Journal of Applied Research and Technology

www.jart.icat.unam.mx

Journal of Applied Research and Technology 23 (2025) 82-93

Original

Multi-label multi-class text classification-enhanced attention in transformers with knowledge distillation

U. Jain[a]* • P. Mishra[a] • A. Dash[a] • A. Pandey[b]*

[a]*School of Computer Engineering, KIIT Deemed to be University, Odisha, India*
[b]*Mechatronics Laboratory, School of Mechanical Engineering, KIIT Deemed to be University, Odisha, India*

Received 03 05 2024; accepted 09 26 2024
Available 02 28 2025

Abstract: This scholarly paper introduces an innovative and comprehensive ideology that aims to significantly expand the utility of named entity recognition (NER) through the application of transformers in various natural language processing (NLP) tasks. One prominent task that necessitates attention is the intricate classification of emails into multiple labels, wherein each label can be associated with not just one but potentially multiple independent classes. Despite the existence of several research methodologies attempting to address numerous challenges in this domain, the industry continues to face a substantial hurdle when it comes to accurately categorizing multi-label texts like financial emails, which can encompass diverse categories such as Payment Information, Invoice Information, Disputes, and more. Considering these challenges, our proposed methodology serves as a breakthrough solution, demonstrating remarkable performance in the classification task across a wide range of datasets, including financial email and consumer complaint datasets. By leveraging the power of advanced transformers, we have achieved an exceptional accuracy rate of 94% for full match of the multi-label classes, while the accuracy for partial match to individual classes soared to an impressive 97%. This achievement not only highlights the effectiveness of the proposed approach but also showcases its potential to enhance the efficiency and reliability of NER applications in practical settings.

Keywords: Transformers, natural language processing, machine learning, multi-label text classification, deep learning, named entity recognition.

*Corresponding author.
E-mail address: anish06353@gmail.com (A. Pandey).
Peer Review under the responsibility of Universidad Nacional Autónoma de México.

1. Introduction

In the field of natural language processing (NLP), the classification task has expanded beyond sentiment analysis and multi-class classification in recent times. While sentiment analysis remains a valuable tool for assessing the emotional sentiment expressed in written content, its applications have broadened significantly. By assigning a numerical score within a range of -1 to +1, sentiment analysis allows businesses to determine whether individuals hold negative, neutral, or positive views on specific events. This technique finds utility across various domains, including Social Media Monitoring, Customer Support, and Product Analysis. Moreover, multi-class text classification has emerged as a crucial aspect of NLP. Unlike binary classification, multi-class classification involves categorizing textual data into more than two classes or categories. In this task, each data sample is assigned to a single class without the possibility of belonging to multiple classes simultaneously. To illustrate, consider the scenario where a model is trained to classify news headlines into distinct news categories such as business, sports, technology, entertainment, and politics. This allows for better organization and retrieval of information, enhancing the efficiency of news dissemination and content filtering. As NLP continues to advance, the scope of classification tasks expands to address diverse challenges in text analysis. By developing more sophisticated algorithms and leveraging larger datasets, researchers and practitioners aim to improve the accuracy and applicability of classification techniques, ultimately benefiting various industries and domains that rely on effective text classification.

But the question arises what if the text belongs to multiple classes simultaneously where the classes are mutually independent of each other? The existing methods using Word2vec (Mikolov et al., 2013) and Bag of Words have failed to establish a state-of-the-art (SOTA) mechanism for the task. The reason behind that is the lack of understanding of the relation between the context of the text present in sentences or paragraphs. In recent times (Chen et al., 2017), the automation industries working on NLP tasks have faced this issue, and a viable solution is yet to be found.

The research presented in this article aims to solve the problem of multi-label text classification by introducing enhanced attention with NER (Finkel et al., 2005) inside the transformer's mechanism. The presented architecture uses the power of attention in encode-decoder models and maps the classes obtained via NER to obtain an efficient classification of long multi-label texts such as emails and consumer complaints. The architecture achieves a SOTA performance on the given task and establishes an industry standard reputation towards the task. Figure 1 shows the overall flow diagram of the whole process.

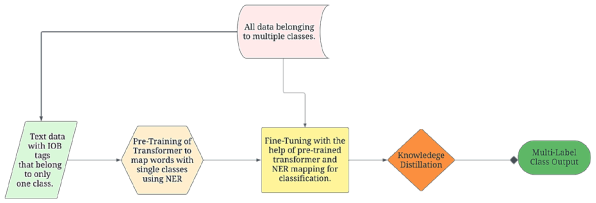


Figure 1. The flow diagram of the whole process proposed in this paper.

The process begins by pre-training a transformer model on text belonging to only a single class. This pre-training is done for the NER task, which will map the specific keyword unique to a class to that class. This mapping information is useful and, in fact, a very efficient approach to train, or even fine-tune, a transformer for the classification task because if multiple keywords from different classes are present in a single text, the model trained on the mapping, or the NER will classify those texts to multiple classes simultaneously. The knowledge distillation technique has been applied to reduce the complexity of the model and reduce the inference time.

2. Literature survey

The literature survey reveals a growing challenge in the field of natural language processing (NLP) related to the classification of text that can belong to multiple classes simultaneously. While existing methods, such as Word2vec and Bag of Words, have been widely used for text classification, they have failed to establish a state-of-the-art mechanism for this specific task. The main reason behind their limitations is the lack of understanding of the contextual relationships within sentences or paragraphs of the text. This issue has become particularly relevant in the automation industries that heavily rely on NLP tasks, where finding a viable solution is still an ongoing endeavor. Addressing this challenge requires advancements in algorithms and techniques that can capture the intricate dependencies and relationships within the text's context, paving the way for more accurate and effective multi-class text classification approaches. By tackling this problem, researchers and practitioners aim to improve the performance and applicability of NLP methods, thereby benefiting various industries that depend on sophisticated text classification for their operations.

The early efforts toward multi-label text classification were based on bag-of-words and graphs (Diera et al., 2022), but due to a lack of attention mechanism, they were not efficient for large texts or even a paragraph. Their efficiency was limited to sentence classification. A unique metadata-based learning approach (Zhang et al., 2022) was introduced but lacked efficiency for the full match. This method performed well for partial match of the multiple classes. However, given the

length of texts in the dataset for which the experiment was performed, it was still efficient.

With the advancement in deep learning methods, such as the Transfer Learning approach, this problem was addressed by a few researchers using the technique. One of them was based on a method known as 'Hierarchical Transfer Learning' (Banerjee et al., 2019), which combined multiple binary classifiers for each class, which performed significantly better than GRU (Cho et al., 2014) and turned out to be one of the most efficient architectures without deep learning. However, as the authors themselves have presented in the results that their Macro-F1 was not so significant, which necessitated the use of deep learning-based methods to achieve benchmark results on the task. A few more approaches without deep learning were employed for sentiment classification (Bhadra et al., 2023) and using advanced NLP techniques, such as key phase extraction (Dash et al., 2023) and simple DNNs (Prasad et al., 2023), which were identical in terms of efficiency. More development in entity extraction (Dash et al., 2019) broadened the possibility of efficient models and their use cases in other domains, such as biomedical NER (Suman et al., 2021), which was phenomenal work toward the task.

After the groundbreaking discovery of the attention mechanism by Vaswani et al. (2017) in transformers, BERT (Bidirectional Encoder Representations from Transformers) was introduced by Devlin et al. (2018), revolutionizing numerous NLP tasks in both academic and industrial domains. Subsequently, knowledge distillation, proposed by Hinton et al. (2015), facilitated parameter reduction in the BERT model. The resulting DistillBERT, as presented by Sanh et al. (2019), achieved state-of-the-art (SOTA) performance when fine-tuned on various NLP tasks, including Named Entity Recognition (NER) and classification. Notably, DistillBERT demonstrated exceptional performance while utilizing 40% fewer parameters compared to the original BERT model. The introduction of such efficient mechanisms motivated researchers to apply them for multi-label classification tasks where they fine-tuned (Zhang et al., 2021) the pre-trained transformer models and applied sampling methods (Jiang et al., 2021) over transformer layers to achieve decent efficiency on the task. A very similar approach to NER was joint learning from label embedding (Chang et al., 2020) and the correlation between them. The correlating mechanism was a time-efficient approach but lacked the accuracy that the attention mechanism provided in the transformers. Significant advancements using transformer mechanisms include TransUnet (Mishra et al., 2023) which also leveraged CNN. The implementation of different machine learning algorithms for classification and training purpose were discussed in articles (LK et al., 2023; Surjandari et al., 2023). Similarly, some more applications of machine learning and deep learning can also be found in various studies (Atikah et al., 2023; Heni et al., 2023).

The research presented in this paper provides an efficient SOTA approach toward multi-label text classification with multiple simultaneous classes. The method of first extracting keywords specific to individual classes and then mapping them with each class in a multi-class classification task provides an upper edge over the algorithms, which involves only attention. The approach presented in this research performs better even on large texts such as long financial emails having keywords belonging to multi-class simultaneously. The NER-extracted keywords provide an added advantage to the attention algorithm, which helps the model classify multi-label texts with 94% full match accuracy and 97% partial match accuracy. The final process of knowledge distillation reduces the model's complexity and, hence, its the inference time. The model's results have been compared with other architectures and methodologies in the Results section.

3. Proposed approach

The task involves multi-label text classification where multiple classes can be positive simultaneously. The proposed approach uses NER for pre-training a transformer model and then fine-tuning a transformer model such as BERT or DistillBERT for the classification task. The overall flow requires data preparation and defining the model proceeded by training, knowledge distillation, and finally, the model evaluation. The architecture is based on a full encoder-decoder structure. This configuration enables comprehensive context capture by leveraging both the backward and forward contextual information, which is essential for the nuanced demands of NER and multilabel text classification. A simple web application has been demonstrated for the prediction task.

3.1. Data preparation

Unlike other tasks, where input is just plain text, the NER task requires input in a certain format. For example, to recognize the word 'INV0011', which is an invoice number in a financial email text, one has to tag the word to a particular class, here 'Request for Invoice.' The most commonly used format for NER data is the BIO (Beginning, Inside, Outside) or IOB (Inside, Outside, Beginning) format. In this model, the IOB format is used for the preparation of the data, where each word in the dataset is assigned a label that indicates its entity type. For example, consider the following text from a financial email dataset:

"Dear Vendor,
Invoice INV00001 was paid with cheque CH890 on 8/9/20."

The resulting labels are - 'INV00001' as 'Invoice', 'CH890' as 'Cheque', and 8/9/20 as 'Date'. Hence after labelling with IOB format where the 'B-' prefix denotes the beginning of an entity, and since there are no nested entities in the provided text, there are no 'I-' (Inside) tags present, 'Invoice' (INV00001) is labelled as B-Invoice, 'Cheque' (CH890) is labelled as B-Cheque and 'Date' (8/9/20) is labelled as B-Date. All other words that are not part of the labelled entities are tagged as 'O' (Outside).

There are a total of 4 classes in the financial email dataset (Request for Invoice, Confirmation of Payment, Dispute, and Other) and three classes in the Consumer complaints dataset (Consumer Financial Protection Bureau, 2020) (Service Related, Product Related, and Payment Related). The multilabel classes in the dataset have been represented as one-hot encodings inside a list. For example, a text belonging to the classes "Request for Invoice" and "Payment Confirmed" simultaneously will be labelled as [1,0,1,0]. Similarly, a text from the consumer complaints dataset belonging to classes Service Related and Product Related simultaneously belongs to the label [1,1,0]. A POS tagger has been used to label the datasets manually for the entity relation mapping as part of the NER task. The dataset has been cleaned for greeting and salutation text along with the html tags for the multiclass classification task.

3.2. Pre-training the transformer for NER

The task of pre-training a transformer-based model on the IOB labelled datasets to learn general language representations. This pre-training step allows the model to capture rich contextual information and linguistic patterns, which will later be fine-tuned for the multi-label classification scenario. Figure 2 illustrates the architecture of the whole pre-training process.

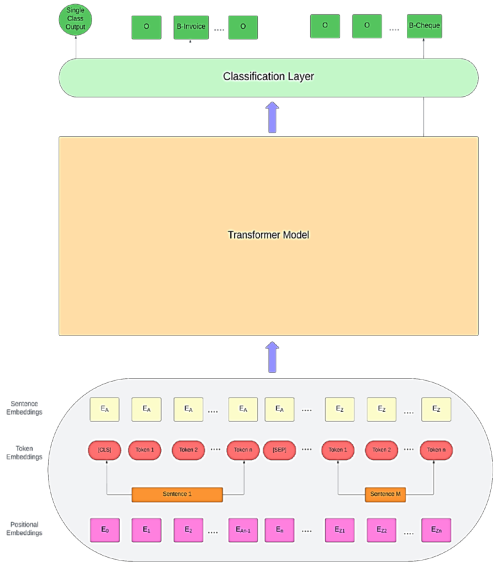


Figure 2. The model architecture for the NER pre-training task.

The dataset is cleaned for punctuation, stopwords, and HTML tags as part of the pre-processing step. It is an essential preprocessing step to improve the quality and consistency of the data. Removing stopwords can reduce noise in the data and improve computational efficiency. The next step is to break down the sentences into words or tokens. This step allows for further processing and analysis at the word level. The tokenized text is converted into token embeddings, which is the numerical representation for the model to train. These token embeddings are augmented with the corresponding positional embeddings, enabling the model to understand the sequential order of words in a sentence or input sequence. Positional embeddings are fixed-size vectors that are added to the word embeddings of each token in the input sequence. These vectors encode the relative or absolute position of the tokens within the sequence and are generated using sine and cosine functions with different frequencies. The positional encodings are implemented using a sine and cosine function approach with a lambda parameter set to 0.01. This λ value determines the wavelength of the sinusoidal functions, thereby affecting the encoding's ability to model positional relationships at various sequence lengths. Given the position i in the input sequence of length N in the embedding dimension D, then for each dimension j in the positional embedding vector:

$$P_{ij} = sin(\frac{i}{\lambda^{(2k/D)}}) \quad (1)$$

The above equation holds true for every even-indexed dimension where $j = 2k$ and λ is a hyperparameter that determines the scale of the positional encoding. For odd-indexed dimensions where $j = 2k + 1$.

$$P_{ij} = cos(\frac{i}{\lambda^{(2k/D)}}) \quad (2)$$

By incorporating these equations, the transformer model can generate unique positional embeddings for each token in the sequence, enabling the model to differentiate words based on their position and capture sequential dependencies effectively.

The next step is to train the transformer model using the input embeddings. This can be done either by using a pre-trained architecture and fine-tuning it or by training the transformer from scratch. For this task, the output from the NER transformer must be converted to the corresponding one-hot labels of the classes to which they belong.

3.3. Fine-Tuning with an enhanced attention transformer model

A transformer is a neural network capable of sequentially processing data of a diverse nature, including text and audio, and now extended to videos and images. Unlike CNN, which

uses convolution operation, transformers have a special mechanism known as attention. They also have basic neural network layers such as normalization, feed-forward, embedding, and positional encoding layers. In this research, an extra layer has been added, which contains sequence-wise outputs from the NER transformer. This allows the model to map the whole text to the correct individual classes by using the information of the keyword-classes mappings extracted from the NER transformer. The encoder-decoder structure allows the model to capture the contexts from both sides of the sentence rather than only a forward pass capture. The input is processed by the encoder, and the output is generated by the decoder module. The initial task of the transformers was for text generation, but they can be fine-tuned for specific tasks such as classification and entity extraction. The outbreak of Large Language Models after BERT, such as generative pre-trained transformers (GPTs) (Radford et al., 2018), has improved the efficiency of various natural language tasks. However, complex tasks such as multi-label text classification and NER are still far-fetched in terms of efficiency for these models.

The principal component of a transformer is attention, which allows the model to focus on meaningful information from the input text. It has the capability to translate a sentence in the right order based on the context of the tokens. The attention function mainly takes three inputs - queries (Q), keys (K), and values (V), which are the vectors from the input text to the model. In this paper, those inputs are different from the traditional inputs of the attention layer. The query vectors come from the input text, whereas the key vectors are sourced from the NER Transformer previously trained. These key vectors represent the tagged entities recognized in the text, encoded to include both the tag type and positional information, thus preserving the spatial and categorical data essential for accurate entity recognition. It is the entity that is related to multi-modal architecture. The embeddings used for the NER labels are consistent with those employed in the initial training of the NER Transformer, ensuring that each entity type is distinctly represented. Additionally, positional encodings are applied to these embeddings before inputting them as key vectors into the attention mechanism. The value vectors are formulated based on the interactions between Q and K vectors, embodying the model's learned response to the identified entities and their contexts. If we consider the dimension of these vectors to be d_k, then:

$$Attention = Softmax(\frac{QK^T}{\sqrt{d_K}}) \cdot V \quad (3)$$

The transformer model contains 12 attention heads in each encoder and decoder layer. This multiplicity of heads allows for a broader range of attention to different parts of the input sequence, which is crucial for capturing the varied dependencies in a multi-label setting. But this equation has a quadratic computational complexity. This reduces the model response time and increases the training time, even for small datasets. To overcome this, this paper proposes enhanced attention, which is memory-aware and does not trade the trainable parameters. The number of input and output operations are optimized, thus reducing the access of GPU to a free number of times during training. The multiplication of a large N × N matrix slows down the GPU. The proposed attention mechanism first splits the vectors Q, K, and V and loads them into SRAM, which is faster than the GPU core. The attention calculation is done here with respect to the split blocks, one block at a time, and then the result is combined. The calculation involves iterating over the Q vector for each K and V vector of the block for computing the immediate or temporary values inside the SRAM only. So, instead of storing all three matrices, only the output is stored for the backpropagation, where gradients come into play. This can be termed as a checkpointing of the calculated gradients, which is selective in nature. Due to the reduced GPU access and most of the calculation being done in SRAM of the GPU, the selective nature helps towards fast processing of backpropagation and gradient calculation. The final result of the attention calculation is then written back to the GPU cores, hence preventing large calculations there. Only the vectors are being transferred to the core after and before calculation, thus preventing the repeated to and fro writing of inputs and temporary variables. A comparison of SRAM and GPU core capabilities, along with the response time of conventional attention vs. enhanced attention, is shown in Figure 3 below.

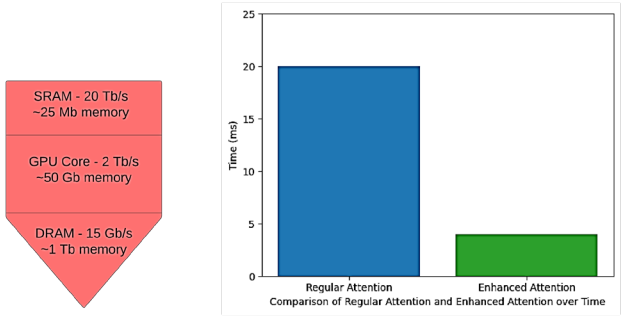


Figure 3. Comparison of regular vs. enhanced attention and overview of GPU capabilities.

Within the encoder, one can find multiple sublayers that come into play right after attention. These sublayers include layer normalization and multi-layer perceptrons. Elements like dropouts and residual layers contribute to the encoder's capabilities. Now, let's delve into the proposed fine-tuning transformer. This model has a total of 12 layers, which are divided between the encoder and decoder. This is quite a step up from the standard transformers, which usually have just 6 layers. The extra layers here bring a significant enhancement.

They make it possible for the model to pay even more attention to different parts of the input, allowing it to capture the global context of sentences quite effectively. This enhanced ability to understand the broader context greatly benefits the model's performance, particularly in tasks like multi-class classification. So, the increased depth in our transformer is a key factor in its success. The extra layers help to focus on the keywords extracted from the NER model, which are injected after successive layers and help to determine the class.

The decoder is quite similar to the encoder, with just a few extra elements. One key addition is the multi-head attention mechanism, which plays a crucial role. This part takes the output from the encoder layer and combines it with the NER model output. This fusion is essential because it allows the decoder to understand how keywords relate to specific classes within the text.

Now, let's talk about the final layer. It is a fully connected dense layer, and it uses the sigmoid function for classification. This layer is like the decision-maker of the model, determining which class or category the input belongs to. This choice is beneficial in scenarios where each class is considered independent, as the sigmoid function treats each output neuron independently, allowing for multiple labels to be assigned to each input simultaneously. The model addresses a multi-label classification problem where each label is independent and not mutually exclusive. This scenario is distinct from multi-class classification tasks where a single output is selected from multiple categories, typically handled by a softmax activation function. The residual connections are like breadcrumbs that help the model find its way through the layers of the transformer. They carry information about the position of words in the text, which is crucial for understanding context. This makes the training process smoother and more effective.

To improve things, there is Layer Normalization after each attention and MLP layer. It's like a traffic cop for each layer's activations, ensuring they're at the right intensity. This helps speed up training and ensures that our model behaves consistently both during training and when it is making predictions. So, all these elements work together to make the model less latent.

4. Model evaluation and result analysis

This section discusses the proposed architecture's performance on two datasets. The model has been evaluated separately on the accuracy for determining the performance on a full match and on subset accuracy for determining the model correctness on partial matches of the classes.

4.1. Datasets used

Two datasets have been used for training and evaluation of the model. The first dataset is a private financial email dataset that has been masked to replace sensitive data (such as account numbers, cheque details, etc.) with similar random numbers or alphanumeric characters wherever applicable. This dataset has originally 500000 financial emails, which was augmented to a final number of 800000 emails. The dataset has five classes, the details about which are presented in Table 1 below.

Table 1. Description of the classes for the financial email dataset.

Class Name	Description
Payment Confirmed	Emails containing Cheque numbers, wire transfers, UTR IDs etc., along with invoice numbers for which payment was made for the purchase on certain dates for a certain amount.
Request for Invoice	Emails with invoice numbers or date ranges requesting invoice copies.
Account Statement	Emails with account numbers and date ranges request statements for those accounts.
Dispute	Emails containing natural language description and/or amount of dispute or asking for clarifications.
Others	Any other content apart from the abovementioned categories, such as automated portal replies, etc.

The second dataset is about consumer complaints (Consumer Financial Protection Bureau, 2020), which have natural language text about complaints of consumers towards financial services and products in the various sectors of finance, such as prepaid cards, loans, money transfers, etc. There are originally 18 categories in the original datasets, along with approximately 1300000 rows of complaints. Nevertheless, it has been noticed that certain classes can be found within others. For example, 'Credit card' and 'Prepaid card' fall under the 'Credit card or prepaid card' category. Hence, after filtration and data cleaning, the final dataset that was used for training and evaluation has 500000 consumer complaints within ten categories, namely - ‘Checking or savings accounts’, ‘Debt Collection’, ‘Credit Cards’, ‘Student loans’, ‘Vehicle loans’, ‘Payday loans’, ‘Consumer Loan’,’ Money transfers’, ‘Virtual currency’ and ‘Other financial services.’

4.2. Setup for training

The datasets have been cleaned for punctuation and special characters and have been passed through HateBert (Caselli et al., 2020) to remove any hate and abuse text. The financial

emails dataset has also been pre-processed to remove useless text such as greetings and salutations. The NER annotation for entities has been done according to the IOB format.

The training parameters include a batch size of 64, run for 50 epochs on a cluster of seven Nvidia 3080 GPUs. The checkpoints of every epoch are saved, and early stopping based on validation performance is implemented to avoid overfitting. The learning rate scheduler from TensorFlow has been implemented to adjust the learning rate after each epoch, starting with the initial learning rate of 0.1. The categorical cross-entropy loss function, along with the Adam optimizer, has been used to train the classification model. Adam is widely recognized for its efficiency in large datasets and deep learning models because of its adaptive learning rate capability, which helps converge faster and more effectively even in complex neural networks.

4.3. Evaluation criteria

The model has been evaluated on two criteria. A full match of the classes means that all the simultaneous classes of a text have been predicted correctly by the model. For example, if the original label is [0,1,0,0,1] and the model also predicts the exact same label, then it is a full match. The accuracy equation can directly calculate this ratio or percentage:

$$Full\ match\ accuracy\ = \frac{No.of\ rows\ predicted\ correctly\ exactly\ as\ it\ is}{Total\ number\ of\ rows} \quad (4)$$

The partial match of the classes refers to the scenario when not all classes are predicted correctly. For instance, if the actual label is [0,0,1,1,0] and the predicted label is [0,0,1,0,1], then it's a partial match since only one of the five classes is predicted correctly. The accuracy of the partial match can be calculated as follows:

$$Partal\ Match\ ratio\ of\ a\ class\ = \frac{Total\ number\ of\ rows\ predicted\ correctly\ for\ the\ class}{Total\ number\ of rows\ that\ belong\ to\ the\ class} \quad (5)$$

The accuracy would be a strict measure because even if a single class is misclassified it would be negative label for a particular text. To tackle this problem, the partial match has also been calculated.

The partial match of each class is calculated separately. So, if there is a dataset of 100 rows where a particular class is predicted correctly 5 times, and the total number of rows that belong to that class is 25, then the partial match ratio would be 5/25 = 0.2. This metric ensures that even if the model predicts a partial match, it is not classified as Negative despite some classes that are predicted correctly.

4.4. Model performance

The financial emails dataset has been tested on unseen data of 200000 emails while the consumer complaints dataset has been evaluated on 125000 unseen consumer complaints text. Table 2 shows the details about the exact match accuracy of the model on both datasets:

Table 2. Accuracy of the model for a full match on both the datasets.

Dataset	Total Evaluated Emails/ Complaints	Total Number of Fully matched text	Accuracy or Full match percentage
Consumer Complaints	125000	113,750	91%
Financial Emails	200000	188,000	94%

For the partial match, the accuracy of each class has been calculated separately for the datasets. The overall partial accuracy is the average of the accuracies of each individual class. Tables 3 and 4 show the performance of the model for partial accuracy on both datasets.

Table 3. Match percentage of each class prediction by model.

Class Name	Number of rows predicted correctly	Total rows for which the class is positive	Match percentage
Payment Confirmed	45,107	45,107	100
Request for Invoice	40,949	41,279	99.2
Account Statement	36,135	37,846	95.5
Dispute	35,403	38,524	91.9
Others	39,399	40,244	97.9

The reason why classes 'Payment Confirmed', and 'Request for Invoice' have been almost correctly classified is because they clear entities such as Invoice numbers and cheque numbers, which are detected correctly by the NER model, while the class 'Others' have totally different types of emails such as automated replies after a transaction, system emails or email by clients who have totally different subject in which no entities are present. Hence, they are clearly distinguishable from the rest of the classes and predicted with greater accuracy. The 'Account statement' has account numbers, the length of which varies by banks and customers, so the model predicts that class with

somewhat lower accuracy than the first two classes. On the other hand, the class 'Dispute' has natural language text, and only the amount or sometimes date entity is present; hence it is predicted with the lowest accuracy since it doesn't get much input from the NER model and is solely dependent on the fine-tuning process.

For the Consumer complaints dataset, a similar trend follows for classes like 'Debt Collection' and 'Credit Cards', for which there is more text than other classes. Hence, to prevent bias while training, class weights have been used. Since they have very little data, more weightages have been provided to classes such as 'Consumer Loan',' Money transfers', and 'Virtual currency'. Table 4 shows the partial match percentage of each class.

Table 4. Match percentage of each class prediction for the consumer complaint dataset.

Class Name	Number of rows predicted correctly	Total rows for which the class is positive	Match percentage
Checking or savings accounts	17,622	18,546	94.9%
Debt Collection	18,210	19,119	95.2%
Credit Cards	22,085	22,566	97.7%
Student loans	19,603	20,555	95.4%
Vehicle loans	9,414	10,129	92.8%
Payday loans	19,238	19,889	96.5%
Consumer Loan	5,596	6,123	91.3%
Money transfers	3,652	3,999	91.2%
Virtual currency	3,507	3,884	90.2%
Other financial services	2,110	2,365	89.1%

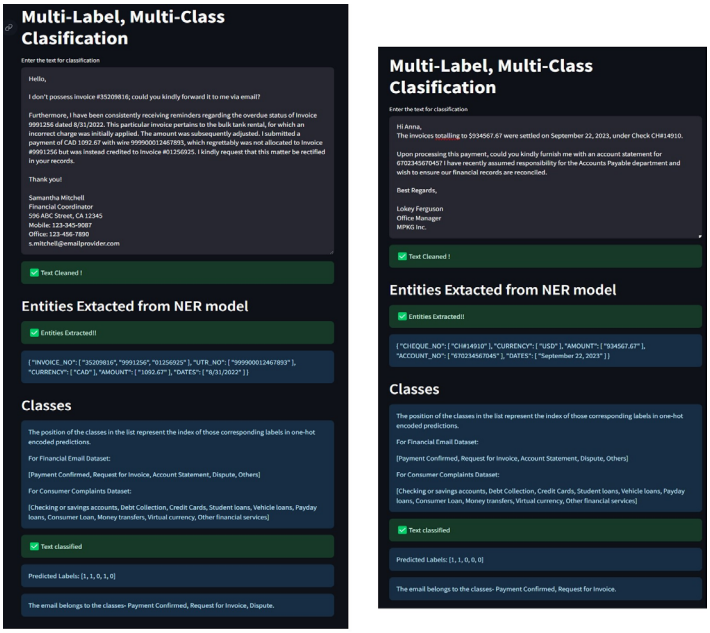
In a comprehensive analysis, a meticulous fine-tuning process was conducted on the datasets employing a decoder-only model, specifically BERT (Devlin et al., 2018), in isolation, devoid of any assistance from the NER transformer. Parallel evaluations were also conducted with a Large Language model, Llama (Touvron et al., 2023), to discern disparities in latency and accuracy. The outcomes of this comparative study revealed insightful findings: BERT exhibited notably favorable results for the multi-class classification task, underscoring the model's prowess. Conversely, Llama displayed shortcomings in accuracy and latency, warranting attention to these critical aspects. A detailed breakdown of the comparative results is presented in Table 5, offering valuable insights into the model performances.

Table 5. Comparison of the proposed models with others.

Model	Dataset	Full match accuracy	Partial Match accuracy	Average Latency (seconds)
Llama	Consumer Complaints	76%	80%	5.1
	Financial Emails	79%	84%	7.8
BERT	Consumer Complaints	82%	87%	3.6
	Financial Emails	88%	91%	4.4
Our Model	Consumer Complaints	91%	94%	3.2
	Financial Emails	94%	97%	3.5

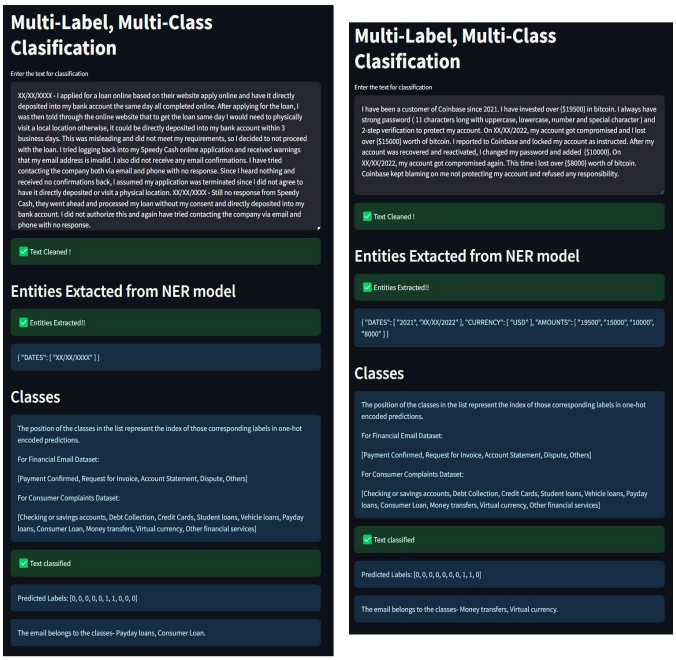
The reason for the low latency in the proposed model is solely the enhanced attention mechanism used in the transformers. The efficient utilization of the different memory components of the GPU for specific tasks around the input matrices enables better and faster model performance.

A streamlit app has been developed for the interactive inference of the model. Figure 4 (a) and (b) show some of the samples from the financial email dataset. The user interface (UI) shows the predicted labels and the extracted entities from the model. A legend has been provided to correctly understand the position of the classes from the predicted one-hot encoded labels. The entities detected have been post-processed before the final display. For example, the entity 'AMOUNT' has been separated from the currency. Figure 5 (a) and (b) represent some output predictions from the consumer complaint dataset.



(a) (b)

Figure 4. Sample outputs on the financial email dataset.



(a) (b)

Figure 5. Prediction samples from the consumer complaint dataset.

5. Conclusions

The proposed multi-modal architecture represents a cutting-edge approach to natural language understanding, seamlessly combining various components to enhance text processing capabilities. At its core, this architecture integrates a pre-trained named entity recognition (NER) model with a fine-tuned transformer for classification, forming a robust framework for multi-label, multi-class classification tasks. One of the distinctive features of this architecture lies in its ability to extract entities using the NER model. By identifying and tagging entities within the text, the model enriches the understanding of the content, providing a foundation for more accurate classification. This entity recognition step is crucial, especially when dealing with complex text data containing diverse and contextually significant entities.

The architecture further leverages enhanced attention mechanisms embedded within both the pre-trained and fine-tuning transformers. These mechanisms enable the model to focus on the most informative parts of the input, facilitating efficient information flow and reducing computational overhead. It enhances the model's predictive accuracy and optimizes GPU memory usage, making it computationally efficient and capable of handling large datasets. A notable advantage of this architecture is its user-friendly interactive interface. Incorporating an intuitive graphical user interface (UI) streamlines the inference process, providing users with a smooth and highly detailed experience.

In conclusion, the proposed multi-modal architecture redefines the landscape of text classification by combining NER capabilities with enhanced and memory-aware attention mechanisms. This holistic approach enhances the accuracy and efficiency of multi-label, multi-class classification and ensures a low response time for the task, making it a powerful tool for various natural language understanding tasks.

Building upon the foundation laid by this multi-modal architecture, the integration of generative AI models, such as Large Language Models (LLMs), presents an exciting avenue for exploration. Additionally, further research can delve into fine-tuning strategies for LLMs in conjunction with entity recognition, paving the way for more advanced and nuanced text understanding. This convergence of generative AI and user-centric design principles holds the potential to redefine the landscape of natural language understanding, offering versatile applications across a broad spectrum of domains and industries.

Conflict of interest

The authors have no conflict of interest to declare.

Funding

The authors received no specific funding for this work.

References

Atikah, S., Oyas, W., & Cahyadi, A. I. (2023). A Trajectory Control for Bipedal Walking Robot Using Stochastic-Based Continuous Deep Reinforcement Learning. *Evergreen*, *10*(3), 1538-1548. https://doi.org/10.5109/7151701

Banerjee, S., Akkaya, C., Perez-Sorrosal, F., & Tsioutsiouliklis, K. (2019). Hierarchical transfer learning for multi-label text classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics* (pp. 6295-6300). https://doi.org/10.18653/v1/P19-1633

Bhadra, K., Dash, A., Darshana, S., Pandey, M., Rautaray, S. S., & Barik, R. K. (2023). Twitter Sentiment Analysis of COVID-19 In India: VADER Perspective. In *2023 International Conference on Communication, Circuits, and Systems (IC3S)* (pp. 1-6). IEEE. https://doi.org/10.1109/IC3S57698.2023.10169701

Caselli, T., Basile, V., Mitrović, J., & Granitzer, M. (2020). Hatebert: Retraining bert for abusive language detection in english. *arXiv preprint arXiv:2010.12472*. https://doi.org/10.48550/arXiv.2010.12472

Chen, G., Ye, D., Xing, Z., Chen, J., & Cambria, E. (2017). Ensemble application of convolutional and recurrent neural networks for multi-label text categorization. In *2017 International joint conference on neural networks (IJCNN)* (pp. 2377-2383). IEEE. https://doi.org/10.1109/IJCNN.2017.7966144

Chang, W. C., Yu, H. F., Zhong, K., Yang, Y., & Dhillon, I. S. (2020). Taming pretrained transformers for extreme multi-label text classification. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 3163-3171). https://doi.org/10.1145/3394486.3403368

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1724-1734.
https://doi.org/10.48550/arXiv.1406.1078

Consumer Financial Protection Bureau. (2020). Consumer Complaints Database. Retrieved from https://catalog.data.gov/dataset/consumer-complaint-database

Dash, A., Mohanty, A., & Ghosh, S. (2023). Advanced NLP Based Entity Key Phrase Extraction and Text-Based Similarity Measures in Hadoop Environment. In *2023 6th International Conference on Information Systems and Computer Networks (ISCON)* (pp. 1-6). IEEE.
https://doi.org/10.1109/ISCON57294.2023.10112121

Dash, A., Pandey, M., & Rautaray, S. (2019). Enhanced Entity Extraction Using Big Data Mechanics. In *International Conference on Advanced Computing Networking and Informatics: ICANI-2018* (pp. 57-67). Springer Singapore.
https://doi.org/10.1007/978-981-13-2673-8_8

Devlin, J., Chang, M. W., Lee, K., and Toutanova, K. (2018). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*.
https://doi.org/10.48550/arXiv.1810.04805

Diera, A., Lin, B. X., Khera, B., Meuser, T., Singhal, T., Galke, L., & Scherp, A. (2022). Bag-of-words vs. sequence vs. graph vs. hierarchy for single-and multi-label text classification.

Finkel, J. R., Grenager, T., & Manning, C. D. (2005). Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd annual meeting of the association for computational linguistics (ACL'05)* (pp. 363-370).
https://aclanthology.org/P05-1045.pdf

Hinton, G. (2015). Distilling the Knowledge in a Neural Network.
https://doi.org/10.48550/arXiv.1503.02531

Jiang, T., Wang, D., Sun, L., Yang, H., Zhao, Z., & Zhuang, F. (2021). Lightxml: Transformer with dynamic negative sampling for high-performance extreme multi-label text classification. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 35, No. 9, pp. 7987-7994).
https://doi.org/10.1609/aaai.v35i9.16974

LK, J. G., Maneengam, A., KV, P. K., & Alanya-Beltran, J. (2023). Design and Implementation of Machine Learning Modelling through Adaptive Hybrid Swarm Optimization Techniques for Machine Management.
https://doi.org/10.5109/6793672

Mishra, P., Shrivastava, M., Jain, U., Prasad, A. O., & Satapathy, S. C. (2023). Multi-attention TransUNet—a transformer approach for image description generation. In *International Conference on Frontiers of Intelligent Computing: Theory and Applications* (pp. 21-34). Singapore: Springer Nature Singapore.
https://doi.org/10.1007/978-981-99-6702-5_2

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, *26*.
https://doi.org/10.48550/arXiv.1310.4546

Prasad, A. O., Singh, M., Mishra, P. K., Srivastava, S., Banerjee, D., & Sahoo, A. K. (2023). Prediction of Covid-19 Disease using Machine-learning-based Models. In *Machine Learning for Healthcare Systems* (pp. 109-129). River Publishers.

Radford, A. (2018). Improving language understanding by generative pre-training.
https://gluebenchmark.com/leaderboard

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.
https://doi.org/10.48550/arXiv.1910.01108

Heni, S., Anggita, S. R., Hartono, F. R. P., & Tasyakuranti, A. N. (2023). Texture-Based Classification of Benign and Malignant Mammography Images using Weka Machine Learning: An Optimal Approach. *Evergreen*, *10*(3), 1570-1580.
https://doi.org/10.5109/7151705

Suman, S., Dash, A., & Rautaray, S. S. (2021). A Literature Survey on Biomedical Named Entity Recognition. *Advances in Power Systems and Energy Management: Select Proceedings of ETAEERE 2020*, 109-119.
https://doi.org/10.1007/978-981-15-7504-4_12

Surjandari, I., Rindrasari, R., & Dhini, A. (2023). Evaluation of Efficiency in Logistics Company: An Analysis of Last-Mile Delivery. *Evergreen*, *10*(2), 649-657.
https://doi.org/10.5109/6792811

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M. A., Lacroix, T., ... & Lample, G. (2023). Llama: Open and efficient foundation language models. https://doi.org/10.48550/arXiv.2302.13971

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, pp. 6000-6010.

Zhang, J., Chang, W. C., Yu, H. F., & Dhillon, I. (2021). Fast multi-resolution transformer fine-tuning for extreme multi-label text classification. *Advances in Neural Information Processing Systems*,

Zhang, Y., Shen, Z., Wu, C. H., Xie, B., Hao, J., Wang, Y. Y., ... & Han, J. (2022). Metadata-induced contrastive learning for zero-shot multi-label text classification. In *Proceedings of the ACM Web Conference 2022* (pp. 3162-3173). https://doi.org/10.1145/3485447.3512174