Laboratories. His research interests include data science, machine learning, artificial intelligence, image processing, computer vision, video analysis, medical imaging, bio-imaging, biomedical engineering, healthcare technologies, automatic control, and nonlinear filtering. He is a Senior Fellow of the Higher Education Academy. Together with his colleagues, he has won the Most Influential Paper over the Decade Award at MVA 2019; and the Best Paper Awards at Bioimaging 2018, HIS 2012, BMVC 2007, BMVC 2001, and RATFG 2001.