Leveraging Language Models for Document Type Classification in Low-Resource Afrikaans Archives

Eduan Kotzé[1][0000−0002−5572−4319], Burgert A. Senekal[1][0000−0002−1385−9258], and Walter Daelemans[2][0000−0002−9832−7890]

[1] Department of Computer Science and Informatics, University of the Free State, Bloemfontein, South Africa
kotzeje@ufs.ac.za, burgertsenekal@yahoo.co.uk
[2] CLiPS - Computational Linguistics Group, University of Antwerp, Antwerp, Belgium
walter.daelemans@uantwerpen.be

Abstract. Document type classification is essential for effective information retrieval and management within archival systems, particularly in low-resource languages like Afrikaans. This study examines the feasibility of utilising multilingual transformer-based language models for document classification within a South African archival context. We followed a basic linguistic approach to prepare Afrikaans text documents for classification into six categories: academic papers, media reports, books, interviews, book reviews, and theses or dissertations. We compare fine-tuned transformer models, hybrid models combining traditional classifiers with contextual embeddings, and a baseline SVM (TF-IDF) classifier, using stratified 5-fold cross-validation and a hard voting ensemble for robust evaluation. Our findings reveal that the SERENGETI transformer-based model outperformed other multilingual models, achieving a weighted F1 score of 0.964, while hybrid approaches performed competitively. However, the baseline SVM (TF-IDF) model outperformed all transformer and hybrid models, with a weighted F1 score of 0.978. This research demonstrates the potential and current limitations of neural language models and hybrid strategies for enhancing document classification in Afrikaans archival systems. If implemented, the classifier can improve indexing efforts and reduce pressure on archival personnel who handle over 5,000 new items annually.

Keywords: Document Classification · Archival Systems · BERT · XLM-V · SERENGETI · Hybrid models · Afrikaans · Low-resource languages.

1 Introduction

Archives face numerous challenges in the digital age. The volume of data that needs to be processed and preserved has grown exponentially, making it difficult

for archives to manage collections effectively. This, coupled with the fact that the data is continuously evolving, makes archival tasks even more challenging [26, p. 180]. Simultaneously, archives have to collect and index documents with fewer resources [22, p. 22].

Founded in 1973, the National Afrikaans Literary Museum and Research Centre (NALN) is the primary archive for Afrikaans literary material, but budget cuts and staff shortages have created a major backlog in its collection and indexing processes [4, 29]. This has made it difficult for NALN to keep up with the volume of published material, particularly with its manual metadata generation and indexing of over 5,000 new clipping items annually, impacting the quality of research [4, 29].

Automatic and semi-automatic metadata generation tools can help address these challenges [22, p. 23]. One application of machine learning in an archival context is identifying the document type, such as articles, reviews, interviews, and media reports. Classifying text documents has been successfully demonstrated using machine learning [28] and deep learning [19]. At the same time, deep learning has also been successfully applied in information retrieval of visual material in an archival context [32].

This paper investigates the feasibility of using automatic document classification to process large amounts of digital material in order to reduce the workload of staff at NALN. As a baseline, we compare a Support Vector Machine (SVM) paired with TF-IDF vectorisation against various pre-trained transformer-based models. These models include BERT, mBERT, DistilmBERT, XLM-RoBERTa, Afro-XLM-R, SERENGETI, and XLM-V. We evaluate the suitability of each model in classifying Afrikaans documents. Our research aims to assess these models effectiveness across the various labels using accuracy and F1 score [30].

The main scientific contributions of this paper are: (i) the creation of a labelled Afrikaans dataset (n=1,997) spanning six document types; and (ii) a systematic benchmark comparing traditional, transformer-based, and hybrid approaches to document-type classification. We also provide (iii) a practical integration blueprint by proposing an amended NALN archival pipeline with an ML classifier module, and methodological insights via stratified 5-fold cross-validation with a hard-voting ensemble across folds, demonstrating both the promise of multilingual language models and the continued competitiveness of traditional classifiers in low-resource archival settings.

2 Related Research

The automatic classification of textual documents is a well-studied natural language processing (NLP) problem [28]. In recent years, machine learning has seen increasing adoption in archival contexts for document classification [9, 26]. A body of work introduces frameworks for metadata extraction and subject indexing in digital archives [21, 14], but generally recommends machine-assisted rather than fully automatic deployment. Reviews by Hutchinson [15] and Colavizza et al. [6] outline how NLP and artificial intelligence (AI) are being adopted for

archival workflows, including appraisal, automation, and emerging digital practices. Other applied projects include AI-driven frameworks for public engagement [13], semantic metadata generation for historical archives [7], and document annotation in philosophy using hybrid ML techniques [5].

This study addresses the challenges of document classification in low-resource languages, specifically Afrikaans. Unlike previous research focused on high-resource languages with extensive datasets, our work utilises a limited Afrikaans corpus comprising newspaper clippings, academic articles, theses, and dissertations. This presents unique difficulties due to the morphological complexity of Afrikaans, which features a rich inflectional system and can significantly impact feature extraction and model performance.

Furthermore, the diverse nature of our corpus, with documents ranging from short news articles to lengthy academic works (1 to 500 pages), introduces complexities in text representation and classification. The wide variability in document length necessitates careful consideration of feature engineering and model selection to effectively capture the nuances of short and long texts. This variability also poses challenges to traditional classification methods that may be biased towards shorter documents, requiring us to explore and adapt techniques to accommodate the unique characteristics of our corpus.

3 Data

3.1 Data Source and Collection

Our training data consisted of a wide range of digital materials related to Afrikaans literature, including book reviews, theses and dissertations, academic papers, interviews, and other scholarly publications. Documents were gathered from academic journals such as *Stilet* and *Tydskrif vir Letterkunde*, online news sources such as *Maroela Media* and *Netwerk24*, discussion forums such as *LitNet*, and university websites. When reviews or media reports, such as those from Maroela Media or Netwerk24 were not already available in PDF they were preserved as PDFs during the collection phase of the project. Since our focus was on training a machine learning classifier and not making material available, and NALN has their own ways of addressing the latter, we did not obtain any specific permissions.

We organised the documents into six distinct groups: *academic papers, media reports, books, interviews, book reviews, and dissertations/theses*. These categories are based on the classification system used by Galloway and Roux [12], and are also adopted by NALN. Given the self-evident nature of these categories, the researchers were able to conduct the annotation process themselves. For this study, we collected a total of 1,997 PDF documents. Table 1 provides a detailed breakdown of the documents categorised by type.

Table 1: Distribution of labels in the dataset.

Label (Afrikaans)	Label (English)	Count
Artikels	Papers	510
Berigte	Reports	400
Boeke	Books	385
Onderhoude	Interviews	362
Resensies	Reviews	313
Verhandelinge	Dissertations	27
Total		**1997**

3.2 Data Preparation

The PDF documents were converted to text using PDFPlumber[3], an open-source Python library for extracting text from machine-generated and other PDF files. Data cleaning included removing all tags (i.e. <i> and <html>), punctuation, whitespace, and numeric characters using the Gensim open-source Python library [25]. For this study, words were not converted to lowercase to preserve information about named entities. Thereafter, we removed all non-Afrikaans tokens from the dataset using the English corpus provided by NLTK [3]. As a final step, we evaluated the impact of removing Afrikaans stop words, following the methodology outlined by Kotzé et al. [18, p. 2], and found improved performance in their absence. Consequently, all Afrikaans stop words were excluded from the corpus.

4 Pre-trained Language Models

The study examined several pre-trained language models that include African languages in their training data, especially focusing on Afrikaans. Table 2 indicates whether the models were pre-trained on Afrikaans, Dutch or other African languages, and summarises each model's architecture, number of parameters, and vocabulary size. Dutch is included due to its close linguistic relationship with Afrikaans; its presence in the pre-training data may facilitate cross-lingual transfer and partially compensate for Afrikaans' low-resource status.

- **BERT** [10]: A monolingual language model pre-trained on English corpora only (BooksCorpus and Wikipedia). Used as a baseline to evaluate how a model without multilingual or Afrikaans exposure performs on this task.
- **mBERT** [10]: A multilingual variant of BERT trained on Wikipedia text from 104 languages, including Dutch, Afrikaans, and three other African languages. Designed to handle a wide range of languages without requiring language-specific models.

[3] https://github.com/jsvine/pdfplumber

Table 2: Pre-trained Multilingual Language Models.

Model	Params (M)	Vocab Size	Afrikaans	Dutch	African Languages
$\text{BERT}_{\text{Base}}$	108M	28,996	✗	✗	✗
$\text{mBERT}_{\text{Base}}$	178M	119,547	✓	✓	✓
$\text{DistilmBERT}_{\text{Base}}$	135M	119,547	✓	✓	✓
$\text{XLM-RoBERTa}_{\text{Base}}$	278M	250,002	✓	✓	✓
$\text{Afro-XLM-R}_{\text{Base}}$	278M	250,002	✓	✗	✓
SERENGETI	278M	250,004	✓	✗	✓
$\text{XLM-V}_{\text{Base}}$	778M	901,629	✓	✓	✓

- **DistilmBERT** [27]: A distilled version of mBERT, retaining 97% of its capabilities while reducing model parameters by 40% and improving inference speed by 60%. Shares the same multilingual scope as mBERT.
- **XLM-RoBERTa (XLM-R)** [8]: A multilingual language model based on RoBERTa, trained on 2.5TB of CommonCrawl data from 94 languages, including Afrikaans, Dutch, and several other African languages.
- **Afro-XLM-R** [2]: A variant of the XLM-RoBERTa using continued pre-training on 17 high-resource African languages, which includes Afrikaans, as well as three widely spoken non-African languages: English, French, and Arabic. Dutch is excluded from its training data.
- **SERENGETI** [1]: A large-scale multilingual language model covering 517 African languages and the top 10 most widely spoken global languages. Afrikaans is included in the training data (but not Dutch). This study uses the XLM-R-based variant.
- **XLM-V** [20]: An advanced multilingual language model with a million-token vocabulary, trained on 2.5TB of CommonCrawl data. Its training data includes Afrikaans, Dutch, and various African languages.

5 Experiment and Results

This section presents the key findings and analysis derived from our study. We first report the performance of the baseline model, followed by results from our fine-tuned language models. Finally, we will report the results from a set of hybrid models. All models were fine-tuned and cross-validated on 80% of the dataset, with the remaining 20% reserved as a holdout test set. Due to the unbalanced class distribution, we used stratified cross-validation to maintain the imbalance within each fold. We used scikit-learn [24] for training the baseline SVM and Huggingface [31] with PyTorch [23] for fine-tuning the pre-trained language models. All experiments were performed on a desktop computer with a NVIDIA 24GB RTX-4090 GPU.

5.1 Dataset Resampling

During our initial fine-tuning experiments, we observed suboptimal weighted F1 scores (0.317) in the training and testing phases. This was primarily due to the limitations of BERT, DistilBERT, and XLM-RoBERTa tokenisers, which are restricted to processing a maximum of 512 tokens, resulting in significant data truncation. To address this issue, we developed a `Python` method to preprocess the documents by splitting each text entry into smaller segments of up to 512 words. We then restructured the dataframe to place each segment in a seperate row, utilising the `pandas.explode()` function. This approach is particularly beneficial for models and tokenisers constrained by maximum input sizes.

Due to the significant class imbalance in our dataset, most notably the over-representation of the *Dissertations* label (n = 23,564), we applied stratified sampling with a threshold of 4,000 instances per class. This approach allowed us to construct a more balanced subset of the data by limiting the number of samples from the over-represented classes, particularly *Dissertations*.

This strategy enhances computational efficiency by reducing the dataset size, which is crucial for training models effectively, as processing nearly 32,000 instances can be resource-intensive. Limiting the dataset size to 4,000 instances per class, expedites data processing and model training, making it feasible to work within our available resources. Table 3 presents the results for the resampled dataset, organised according to segments of 512 words.

Table 3: Distribution of labels during data preparation stages.

			Resampled		
Label	**Original**	**512-words**	Train	Test	Total
Papers (Artikels)	510	3,787	3,029	758	3,787
Reports (Berigte)	400	347	278	69	347
Books (Boeke)	385	2,277	1822	455	2,277
Interviews (Onderhoude)	362	923	738	185	923
Resensies (Reviews)	313	946	757	189	946
Dissertations (Verhandelinge)	27	23,564	3,200	800	4,000
Total	**1,997**	**31,844**	**9,824**	**2,456**	**12,280**

5.2 Performance Metrics

Our classification task was a multi-class problem, where each input document had to be assigned to one of six non-overlapping classes. Since traditional binary classification metrics do not directly generalise to the multi-class setting, it is essential to apply suitable averaging strategies, namely macro, micro, and weighted averaging, to compute performance measures such as precision, recall, and F1 score.

We selected *weighted averaging* as the primary evaluation strategy, as it accounts for class imbalance by weighting each class according to its frequency. All reported accuracy, precision, recall, and F1 scores in Sections 5.3 to 5.7 and the associated tables use this strategy (denoted with subscript w). For transparency, we also report macro-averaged F1 scores (subscript m) to provide equal insight into performance across all classes equally. The macro F1 score was additionally used during training to monitor validation performance for early stopping (see Section 5.4).

5.3 Baseline Model

A Support Vector Machine (SVM) model was implemented with the Python scikit-learn library [24] using unigram TF-IDF-weighted features and a linear SVM (LinearSVC; LIBLINEAR) [11] implementation. We selected SVM with TF-IDF as a strong baseline, consistent with prior Afrikaans studies showing that TF-IDF with a linear SVM provides a competitive benchmark for document classification [16, 17].

Grid search under 10-fold stratified cross-validation was conducted to determine the optimal setting for the *penalty parameter (C)*, *loss function (loss)*, and *tolerance of stopping (tol)* parameters. Since the classes are not distributed equally for the document classification, the *class weight* parameter was also optimised for the prediction task. Following the hyperparameter optimisation, the model was trained on 80% of the dataset and evaluated on the remaining 20% holdout set. To ensure robustness, we used 10-fold stratified cross-validation during training.

Table 4 provides a comprehensive evaluation of model performance, reporting *accuracy*, *weighted-average F1 score* (used as the primary metric), and *macro-average F1 score* (included to equally reflect performance across all classes).

Table 4: SVM results.

Model	**Dataset**	**Accuracy**	**F1 score**$_W$	**F1 score**$_M$
SVM	Train	0.974	0.974	0.969
	Test	0.978	0.978	0.969

5.4 Fine-Tuned Multilingual Language Models

We fine-tuned all seven language models, BERT, mBERT, DistilmBERT, XLM-R, Afro-XLM-R, SERENGETI, and XLM-V, using the same training configuration to ensure comparability across experiments. We used the HuggingFace Transformers [31] library and the Trainer API to manage training, evaluation, and checkpointing.

To ensure robust evaluation, we applied 5-fold stratified cross-validation on 80% of the dataset. This approach was chosen over 10-fold cross-validation due to the high computational demands of transformer-based models. Stratification was used to maintain proportional class representation across the folds, addressing the inherent class imbalance in the dataset.

All models were fine-tuned for a maximum of 10 epochs, using a batch size of 16, a learning rate of `2e-5`, and the `AdamW` optimiser with weight decay set to 0.01. We employed HuggingFace's `EarlyStoppingCallback` to prevent overfitting, monitoring the macro F1 score on the validation set.

Training was stopped early if no improvement in this metric was observed for three consecutive epochs *(patience=3)*. We enabled the `load_best_model_at_end` parameter to ensure that the best-performing model on the validation data was retained for final evaluation.

All input documents were tokenised using each model's respective tokeniser and truncated to a maximum sequence length of 512 tokens. Documents longer than this limit were truncated (not chunked), ensuring input consistency across models. While macro F1 score was used during training for early stopping, all final evaluation metrics, including accuracy, precision, recall, and F1 score, were calculated using `weighted averaging` to reflect the influence of class imbalance. All experiments were seeded with a fixed value `(seed=42)` to ensure reproducibility and deterministic behaviour.

5.5 Cross-Validation Results

The weighted average results, fine-tuning durations, are presented in Table 5. The highest scoring model is formatted in bold for convenience.

Table 5: Stratified 5-fold cross-validation results.

Model	**Accuracy**	**Precision**$_W$	**Recall**$_W$	**F1 score**$_W$	**Duration**
BERT	0.874 (0.008)	0.876 (0.007)	0.874 (0.008)	0.874 (0.008)	01:16:57
mBERT	0.946 (0.005)	0.947 (0.005)	0.946 (0.005)	0.946 (0.005)	01:28:09
DistilmBERT	0.937 (0.004)	0.938 (0.004)	0.937 (0.004)	0.937 (0.004)	01:07:19
XLM-R	0.942 (0.005)	0.943 (0.005)	0.942 (0.005)	0.942 (0.005)	02:15:27
Afro-XLM-R	0.952 (0.006)	0.953 (0.006)	0.952 (0.006)	0.952 (0.006)	02:03:27
SERENGETI	**0.968** (0.004)	**0.968** (0.004)	**0.968** (0.004)	**0.968** (0.004)	01:31:02
XLM-V	0.959 (0.008)	0.960 (0.008)	0.959 (0.008)	0.959 (0.008)	17:39:27

BERT achieves a weighted F1 score of 0.874. While this performance is respectable, especially considering that BERT was pre-trained exclusively on English, it underscores the limitations of applying monolingual models to multilingual or low-resource contexts such as Afrikaans.

Both mBERT and DistilmBERT demonstrate notable improvements over the BERT. mBERT, with a weighted F1 score of 0.946, benefits from its multilingual

pre-training, including exposure to Afrikaans and Dutch. DistilmBERT, achieving a weighted F1 score of 0.937, illustrates the effectiveness of model distillation, which retains essential knowledge while reducing model size and inference time.

Among the cross-lingual RoBERTa-based models, XLM-R and Afro-XLM-R show strong performance. XLM-R achieves a weighted F1 score of 0.942, while Afro-XLM-R slightly outperforms XLM-R with a weighted F1 score of 0.952. These results suggest that pre-training on African languages, including Afrikaans, contributes positively to classification performance in this task.

The SERENGETI model achieves the highest weighted F1 score among the fine-tuned transformer-based models, with a weighted F1 score of 0.968. Its broad coverage of African languages and inclusion of Afrikaans likely support its strong results. XLM-V also demonstrates robust performance, with a weighted F1 score of 0.959, indicating its competitiveness in this domain.

The close performance between SERENGETI, XLM-V, and Afro-XLM-R indicates that multilingual pre-training that includes Afrikaans, combined with larger vocabularies, can improve outcomes in Afrikaans classification tasks.

Finally, the relatively small standard deviations observed across all models (as indicated in Table 5) indicate consistent performance, reinforcing the robustness of the cross-validated training.

5.6 Classification Results

As explained in Sections 5.2 and 5.3, we report both weighted and macro F1 scores to reflect overall performance and class-wise balance, respectively. Bold indicates the highest scoring model. The results presented in Table 6 provide a comprehensive overview of the classification performance of various models on the test holdout set.

Table 6: Average 5-fold classification results.

Model	**Accuracy**	**Precision**$_W$	**Recall**$_W$	**F1 score**$_W$	**F1 score**$_M$
SVM (TF-IDF)	0.978 (0.000)	0.978 (0.000)	0.978 (0.000)	0.978 (0.000)	0.969 (0.000)
BERT	0.862 (0.007)	0.864 (0.006)	0.862 (0.007)	0.862 (0.007)	0.884 (0.006)
mBERT	0.933 (0.002)	0.935 (0.003)	0.933 (0.002)	0.933 (0.002)	0.942 (0.004)
DistilmBERT	0.932 (0.003)	0.933 (0.003)	0.932 (0.003)	0.932 (0.003)	0.936 (0.003)
XLM-R	0.941 (0.002)	0.943 (0.001)	0.941 (0.002)	0.941 (0.002)	0.950 (0.004)
Afro-XLM-R	0.947 (0.002)	0.949 (0.002)	0.947 (0.002)	0.948 (0.002)	0.958 (0.002)
SERENGETI	**0.964** (0.005)	**0.965** (0.005)	**0.964** (0.005)	**0.964** (0.005)	**0.965** (0.005)
XLM-V	0.954 (0.004)	0.955 (0.004)	0.954 (0.004)	0.954 (0.004)	0.961(0.003)

The baseline model, SVM (TF-IDF), achieved both an accuracy and a weighted F1 score of 0.978. This model demonstrated strong performance, particularly in the weighted metrics, underscoring its effectiveness in handling class imbalances. Notably, SVM outperformed all fine-tuned language models across all evaluation metrics in this study. This outcome highlights the continued relevance of tradi-

tional feature-based classifiers in text classification tasks, where TF-IDF features can remain highly competitive.

BERT achieved a weighted F1 score of 0.862, which underscores the limitations of using a monolingual model in a multilingual context. The multilingual variant, mBERT, and DistilmBERT achieved similar performance, with weighted F1 scores of 0.933 and 0.932, respectively. The close performance suggests that both models are effective for multilingual text classification tasks.

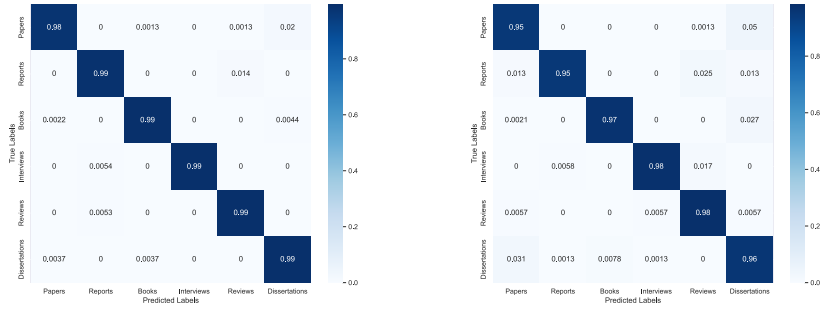
The performance of XLM-R and Afro-XLM-R surpassed that of mBERT and DistilmBERT. XLM-R achieved a weighted F1 score of 0.941, while Afro-XLM-R slightly outperformed XLM-R with a weighted F1 score of 0.948. SERENGETI emerged as the best-performing transformer-based model, achieving a weighted F1 score of 0.964. The second-best transformer-based model, XLM-V, achieved a weighted F1 score of 0.954, demonstrating strong performance. The evaluation results for each class are included and summarised in Table 7 for the best performing models: XLM-R, Afro-XLM-R, SERENGETI and XLM-V.

Table 7: Average 5-fold evaluation scores (precision, recall, and F1) for each class.

	XLM-R			**Afro-XLMR**			**SERENGETI**			**XLM-V**		
	P	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
Papers	0.96	0.92	0.94	0.96	0.92	0.94	0.98	0.95	0.96	0.97	0.93	0.95
Reports	0.98	0.89	0.93	0.98	0.94	0.96	0.96	0.93	0.95	0.97	0.93	0.95
Books	0.97	0.96	0.96	0.99	0.96	0.97	0.99	0.98	0.98	0.98	0.97	0.98
Interviews	0.98	0.98	0.98	0.97	0.98	0.97	0.99	0.98	0.98	0.99	0.98	0.98
Reviews	0.96	0.98	0.97	0.97	0.97	0.97	0.96	0.97	0.96	0.97	0.98	0.97
Dissertations	0.90	0.94	0.92	0.90	0.96	0.93	0.94	0.97	0.95	0.92	0.96	0.94
Micro	0.94	0.94	0.94	0.95	0.95	0.95	0.96	0.96	0.96	0.95	0.95	0.95
Macro	0.96	0.94	0.95	0.96	0.95	0.96	0.97	0.96	0.97	0.96	0.96	0.96
Weighted	0.94	0.94	0.94	0.95	0.95	0.95	0.96	0.96	0.96	0.95	0.95	0.95

The detailed confusion matrices for the fine-tuned SERENGETI and XLM-V models are presented in Figures 1a and 1b, respectively. This offers a comparative view of each model's classification behaviour across the six document types, highlighting their strengths and common misclassification patterns.

To ensure robust and generalisable evaluation, predictions on the held-out test set were generated using a *hard voting ensemble* across all five cross-validated model folds for each architecture. Specifically, for each test instance, the predicted class label was determined by majority vote from the five independently fine-tuned model folds. This ensemble approach mitigates the risk of overfitting or selection bias that can arise from relying solely on the best-performing single fold, which may yield over-optimistic or unstable estimates of model performance due to variability in data splits. By aggregating predictions across folds, hard voting yields a more stable and reliable estimate of overall model effectiveness.



(a) Fine-tuned SERENGETI model. (b) Fine-tuned XLM-V model.

5.7 Hybrid Models

For this study, we also explored a hybrid approach that combines fine-tuned transformer-based language models with traditional machine learning, specifically Logistic Regression (LR) and Support Vector Machines (SVM). Our objective was to evaluate the effectiveness of these classifiers in conjunction with embeddings derived from language models. After considering several pooling methods, including *max pooling* and *[CLS] token pooling*, we selected *mean pooling* as our aggregation strategy, applying it to the last hidden layer embeddings to obtain fixed-size representations for each document.

Using the fine-tuned transformer-based models described in Section 5.4, we extracted contextual embeddings from the *last hidden layer* for each input sentence. These embeddings are high-dimensional vector representations that encode semantic information about the input text. We then applied the *mean pooling* strategy to convert these embeddings into a form suitable for traditional classifiers. This strategy involves averaging the embeddings of all tokens in a sentence to produce a single fixed-size vector that encapsulates the overall semantic content of the sentence as represented by the model's final layer.

The resulting pooled embeddings served as input features for our baseline machine learning classifiers, LR and SVM. By integrating the language model embeddings with traditional classifiers, our hybrid approach aimed to capitalise on the strengths of both methods. This integration allowed us to comprehensibly evaluate the combined effectiveness of these approaches in sentence-level classification tasks. The results of our experiments, which highlight the performance of this hybrid methodology, are summarised in Table 8.

Results for the Hybrid Models: The SVM classifier using SERENGETI embeddings demonstrated the best performance among the hybrid models, achieving a weighted F1 score of 0.967. This indicates that combining SVM with SERENGETI contextual embeddings can lead to high performance in text clas-

Table 8: Last hidden layer classification results.

Model	**Accuracy**	**Precision**$_W$	**Recall**$_W$	**F1 score**$_W$	**F1 score**$_M$
SVM (TF-IDF)	0.978	0.978	0.978	0.978	0.969
Logistic Regression					
-BERT	0.868	0.870	0.868	0.868	0.890
-mBERT	0.937	0.937	0.937	0.937	0.944
-DistilmBERT	0.928	0.929	0.928	0.928	0.933
-XLM-R	0.944	0.945	0.944	0.944	0.951
-Afro-XLM-R	0.947	0.948	0.947	0.947	0.954
-SERENGETI	0.966	**0.967**	0.966	0.966	0.967
-XLM-V	0.957	0.957	0.957	0.957	0.962
SVM					
-BERT	0.869	0.870	0.869	0.869	0.891
-mBERT	0.935	0.936	0.935	0.935	0.942
-DistilmBERT	0.930	0.931	0.930	0.930	0.935
-XLM-R	0.944	0.944	0.944	0.944	0.951
-Afro-XLM-R	0.948	0.949	0.948	0.949	0.954
-SERENGETI	**0.967**	**0.967**	**0.967**	**0.967**	**0.969**
-XLM-V	0.957	0.958	0.957	0.957	0.962

sification tasks. In the context of this study, it highlights the effectiveness of hybrid approaches for improving document classification in Afrikaans.

Similarly, the LR classifier with SERENGETI embeddings also achieved strong results, with a weighted F1 score of 0.966. This demonstrates that Logistic Regression is also capable of effectively leveraging SERENGETI contextual embeddings, offering a viable alternative to SVM for this classification task.

The XLM-V embeddings also demonstrated competitive results. Both SVM and LR classifiers using XLM-V embeddings achieved a weighted F1 score of 0.957, with good precision and recall, indicating their effectiveness in capturing relevant features for the task.

Embeddings derived from BERT, mBERT, DistilmBERT, XLM-R, and Afro-XLM-R, also yielded respectable results, although they did not match the performance of the SERENGETI or XLM-V based models. Among these, the highest score was achieved by the SVM classifier with Afro-XLM-R embeddings (weighted F1 score of 0.949), followed closely by the LR classifier (weighted F1 score of 0.947).

While these hybrid models demonstrate strong performance, it is essential to note that none of the models outperformed the baseline SVM (TF-IDF) model reported in Section 5.3, which remained the top-performing model overall with a weighted F1 score of 0.978. Nonetheless, these findings underscore the potential of combining contextual embeddings from models like SERENGETI and XLM-V with traditional classifiers for Afrikaans text classification tasks. The consistently strong performance of SVM and LR across multiple embedding sources suggests their continued relevance for low-resource language applications.

6 Discussion

Our findings suggest that multilingual transformer models that include Afrikaans in their pre-training data can, in some cases, transfer this knowledge to improve performance on low-resource language tasks such as Afrikaans document classification. This is demonstrated by the strong performance of models like SERENGETI and XLM-V. However, this effect is not uniform across all models, as illustrated by the comparatively lower performance of the monolingual BERT model.

6.1 Model Adaptation and Fine-Tuning

The study highlights the importance of fine-tuning pre-trained language models on domain-specific labelled datasets. Even with limited annotated data for Afrikaans, consistent fine-tuning, using stratified cross-validation and uniform training conditions, produced competitive results. This challenges the assumption that large volumes of language-specific data are always necessary, and underscores the adaptability of transformer architectures in low-resource settings.

However, model performance in this study was not determined by fine-tuning alone. Several upstream factors appear to influence outcomes, including the inclusion of Afrikaans and Dutch in the pre-training corpora, the underlying model architecture (e.g., BERT vs. RoBERTa), vocabulary size, and the number of model parameters.

For example, BERT, which was trained exclusively on English, achieved the lowest weighted F1 score (0.862). In contrast, mBERT and DistilmBERT, which included Afrikaans and Dutch, performed significantly better, with weighted F1 scores of 0.933 and 0.932, respectively. However, even stronger performance was observed from RoBERTa-based models such as XLM-R (0.941), Afro-XLM-R (0.948), SERENGETI (0.964), and XLM-V (0.954). This pattern suggests that the architectural improvements of RoBERTa over BERT may contribute meaningfully to the downstream classification task in this study.

In addition, larger models with broader vocabularies, such as XLM-V (778M parameters, 900K vocab size), also tended to perform better. SERENGETI and Afro-XLM-R achieved high F1 scores despite lacking Dutch in their training data, indicating that coverage of a wide range of African languages and possibly higher representation of Afrikaans itself, may contribute more directly to improved classification performance than relying solely on typological similarity. While Dutch and Afrikaans share a close linguistic relationship, our results suggest that this typological similarity alone does not necessarily guarantee better model performance.

It is important to note that the precise proportion of Afrikaans present in the pre-training corpora of these models is unknown. As such, any interpretation of performance differences must be approached with caution. Language coverage, architecture, and scale, each appear to play important roles in downstream classification performance, but further empirical work is required to isolate the

contribution of these individual factors, particularly in the context of Afrikaans and other low-resource languages.

6.2 Embedding-Based Classification

The hybrid models evaluated in this study, combining contextual embeddings from transformer-based models with traditional classifiers such as SVM and Logistic Regression, demonstrated consistently strong performance. Notably, SVM with SERENGETI embeddings achieved a weighted F1 score of 0.967, closely matching the fine-tuned SERENGETI model (0.964). Similarly, SVM with XLM-V and Afro-XLM-R embeddings also performed competitively: SVM+XLM-V achieved a weighted F1 score of 0.957 (vs. 0.954 for the fine-tuned version), while SVM+Afro-XLM-R achieved a weighted F1 score of 0.949 (vs. 0.948 for the fine-tuned model). While none of the hybrid approaches surpassed the SVM (TF-IDF) baseline (weighted F1 score of 0.978), their performance highlights the practical viability of using language model embeddings as feature extractors, particularly in low-resource settings where full fine-tuning may be computationally expensive or difficult to maintain.

6.3 Impact on Language Preservation and Archiving

The application of multilingual language models for document classification in Afrikaans has implications for language preservation and archival efforts. This study shows how NLP can aid in preserving cultural heritage by automating the classification of large volumes of documents in a low-resource language.

Our results show that transformer-based and hybrid models, especially SVM and SERENGETI contextual embeddings, perform well in this context. However, for such models to be practically useful, they must be integrated into existing archival workflows. While prior work demonstrated cost-effective metadata extraction using cloud infrastructure [29], full integration of classification models would require a redesign of current systems, a task beyond this paper's scope.

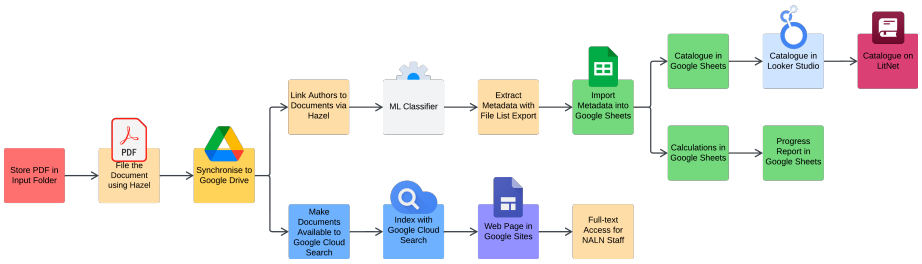


Fig. 2: Illustration of a potential NALN document classification pipeline, adapted from [29].

To illustrate this potential, we extend the NALN archival pipeline from [29] by inserting a **machine-learning (ML) classifier module** into the metadata

generation stage (labelled ML classifier in Fig. 2). Figure 2 presents an updated workflow where the classifier operates as an automated module in the metadata generation process. This integration could streamline indexing and improve the discoverability of new archival materials, though further implementation and evaluation are left to future research.

7 Limitations

While this study offers valuable insights into the classification of Afrikaans documents using multilingual language models, it is not without limitations. One potential constraint is the challenge of capturing the linguistic diversity and complexity of the Afrikaans language, which may affect the generalisability of the model's performance across various contexts and document types. This can be mitigated by expanding the dataset to include a broader range of document types, thereby enhancing the model's ability to generalise across different linguistic nuances. Additionally, the focus on document classification may overlook other essential aspects of archival processing, such as metadata extraction and document summarisation, which could further enhance information retrieval in Afrikaans archival systems.

Moreover, the reliance on pre-trained multilingual models introduces another layer of complexity. While these models have shown promising results, their performance can be influenced by the quality, coverage, and proportion of Afrikaans data in their pre-training corpora, details which are not always publicly disclosed. As discussed in Sections 6.1 and 6.2, even models that include typologically related languages like Dutch do not necessarily perform better, suggesting that model architecture, scale, and language-specific representation may have a greater impact than linguistic similarity alone. Thus, there remains a risk that such models may not fully capture the unique characteristics of Afrikaans, particularly if the pre-training data lacks sufficient depth or diversity in that language.

8 Conclusions and future research

This paper contributes to the advancement of document type classification in Afrikaans, a language that lacks sufficient resources for training and fine-tuning language models. Our investigation explored various multilingual language models and techniques tailored to the specificity of Afrikaans texts, culminating in the identification of fine-tuning SERENGETI and XLM-V as the most effective transformer-based classifiers.

A notable contribution of this study is the creation of an Afrikaans dataset, which serves as a valuable resource for training and fine-tuning neural language models. This dataset facilitates further research using Afrikaans in NLP tasks and also sets a precedent for similar initiatives in other low-resource languages. Future research should focus on expanding the dataset, incorporating additional document types and genres, to further enrich the training data and improve

model performance. In addition, greater transparency in the composition of pretraining corpora would support a deeper understanding of model behaviour in low-resource languages like Afrikaans.

Moreover, the hybrid models showed promising results, suggesting that combining traditional classifiers with embeddings from multilingual models can be effective for Afrikaans text classification. However, it is noteworthy that in this study the traditional SVM (TF-IDF) baseline outperformed all fine-tuned and hybrid models. This finding reinforces the continued relevance of classical machine learning approaches for well-structured tasks in low-resource settings. Future investigations could explore additional hybrid approaches, including ensemble methods, to further optimise performance.

In conclusion, our findings provide practical insights for information retrieval and archival systems, and pave the way for future studies to optimise multilingual models and hybrid strategies for low-resource languages.

Acknowledgments. We thank the three reviewers for their constructive feedback that improved the quality of the paper.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Adebara, I., Elmadany, A., Abdul-Mageed, M., Alcoba Inciarte, A.: SERENGETI: Massively Multilingual Language Models for Africa. In: Findings of the Association for Computational Linguistics: ACL 2023. pp. 1498–1537. Association for Computational Linguistics, Toronto (2023). https://doi.org/10.18653/v1/2023.findings-acl.97
2. Alabi, J.O., Adelani, D.I., Mosbach, M., Klakow, D.: Adapting Pre-trained Language Models to African Languages via Multilingual Adaptive Fine-Tuning. In: Proceedings of the 29th International Conference on Computational Linguistics. pp. 4336–4349. International Committee on Computational Linguistics, Gyeongju, Republic of Korea (2022), https://aclanthology.org/2022.coling-1.382
3. Bird, S., Klein, E., Loper, E.: Natural Language Processing with Python, vol. 43. O'Reilly, Sebastopol, CA (2009), https://www.nltk.org/book/
4. Brokensha, S., Kotzé, E., Senekal, B.: Machine learning for document classification in an archive of the National Afrikaans Literary Museum and Research Centre. Journal of the South African Society of Archivists **56**, 1–14 (2023), https://www.ajol.info/index.php/jsasa/article/view/260311
5. Carducci, G., Leontino, M., Radicioni, D.P., Bonino, G., Pasini, E., Tripodi, P.: Semantically Aware Text Categorisation for Metadata Annotation. In: Digital Libraries: Supporting Open Science, pp. 315–330. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-11226-4_25
6. Colavizza, G., Blanke, T., Jeurgens, C., Noordegraaf, J.: Archives and AI: An Overview of Current Debates and Future Perspectives. Journal on Computing and Cultural Heritage **15**(1), 1–15 (2022). https://doi.org/10.1145/3479010

7. Colla, D., Goy, A., Leontino, M., Magro, D., Picardi, C.: Bringing Semantics into Historical Archives with Computer-aided Rich Metadata Generation. Journal on Computing and Cultural Heritage **15**(3), 1–24 (2022). https://doi.org/10.1145/3484398
8. Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., Stoyanov, V.: Unsupervised Cross-lingual Representation Learning at Scale. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 8440–8451. Association for Computational Linguistics, Stroudsburg, PA, USA (2020). https://doi.org/10.18653/v1/2020.acl-main.747
9. Connelly, M.J., Hicks, R., Jervis, R., Spirling, A., Suong, C.H.: Diplomatic documents data for international relations: the Freedom of Information Archive Database. Conflict Management and Peace Science **38**(6), 762–781 (2021). https://doi.org/10.1177/0738894220930326
10. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACLHLT 2019. pp. 4171–4186. Association for Computational Linguistics, Minneapolis, MN, USA (2019), https://aclanthology.org/N19-1423.pdf
11. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: LIBLINEAR: A Library for Large Linear Classification. Journal of Machine Learning Research **9**, 1871–1874 (2008), https://www.csie.ntu.edu.tw/ cjlin/liblinear/
12. Galloway, F., Roux, A.: Breyten Breytenbach: Woordenar woordnar. Protea Boekhuis, Pretoria (2019)
13. Giannini, T., Bowen, J.P.: Computational Culture: Transforming Archives Practice and Education for a Post-Covid World. Journal on Computing and Cultural Heritage **15**(3), 1–18 (2022). https://doi.org/10.1145/3493342
14. Golub, K., Hagelbäck, J., Ardö, A.: Automatic Classification of Swedish Metadata Using Dewey Decimal Classification: A Comparison of Approaches. Journal of Data and Information Science **5**(1), 18–38 (2020). https://doi.org/10.2478/jdis-2020-0003
15. Hutchinson, T.: Natural language processing and machine learning as practical toolsets for archival processing. Records Management Journal **30**(2), 155–174 (2020). https://doi.org/10.1108/RMJ-09-2019-0055
16. Kotzé, E., Senekal, B., Daelemans, W.: Exploring the Classification of Security Events using Sparse and Dense Representation of Text. In: Proceedings of the 2020 International SAUPEC/RobMech/PRASA Conference. pp. 1–6. IEEE (2020). https://doi.org/10.1109/SAUPEC/RobMech/PRASA48453.2020.9041092, https://ieeexplore.ieee.org/document/9041092/
17. Kotzé, E., Senekal, B.A.: Afrikaans Literary Genre Recognition using Embeddings and Pre-Trained Multilingual Language Models. In: 2024 International Conference on Artificial Intelligence, Computer, Data Sciences and Applications (ACDSA). pp. 1–6. IEEE, Victoria, Seychelles (feb 2024). https://doi.org/10.1109/ACDSA59508.2024.10467838, https://ieeexplore.ieee.org/document/10467838/
18. Kotzé, E., Senekal, B.A., Daelemans, W.: Automatic classification of social media reports on violent incidents in South Africa using machine learning. South African Journal of Science **116**(3/4), 1–8 (2020). https://doi.org/10.17159/sajs.2020/6557

19. Lai, S., Xu, L., Liu, K., Zhao, J.: Neural Networks for Text Classification. In: Proceedings of the National Conference on Artificial Intelligence. pp. 2267–2273. Austin, Texas (2015), https://dl.acm.org/doi/10.5555/2886521.2886636
20. Liang, D., Gonen, H., Mao, Y., Hou, R., Goyal, N., Ghazvininejad, M., Zettlemoyer, L., Khabsa, M.: XLM-V: Overcoming the Vocabulary Bottleneck in Multilingual Masked Language Models. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. pp. 13142–13152. Association for Computational Linguistics, Stroudsburg, PA, USA (2023). https://doi.org/10.18653/v1/2023.emnlp-main.813
21. Nassar, M., Rogers, A.B., Talo', F., Sanchez, S., Shafique, Z., Finn, R.D., McEntyre, J.: A machine learning framework for discovery and enrichment of metagenomics metadata from open access publications. GigaScience **11** (2022). https://doi.org/10.1093/gigascience/giac077
22. Park, J.r., Brenza, A.: Evaluation of Semi-Automatic Metadata Generation Tools: A Survey of the Current State of the Art. Information Technology and Libraries **34**(3), 22–42 (2015). https://doi.org/10.6017/ital.v34i3.5889
23. Paszke, A., Chanan, G., Lin, Z., Gross, S., Yang, E., Antiga, L., Devito, Z.: Automatic differentiation in PyTorch. In: 31st Conference on Neural Information Processing Systems (2017), https://openreview.net/forum?id=BJJsrmfCZ
24. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, É.: Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research **12**, 2825–2830 (2011). https://doi.org/10.1007/s13398-014-0173-7.2
25. Řehůřek, R., Sojka, P.: Software framework for topic modelling with large corpora. In: Proceedings of the LREC Workshop on New Challenges for NLP Frameworks. pp. 45–50. Valletta (2010), https://github.com/piskvorky/gensim
26. Rolan, G., Humphries, G., Jeffrey, L., Samaras, E., Antsoupova, T., Stuart, K.: More human than human? Artificial intelligence in the archive. Archives and Manuscripts **47**(2), 179–203 (2018). https://doi.org/10.1080/01576895.2018.1502088
27. Sanh, V., Debut, L., Chaumond, J., Wolf, T.: DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv (2019), http://arxiv.org/abs/1910.01108
28. Sebastiani, F.: Machine learning in automated text categorization. ACM Computing Surveys **34**(1), 1–47 (2002)
29. Senekal, B., Kotzé, E.: 'n Pyplyn vir die versameling, ontsluiting en beskikbaarstelling van materiaal vir die Nasionale Afrikaanse Letterkundige Museum en -Navorsingsentrum (NALN) se knipselversameling. Stilet : Tydskrif van die Afrikaanse Letterkundevereniging **35**(2), 38–62 (2023). https://doi.org/10.59507/stilet.2023.35.2.3
30. Van Rijsbergen, C.: Foundation of evaluation. Journal of Documentation **30**(4), 365–373 (1974). https://doi.org/10.1108/eb026584
31. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., Drame, M., Lhoest, Q., Rush, A.: Transformers: State-of-the-Art Natural Language Processing. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. pp. 38–45. Association for Computational Linguistics, Stroudsburg, PA, USA (2020). https://doi.org/10.18653/v1/2020.emnlp-demos.6

32. Yasser, A.M., Clawson, K., Bowerman, C.: Saving Cultural Heritage with Digital Make-Believe: Machine Learning and Digital Techniques to the Rescue. In: Electronic Visualisation and the Arts (EVA 2017), BCS Learning & Development. pp. 1–5 (2017). https://doi.org/10.14236/ewic/HCI2017.97