Multilabel Classification of Bilingual Patents Using OneVsRestClassifier: A Semiautomated Approach

Slamet Widodo[1], Ermatita[2]*, Deris Stiawan[3]
Doctoral Program in Engineering Science, University Sriwijaya Palembang, Indonesia[1, 2, 3]
Department of Computer Engineering, Politeknik Negeri Sriwijaya Palembang, Indonesia[1]
Faculty of Computer Science, University Sriwijaya Palembang, Indonesia[2, 3]

***Abstract*—In response to the increasing complexity and volume of patent applications, this research introduces a semiautomated system to streamline the literature review process for Indonesian patent data. The proposed system employs a synthesis of multilabel classification techniques based on natural language processing (NLP) algorithms. This methodology focuses on developing an iterative and modular system, with each step visualised in detailed flowcharts. The system design incorporates data collection and preprocessing, multilabel classification model development, model optimisation, query and prediction, and results presentation modules. Experimental results demonstrate the promising potential of the multilabel classification model, achieving a micro F1 score of 0.6723 and a macro F1 score of 0.6009. The OneVsRestClassifier model with LinearSVC as the base classifier shows reasonably good performance in handling a bilingual dataset comprising 15,097 patent documents. The optimal model configuration uses TfidfVectorizer with 20,000 features, including bigrams, and an optimal C parameter of 0.1 for LinearSVC. Performance analysis reveals variations across IPC classes, indicating areas for further improvement. The discussion highlights the implications of the proposed system for researchers, patent examiners and industry professionals by facilitating efficient searches within patent databases. This study acknowledges the potential of semiautomated systems to enhance the efficiency of patent analysis while emphasising the need for further research to address identified challenges, such as class imbalance and performance variations across patent categories. This research paves the way for further developments in the field of automated patent classification, aiming to improve efficiency and accuracy in international patent systems while recognising the crucial role of human experts in the patent classification process.**

Keywords*—*Multilabel patent classification; Natural Language Processing (NLP); OneVsRestClassifier; TF–IDF vectorisation; bilingual patent analysis

I. Introduction

At present, conducting manual patent literature reviews involves a relatively challenging level of difficulty. The continuous influx of submissions adds complexity, which demands efficient analysis for intellectual property management and strategic innovation tracking [1]. The intricate technical and legal language in these documents also contributes to the complexity of manual processing [2]. Traditional methods, although widely used, are time consuming, resource intensive and prone to human error and bias, which can lead to inconsistent and unreliable results [1], [2].

While recent advances in natural language processing (NLP) have automated aspects of patent analysis [3], [4], critical gaps remain. First, most systems have focused on monolingual datasets (e.g. English only [5] or Indonesian only [6]), neglecting the bilingual nature of patents in countries such as Indonesia, where filings combine local and international languages. Second, existing methods have often failed to address class imbalance in the International Patent Classification (IPC) system, leading to poor performance in underrepresented technology categories (e.g. Y02A) [7]. Third, few studies have integrated local patent databases (e.g. Indonesian Patent Database) with global repositories (e.g. Google Patents), limiting their applicability to multinational innovation ecosystems. Our work bridges these gaps by proposing a bilingual framework that combines Indonesian and English patents, addresses class imbalance through weighted learning and validates utility across diverse IPC categories.

This study addresses the following research questions:

RQ1: How effective is the OneVsRestClassifier with LinearSVC for bilingual (Indonesian–English) patent classification compared to monolingual approaches?

RQ2: What feature engineering strategies (e.g. TF–IDF with bigrams and class weighting) optimise multilabel IPC classification performance in imbalanced datasets?

RQ3: How does class imbalance affect model performance across different IPC categories, and what mitigation strategies are most effective?

Recognising these limitations, this research seeks to refine the semiautomated process for reviewing Indonesian patent literature by using data from local and international repositories. Our approach uses web-scraping techniques to obtain datasets, followed by preprocessing to clean and structure the data for processing using machine learning algorithms. We use the IPC to train multilabel classification models, which allow for categorisation that represents the diverse nature of patent data [3], [4].

The proposed solution links multilabel classification algorithms to increase efficiency and reduce the resources required for a comprehensive review [3], [4]. This process aims to optimise patent analysis by leveraging computational power. The application of these techniques is intended to address the vast amount of data and complex patent language. By using an approach based on machine learning, the proposed system seeks to simplify this complex task and make it more manageable [5].

The significance of this research is its significant potential to develop and advance patent classification techniques by substantially improving the accuracy and precision of analysis, as well as accelerating systematic, structured and data-driven decision-making processes [1], [4]. This research has high value and strong relevance to patent examiners, research and development institutions and companies that rely heavily on accurate and efficient patent analysis, with much broader implications for innovation tracking, in-depth competitive analysis and future technology forecasting [6], [7]. Ultimately, this study aims to lay a strong foundation and solid groundwork for visionary strategic planning and well-informed policymaking in the dynamic field of intellectual property [8].

Paper Overview. The remainder of this paper is organised as follows. Section II reviews key studies on multilabel patent classification, emphasising the bilingual and imbalanced data contexts. Section III outlines the proposed methodology, detailing the dataset collection, feature engineering and model training processes. Section IV presents the experimental setup, along with the results and discussion of the findings. Finally, Section V concludes the paper, summarising the main contributions, acknowledging current limitations and suggesting avenues for future research.

II. Related Work

The increasing volume of patent applications worldwide has triggered a critical need for advancements in patent analysis methodologies. Traditional manual reviews, characterised by their meticulous yet cumbersome nature, have become unsustainable in the face of rapid technological innovation; the corresponding increase in intellectual property documentation [1] highlights the intrinsic limitations of manual reviews, particularly their vulnerability to human error and the inherent subjectivity in interpreting complex legal and technical terms.

Patent classification using the k-nearest neighbours (KNN) and fastText classifier algorithms individually performs worse than when they are combined by a meta-classifier. The former approach is based on a linguistically supported KNN algorithm using a method of searching for topically similar documents based on comparisons of lexical descriptor vectors. The latter approach employs fastText based on word embeddings, in which sentence (or document) vectors are obtained by averaging n-gram embeddings, and then vectors are used as features in multinomial logistic regression [9].

To address challenges, the field of NLP has emerged as a beacon of innovation. The ability of NLP to parse and interpret complex language structures makes it a powerful tool for the semiautomated analysis of patent documents. The study in [2] underlined the transformative impact of NLP in the domain of summarisation, simplification and generation of patent texts, indicating an urgent need for research specifically tailored to the nuanced demands of patent documentation.

At the forefront of this domain, multilabel classification has been identified as a crucial component for effective patent categorisation, often encapsulating the convergence of various technological domains. The complexity involved in accurately classifying multifaceted documents is further exacerbated in fields such as artificial intelligence, in which the intersection of technology and legal language demands sophisticated computational techniques for precise analysis [3] [4].

The integration of NLP techniques into semiautomated systems for patent analysis signifies a substantial leap from manual review processes, promising enhanced accuracy and efficiency in patent analysis. However, this integration is not without challenges. The need for comprehensive and well-annotated datasets for training and testing NLP models remains an ongoing hurdle, alongside the development of models that can adeptly navigate the intricacies of patent language and accurately reflect the evolving landscape of technological innovation [6], [10].

Three critical gaps persist in the literature, which are as follows:

1) Monolingual bias: Most studies [9], [10] have focused on monolingual patent datasets, overlooking the bilingual complexity inherent in countries such as Indonesia. For instance, [9] combined KNN and fastText but only tested on English patents, neglecting cross-lingual term alignment.

2) Class imbalance: Prior works [3], [11] have often assumed balanced IPC label distributions, leading to poor performance in rare categories (e.g. Y02A). For example, [11] reported high accuracy overall but did not address label skew.

3) Local–global integration: Existing frameworks [12], [13] have rarely combined local patent databases (e.g. Indonesian Patent Database) with international repositories (e.g. Google Patents), limiting their ability to capture region-specific innovations.

Our work directly addresses these gaps by (1) designing a bilingual (Indonesian–English) classification pipeline, (2) optimising for class imbalance via class_weight=‘balanced’ in LinearSVC and (3) integrating local and global patent data to enhance coverage and relevance.

As the discipline evolves, ethical considerations and data sharing become increasingly important. Unbiased data representation in training sets is crucial to mitigating biases that might be perpetuated in patent analysis. Additionally, sharing open-source tools and datasets to catalyse innovation through collaborative efforts underscores the importance of interdisciplinary cooperation in advancing the capabilities of NLP systems in patent informatics. [11], [8] emphasised the importance of collaboration in data sharing and of ethical implications in developing NLP tools for scientific research.

Natural language processing technology has made significant strides in transforming patent informatics, and the field is ripe for further exploration and development. The research in [12], [13] provided evidence of the effectiveness of semiautomated approaches in machine learning-based literature reviews, which can be applied in patent data analysis. Further research is needed to refine NLP models, enhance the understanding and processing of patent data and drive systematic and data-driven approaches to intellectual property management [10], [14]. One approach to handling NLP is the classification chain (CC), which links these binary classifiers in a certain sequential order so that each classifier includes labels predicted by the previous classifier as additional features. Despite the simplicity of this approach, recent comprehensive empirical studies have shown that CC is among the best-performing algorithms [15].

III. Methodology

A. Dataset

This study combines 7,298 patents from the Indonesian Patent Database and 7,801 patents from Google Patents, forming a bilingual corpus of 15,097 documents. This hybrid dataset was strategically selected to address the following three critical requirements for robust multilabel patent classification:

1) Bilingual representation: The Indonesian Patent Database provides local language coverage (Indonesian), while Google Patents ensures international relevance (English). This combination reflects real-world patent ecosystems in multilingual jurisdictions, such as Indonesia.

2) Class diversity: Google Patents broadens the scope of IPC codes beyond region-specific innovations, ensuring the coverage of emerging global technologies (e.g. Y02A for climate adaptation).

3) Imbalanced IPC mitigation: Merging datasets diversifies label distributions, reducing bias towards dominant classes (e.g. A61K) while retaining rare categories for comprehensive analysis.

The dataset includes four key features: patent_id, patent_title, patent_abstract and ipc_code. Table I summarises the dataset composition.

TABLE I. DATASET OF INDONESIAN PATENTS AND GOOGLE PATENTS

No	Dataset	Jumlah Record
1	Indonesia_Patents	7298
2	Google_Patents	7801
	Total	15099

B. Proposed Framework

The framework depicted in Fig. 1 is the basis of this research. Data were taken from Google Patents and the Indonesian Patent Database [10]. We begin by collecting patent data from these two sources, ensuring a comprehensive dataset that covers various innovations. Once collected, these data are then preprocessed, which includes text cleaning, stopword removal and preparation for efficient machine learning classification [1], [15]–[17].

The research methodology is iterative and modular [18], focused on developing a semiautomated system for reviewing Indonesian patent data literature [19]. Each step is visualised in a detailed flowchart, which serves as a guide through various stages of data collection, processing and analysis. Fig. 2 explains the flowchart of this patent classification system research, which uses machine learning techniques to classify patent documents into IPC codes [9]. This system is designed to process and analyse patent data from Google Patents and the Indonesian Patent Database. The feature structure consists of the ID as a unique patent identification, the patent title, the patent abstract or summary and related IPC codes. Next, data loading and cleaning are performed. The clean_text() function performs text cleaning by removing HTML tags and non-alphanumeric characters and digits and converting text to lowercase [20], [21]. Text processing involves tokenisation and stopword removal using a combination of English and Indonesian stopwords [1], [22], [23]. Feature engineering and data splitting combine datasets, convert IPC codes into multilabel formats and split data into training and testing sets. Model training and evaluation conduct the experiments with various parameter configurations, train the OneVsRestClassifier model with LinearSVC as the base classifier and calculates the evaluation metrics for each configuration. The OneVsRest (OVR) model can provide informative hidden representations for unknown examples, and in open-set classification scenarios, the proposed probability model is better than modern approaches [15], [24].

This research begins by collecting patent data, followed by preprocessing procedures to prepare the data for classification. The processed data are then used to train and evaluate multilabel classification models, specifically the OneVsRestClassifier algorithm, to assign multiple IPC labels to each patent document [19]. This research also performs experiments to optimise the model by varying parameter values, such as n-gram range and maximum features for TF–IDF, as well as the C parameter for LinearSVC. The performance of each configuration is assessed using evaluation metrics, such as the F1 score (micro and macro), as well as cross-validation to determine the optimal model configuration [25], [26].

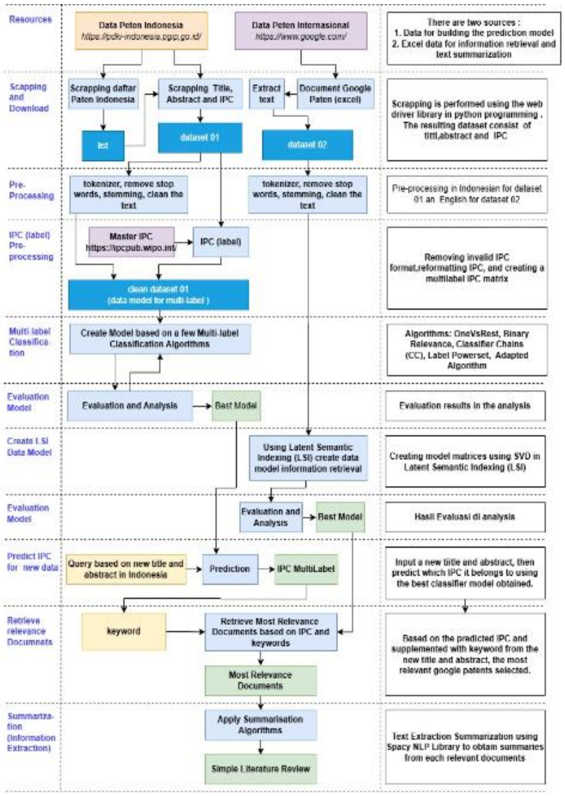
Resources
Data Peten Indonesia
https://pdki-indonesia.dgip.go.id/
Data Peten Internasional
https://www.google.com/
There are two sources :
1. Data for building the prediction model
2. Excel data for information retrieval and text summarization
Scapping and Download
Scrapping daftar Paten Indonesia
Scrapping Title, Abstract and IPC
Extract text
Document Google Paten (excel)
list
dataset 01
dataset 02
Scrapping is performed using the web driver library in python programming . The resulting dataset consist of titl,abstract and IPC
Pre-Processing
tokenizer, remove stop words, stemming, clean the text
tokenizer, remove stop words, stemming, clean the text
Pre-processing in Indonesian for dataset 01 an English for dataset 02
IPC (label) Pre-processing
Master IPC
https://ipcpub.wipo.int/
IPC (label)
clean dataset 01
(data model for multi-label)
Removing invalid IPC format,reformatting IPC, and creating a multilabel IPC matrix
Multi label Classifica-tion
Create Model based on a few Multi-label Classification Algorithms
Algorithms: OneVsRest, Binary Relevance, Classifier Chains (CC), Label Powerset, Adapted Algorithm
Evaluation Model
Evaluation and Analysis
Best Model
Evaluation results in the analysis
Create LSI Data Model
Using Latent Semantic Indexing (LSI) create data model information retrieval
Creating model matrices using SVD in Latent Semantic Indexing (LSI)
Evaluation Model
Evaluation and Analysis
Best Model
Hasil Evaluasi di analysis
Predict IPC for new data
Query based on new title and abstract in Indonesia
Prediction
IPC MultiLabel
Input a new title and abstract, then predict which IPC it belongs to using the best classifier model obtained.
Retrieve relevance Documnets
keyword
Retrieve Most Relevance Documents based on IPC and keywords
Most Relevance Documents
Based on the predicted IPC and supplemented with keyword from the new title and abstract, the most relevant google patents selected.
Summariza-tion (Information Extraction)
Apply Summarisation Algorithms
Simple Literature Review
Text Extraction Summarization using Spacy NLP Library to obtain summaries from each relevant documents

Fig. 1. Proposed framework architecture.

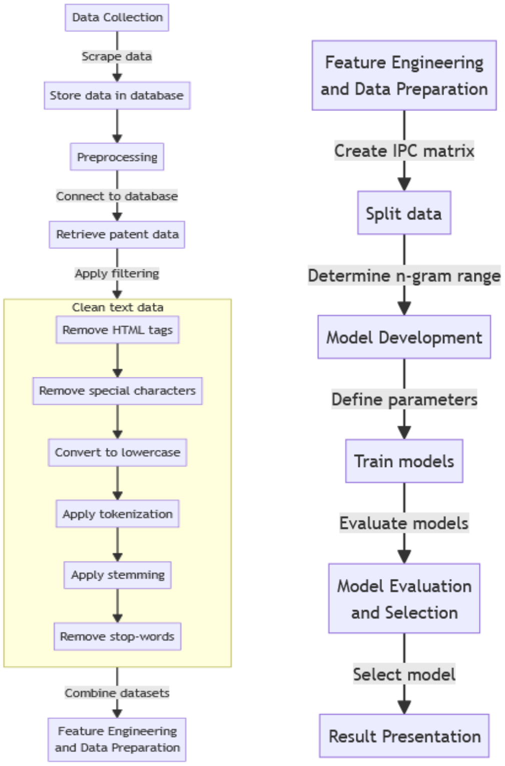

Fig. 2. Research flowchart.

C. OneVsRestClassifier

The OVR method is a strategy used in multiclass classification, in which separate binary classifiers are trained for each class to distinguish a particular class from all other classes [24], [27]. In this approach, for a particular class, samples belonging to it are treated as a positive class, and all other samples are treated as a negative class. This results in the need for only K binary classifiers for K classes, which is a smaller number than that in the one-versus-one method [28].

In this implementation, we use LinearSVC as the base classifier in the OVR framework. The LinearSVC configuration includes the following parameters. 1) class_weight=‘balanced’ is used to address class imbalance by assigning appropriate weights to each class. 2) max_iter=5000 increases the maximum number of iterations to ensure model convergence. 3) dual=False uses the primal formulation of SVM, which is more efficient for cases in which the number of samples is larger than the number of features. 4) tol=1e-4 indicates tolerance for stopping criteria.

The main challenge with the OVR method is the imbalance between positive and negative classes, especially as the number of classes increases. This imbalance can lead to biased classifiers that favour the majority class, resulting in poor classification performance for minority classes [29]. To address this issue, we use the class_weight=‘balanced’ parameter in LinearSVC. We also apply GridSearchCV to search for the optimal value of the C parameter in LinearSVC, with a range of values [0.01, 0.1, 1, 10]. The C parameter controls the trade-off between achieving a low margin and minimising classification errors.

To further optimise model performance, we apply threshold optimisation techniques. This process involves searching for the optimal threshold to convert the output of the decision function into binary predictions. The threshold is optimised in the range of 0.1 to 0.9 to obtain the best F1 score, allowing flexibility in balancing precision and recall [30], [31]. This approach allows the model to handle the complexity of multilabel classification in patent data effectively while maintaining computational efficiency and model interpretability.

D. Data Collection and Processing Module

The data collection and processing module is responsible for collecting and processing patent data from Google Patents and the Indonesian Patent Database, with a total of 15,099 patent documents. This process involves a series of comprehensive preprocessing steps. Text cleaning is performed by removing HTML tags and non-alphanumeric characters and digits, as well as converting all text to lowercase. Stopwords are removed using a combination of 936 English and Indonesian stopwords from NLTK. International Patent Classification codes are processed by extracting sections, classes and subclasses, as well as filtering codes with a minimum of 200 samples. The processed titles and abstracts are combined into a single ‘preprocessed_text’ field for further analysis. This approach ensures that the data used have been cleaned, standardised and optimised for multilabel classification, increasing the potential for model accuracy and reliability [25], [32].

E. Multilabel Classification Model Development Module

The multilabel classification model development module focuses on converting IPC codes into a multilabel format using MultiLabelBinarizer and developing classification models. The processed data are split into training and validation sets. Then, the TF–IDF vectoriser is used with the parameters max_features=20000 and ngram_range=(1, 2) for feature extraction [9]. The main model used is OneVsRestClassifier with LinearSVC as the base classifier. LinearSVC is configured with class_weight=‘balanced,’ max_iter=5000, dual=False and tol=1e-4. Cross-validation with GridSearchCV is used for hyperparameter optimisation, with the F1 score (micro and macro averages) as the main evaluation metric. This approach allows the model to effectively handle the complexity of multilabel classification in patent data while maintaining computational efficiency and model interpretability [30], [31].

F. Model Optimisation Module

The model optimisation module focuses on improving the performance of multilabel classification models through experiments with various parameter combinations [33]. This module uses GridSearchCV to search for the optimal value of the C parameter in LinearSVC, with a range of values [0.01, 0.1, 1, 10]. Additionally, threshold optimisation is performed to convert the output of the decision function into binary predictions, with the threshold optimised in the range of 0.1 to 0.9. Threefold cross-validation is used to assess the effectiveness of each configuration. Evaluation results, especially the F1 scores (micro and macro), are saved and analysed for each parameter combination. The optimal model configuration is selected based on the balance between model performance and computational efficiency [24], [34]. This approach allows for better model adjustment to the specific characteristics of the patent dataset, thereby improving overall classification accuracy.

G. Feature Extraction Using TF–IDF

The method that determines how often each word appears in one document component, called term frequency (TF), and how rarely it occurs in all document components, called inverse document frequency (IDF), is the inverse of the TF document [12]. To calculate weights, the TF–IDF method combines two ideas: the frequency of a word appearing in a particular document and the inverse frequency of documents containing that word. The tf value is divided by the frequency of the most frequently occurring words in the document. This process ensures that the most frequently occurring words obtain the highest if value, which is 1, and that the least frequently occurring words obtain values between 0.5 and 1 [35].

$$if = 0{,}5 + 0{,}5 \text{ x} \frac{tf}{\max(tf)}. \quad (1)$$

Weighting is used with the TF–IDF formula in research conducted with the equation formula from several previous research sources [22].

$$\mathrm{W}_{\mathrm{t,d}} = TF_{\mathrm{t.d}} \text{ x } IDF_{\mathrm{t,d}} = TF_{\mathrm{t,d}} \text{ x}(\log(\frac{N}{dft})) \quad (2)$$

The TF–IDF formula is very important for document analysis because it gives higher values to words that appear frequently in one document but rarely appear in other documents. Eq. (2), representing the change in IDF using log(1 + N/dft), prevents division by zero problems or negative logarithms when dft approaches or equals N. This change ensures that the IDF weight remains well defined, even if a word appears in all documents (preventing IDF from becoming zero or negative), providing stability to TF–IDF weights in real applications.

$$\mathrm{W}_{\mathrm{t,d}} = TF_{\mathrm{t.d}} \text{ x } IDF_{\mathrm{t,d}} = TF_{\mathrm{t,d}} \text{ x}(\log(1+\frac{N}{dft})). \quad (3)$$

In Eq. (3), which represents the change in IDF using log(1 + N/dft), prevents division by zero problems or negative logarithms when dft approaches or equals N. This change ensures that the IDF weight remains well defined, even if a word appears in all documents (preventing IDF from becoming zero or negative), providing stability to TF–IDF weights in real applications.

$$\mathrm{W}_{\mathrm{t,d}} = TF_{\mathrm{t.d}} \text{ x } IDF_{\mathrm{t,d}} = TF_{\mathrm{t,d}} \text{ x}(\log(\frac{N}{1+dft})). \quad (4)$$

To generate a new score, the code-mixed relevance score modifies the TF–IDF score, and weighting and normalisation are applied to obtain the final feature vector EF [36].

H. Model Evaluation

In the new implementation, model evaluation uses LinearSVC as the base classifier in the OneVsRestClassifier framework, replacing the previously used random forest. This method is effective for multilabel classification, in which each instance can have more than one label. Model evaluation is performed using several main metrics, which are as follows. 1) The F1 score (micro and macro averages) is the harmonic mean of precision and recall, providing an overall picture of model performance. F1 = 2 * (precision * recall) / (precision + recall). 2) The classification report provides a summary of the precision, recall and F1 scores for each class. 3) Threshold optimisation optimises the threshold to convert the output of the decision function into binary predictions [37].

The evaluation process also involves GridSearchCV for hyperparameter tuning, specifically the C parameter of LinearSVC. Threefold cross-validation is used to assess model reliability across different subsets of the data. The main evaluation metrics used are as follows:

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN}, \quad (5)$$

$$Precession = \frac{TP+TN}{TP+FP+TN+FN}, \quad (6)$$

$$Recall = \frac{TP}{TN+FN}, \quad (7)$$

$$F1 - Measure = 2x(\frac{prec\ x\ rec}{prec+ewc}), \quad (8)$$

Where TP = true positive, TN = true negative, FP = false positive and FN = false negative.

This evaluation approach allows for a comprehensive assessment of model performance in the context of multilabel classification of patent documents, focusing on the balance between precision and recall represented by the F1 score [38].

I. Query and Prediction Module

The query and prediction module provides an interface for new patent input and performs IPC code predictions. Input data undergo preprocessing consistent with the previous module. The trained OneVsRestClassifier model with LinearSVC is applied for prediction, involving TF–IDF transformation, model application, conversion to probabilities and application of the optimal threshold. Relevant documents are retrieved based on the predicted IPC codes and user keywords, allowing for efficient searching in the patent database [26], [34].

J. Presentation of Results Module

The results presentation module presents a concise overview of relevant patent literature. This module displays related patent documents, key information, predicted IPC codes with confidence levels, matching keywords and visualisation of the IPC code distribution. Using automatic summarisation techniques, this module generates brief but informative summaries of each relevant patent document, facilitating an efficient literature review process and enabling quick identification of the most relevant patents [25], [10].

IV. Results and Discussion

The implementation of OneVsRestClassifier with LinearSVC as the base classifier for multilabel patent classification has yielded promising results. The model achieved a micro F1 score of 0.6723 and a macro F1 score of 0.6009, indicating reasonably good overall performance across various patent categories. These scores suggest that the model has a balanced capability in handling both frequent and rare patent classes, although there is still room for improvement.

Such balanced performance aligns with the broader literature on patent classification complexities [7], in which heterogeneous technology domains often require the careful handling of imbalanced labels.

In comparison to earlier approaches, hybrid methods (e.g. KNN+fastText) [9] and fine-tuned transformer-based models (e.g. BERT and XLNet) [3] have been explored by prior work on monolingual patent classification. While these studies report competitive or even state-of-the-art F1 metrics on single-language datasets, they do not address bilingual corpora (e.g. Indonesian–English). By contrast, our approach handles cross-lingual patent data and addresses class imbalance, thereby filling a gap not extensively covered in previous work.

The hyperparameter optimisation process, using GridSearchCV, identified an optimal C parameter of 0.1 for LinearSVC. This relatively low value indicates that the model prefers a large margin, potentially enhancing its generalisation capability. Interestingly, the threshold optimisation process found that the default threshold of 0.5 was optimal for converting probabilities into binary predictions, suggesting that the raw predictions of the model are well calibrated.

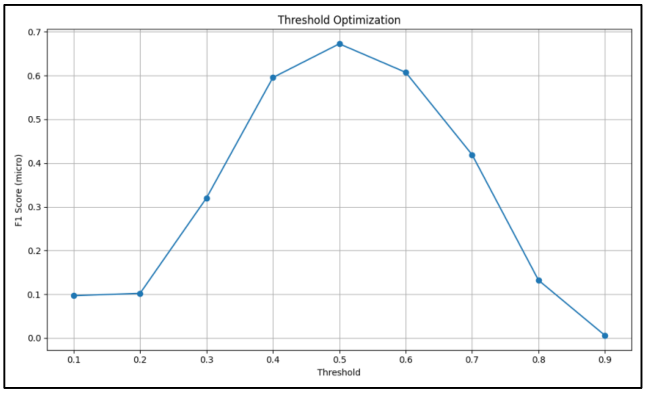

Fig. 3. Performance analysis of IPC patents.

Performance analysis by class revealed significant variations among the different IPC classes. Some categories, such as C22C (Alloys) and A61K (Preparations for medical, dental or toilet purposes), showed very good performance, with an F1 score of 0.88. This result suggests that these categories may have distinct features or terminology that the model can effectively identify. Conversely, categories such as Y02A (Technologies for adaptation to climate change) and G06N (Computer systems based on specific computational models) showed lower performance, with F1 scores of 0.12 and 0.20, respectively. These differences highlight the challenges in handling the inherent complexity and potential imbalances in patent data across various technology domains.

The feature extraction approach, using TfidfVectorizer with 20,000 features and including bigrams, appears to have captured important nuances in the patent texts. The decision to focus on thorough text cleaning and stopword removal, rather than stemming, seems effective, as evidenced by the overall model performance. However, the varying performance across classes suggests that there might be room for further refinement of the feature extraction process for certain technology domains.

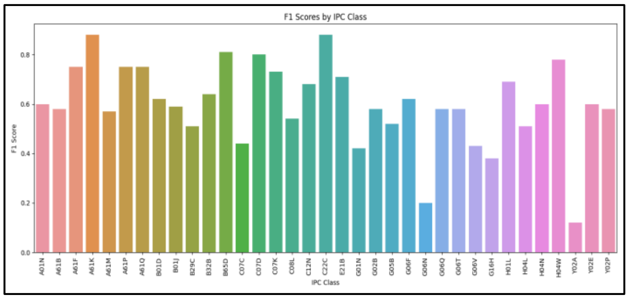

Fig. 4. F1 scores by IPC class.

One of the strengths of this research is its handling of a bilingual dataset comprising 15,097 patent documents from both Indonesian and English sources. The ability of the model to perform reasonably well on this combined dataset demonstrates its potential for handling multilingual patent classification tasks, which is particularly relevant in the context of international patent systems.

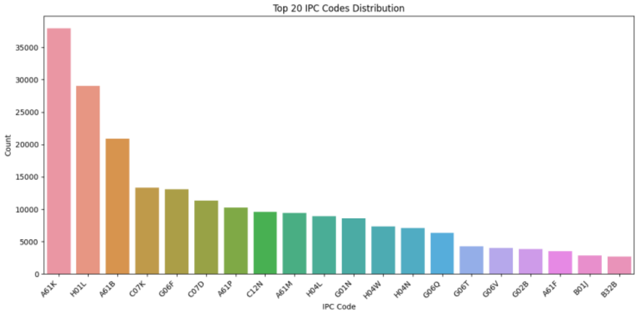

Fig. 5. Distribution of feature extraction results for the top 20 IPC codes.

The computational efficiency of the model is quite good, with a total execution time of about 300 s for processing the entire dataset. This result suggests that the methodology could be scalable for larger datasets, although further testing is needed to confirm this.

While the current methodology shows improvements over previous approaches, particularly in terms of classifier choice and feature extraction, it is important to note that challenges remain. The significant variation in F1 scores across classes indicates that class imbalance and the complexity of certain technological fields continue to pose difficulties. Categories such as G06N and G16H (Healthcare informatics) appear more challenging to classify, possibly because of their interdisciplinary nature or rapidly evolving terminology. Prior reviews confirm that specialised jargon and evolving concepts in the AI or healthcare domains consistently hamper straightforward classification [3], [8].

These results suggest that while the model shows promise in automating aspects of patent classification, it may be most effectively used as a supportive tool in the classification process rather than a standalone solution. For categories with strong performance, the model could potentially streamline the classification process, while for more challenging categories, it could serve as an initial filter, with human experts providing the final classification. Future work could focus on addressing performance disparities across different patent categories. This process can involve exploring more advanced NLP techniques, such as BERT or domain-specific language models pretrained on patent data. Additionally, investigating techniques to improve performance in low-scoring classes, such as oversampling or developing class-specific features, could yield further improvements. Such strategies align with contemporary research calling for data augmentation and specialised embeddings to enhance multilabel patent classification [14].

In conclusion, while the current methodology demonstrates good potential in tackling the complex task of multilabel patent classification across languages, there remains room for improvement. The performance of the model suggests that it could be a valuable tool in assisting patent classification processes, potentially enhancing efficiency and consistency in international patent classifications. However, further research and refinement are needed to address the challenges identified, particularly in handling the diverse and evolving nature of technological innovations reflected in patent documents.

V. Conclusion

This research developed and evaluated a multilabel classification model for patent documents using a machine learning approach. The OneVsRestClassifier model with LinearSVC as the base classifier demonstrated competitive performance, achieving a micro F1 score of 0.6723 and a macro F1 score of 0.6009. These results indicate the potential of the model to handle the complexity of multilabel and multilanguage patent classification.

In contrast to the hybrid KNN–fastText approach proposed by Yadrintsev and Sochenkov [9], which showed improved classification results on Russian and English texts through a stacking meta-classifier, our work specifically addressed bilingual data (Indonesian–English) and class imbalance in the IPC. Similarly, Haghighian Roudsari et al. [3] leveraged BERT, XLNet and other transformer-based models for multilabel patent classification but focused on monolingual English corpora. Our framework addressed this gap by targeting cross-lingual challenges and imbalanced labels within a single methodology, allowing for the robust handling of diverse patents.

The use of TfidfVectorizer with 20,000 features, including bigrams, proved effective in capturing important nuances in patent texts, although there is still room for refinement. Performance analysis revealed variations across IPC classes, indicating the need for targeted improvements in lower-performing categories (e.g. Y02A). Nevertheless, several limitations remain, which are as follows:

1) Vocabulary coverage: The TF–IDF approach, while effective, may not fully capture deep contextual or semantic relationships.

2) Data scope: This study focuses on Indonesian–English patents. Extending to additional languages or specialised subfields may require further adaptation.

3) Class imbalance handling: Although weighted learning helps mitigate skew, advanced sampling or data augmentation strategies could further improve performance for rare IPC codes.

Despite these limitations, this research contributes to the development of an automated patent classification system that has the potential to increase efficiency in patent analysis. Although the results are promising, it is important to remember the crucial role of human experts, especially for highly specialised IPC classes. With further refinements, the methodology outlined here can become a valuable supporting tool in the patent classification process, facilitating effective intellectual property management. This work paves the way for further progress in automated patent classification, addressing the multilingual and imbalanced data challenges inherent in international patent systems.

References

[1] E. Sharma, C. Li, and L. Wang, “BigPatent: A large-scale dataset for abstractive and coherent summarization,” ACL 2019 - 57th Annu. Meet. Assoc. Comput. Linguist. Proc. Conf., pp. 2204–2213, 2020, doi: 10.18653/v1/p19-1212.

[2] S. Casola and A. Lavelli, “Summarization, simplification, and generation: The case of patents,” Expert Syst. Appl., vol. 205, 2022, doi: 10.1016/j.eswa.2022.117627.

[3] A. Haghighian Roudsari, J. Afshar, W. Lee, and S. Lee, “PatentNet: Multi-label classification of patent documents using deep learning based language understanding,” Scientometrics, vol. 127, no. 1, pp. 207–231, 2022, doi: 10.1007/s11192-021-04179-4.

[4] Y. Yoo, T.-S. Heo, D. Lim, and D. Seo, “Multi label classification of artificial intelligence related patents using modified D2SBERT and sentence attention mechanism,” 2023, https://arxiv.org/abs/2303.03165

[5] B. S. Haney, “Patents for NLP software: An empirical review,” SSRN Electron. J., 2020, doi: 10.2139/ssrn.3594515.

[6] H. S. Al-Khalifa, T. AlOmar, and G. AlOlyyan, “Natural language processing patents landscape analysis,” Data, vol. 9, no. 4, 2024, doi: 10.3390/data9040052.

[7] A. Abbas, L. Zhang, and S. U. Khan, “A literature review on the state-of-the-art in patent analysis,” World Pat. Inf., vol. 37, pp. 3–13, 2014, doi: 10.1016/j.wpi.2013.12.006.

[8] R. S. Eisenberg, “Patents and data-sharing in public science,” Ind. Corp. Chang., vol. 15, no. 6, pp. 1013–1031, 2006, doi: 10.1093/icc/dtl025.

[9] V. V Yadrintsev and I. V Sochenkov, “The hybrid method for accurate patent classification,” Lobachevskii J. Math., 2019. [Online]. Available: https://link.springer.com/article/10.1134/S1995080219110325

[10] H. Zhu, C. He, Y. Fang, B. Ge, M. Xing, and W. Xiao, “Patent automatic classification based on symmetric hierarchical convolution neural network,” Symmetry (Basel)., vol. 12, no. 2, pp. 1–12, 2020, doi: 10.3390/sym12020186.

[11] C. Diaz-Asper, M. K. Hauglid, C. Chandler, A. S. Cohen, P. W. Foltz, and B. Elvevåg, “A framework for language technologies in behavioral research and clinical applications: Ethical challenges, implications, and solutions,” Am. Psychol., vol. 79, no. 1, pp. 79–91, 2024, doi: 10.1037/amp0001195.

[12] F. Bacinger, I. Boticki, and D. Mlinaric, “System for semi-automated literature review based on machine learning,” Electron., vol. 11, no. 24, 2022, doi: 10.3390/electronics11244124.

[13] P. H. Santoso, E. Istiyono, Haryanto, and W. Hidayatulloh, “Literature using machine learning,” Data, vol. 7, pp. 1–41, 2022.

[14] Y. Zhang and Z. Lu, “Exploring semi-supervised variational autoencoders for biomedical relation extraction,” Methods, vol. 166, no. November 2018, pp. 112–119, 2019, doi: 10.1016/j.ymeth.2019.02.021.

[15] W. Weng, D. H. Wang, C. L. Chen, J. Wen, and S. X. Wu, “Label specific features-based classifier chains for multi-label classification,” IEEE Access, vol. 8, pp. 51265–51275, 2020, doi: 10.1109/ACCESS.2020.2980551.

[16] A. Kravets, N. Shumeiko, B. Lempert, N. Salnikova, and N. Shcherbakova, “‘Smart queue’ approach for new technical solutions discovery in patent applications,” Commun. Comput. Inf. Sci., vol. 754, pp. 37–47, 2017, doi: 10.1007/978-3-319-65551-2_3.

[17] X. Yu and B. Zhang, “Obtaining advantages from technology revolution: A patent roadmap for competition analysis and strategy planning,” Technol. Forecast. Soc. Change, vol. 145, no. October, pp. 273–283, 2019, doi: 10.1016/j.techfore.2017.10.008.

[18] A. Khurana and V. Bhatnagar, “Investigating entropy for extractive document summarization,” Expert Syst. Appl., vol. 187, 2022, doi: 10.1016/j.eswa.2021.115820.

[19] F. Zhu, X. Wang, D. Zhu, and Y. Liu, “A supervised requirement-oriented patent classification scheme based on the combination of metadata and citation information,” Int. J. Comput. Intell. Syst., vol. 8, no. 3, pp. 502–516, 2015, doi: 10.1080/18756891.2015.1023588.

[20] S. Sarica, J. Luo, and K. L. Wood, “TechNet: Technology semantic network based on patent data,” Expert Syst. Appl., vol. 142, p. 112995, 2020, doi: 10.1016/j.eswa.2019.112995.

[21] N. Shibayama, R. Cao, J. Bai, W. Ma, and H. Shinnou, “Evaluation of pretrained {BERT} model by using sentence clustering,” Proc. 34th Pacific Asia Conf. Lang. Inf. Comput., pp. 279–285, 2020. [Online]. Available: https://aclanthology.org/2020.paclic-1.32

[22] N. Febriyanti, D. Palupi, and O. Arsalan, “Text similarity detection between documents using case based reasoning method with cosine similarity measure (case study SIMNG LPPM Universitas Sriwijaya),” vol. 3, no. 2, pp. 36–45, 2022.

[23] I. O. Suzanti, A. Jauhari, N. Hidayanti, I. Y. Harianti, and F. A. Mufarroha, “Comparison of stemming and similarity algorithms in Indonesian translated Al-Qur’an text search,” J. Ilm. Kursor, vol. 11, no. 2, pp. 91–91, 2022. [Online]. Available: http://kursorjournal.org/index.php/kursor/article/view/280

[24] J. Jang and C. O. Kim, “One-vs-Rest network-based deep probability model for open set recognition,” 2020, http://arxiv.org/abs/2004.08067

[25] X. Chen and N. Deng, “A semi-supervised machine learning method for chinese patent effect annotation,” Proc. - 2015 Int. Conf. Cyber-Enabled Distrib. Comput. Knowl. Discov. CyberC 2015, pp. 243–250, 2015, doi: 10.1109/CyberC.2015.99.

[26] R. Ros, E. Bjarnason, and P. Runeson, “A machine learning approach for semi-automated search and selection in literature studies,” ACM Int. Conf. Proceeding Ser., vol. Part F1286, pp. 118–127, 2017, doi: 10.1145/3084226.3084243.

[27] M. Abazar, P. Masjedi, and M. Taheri, “A binary relevance adaptive model-selection for ensemble steganalysis,” ISeCure, vol. 14, no. 1, pp. 105–113, 2022, doi: 10.22042/isecure.2021.262990.596.

[28] Y. Liu, “Yang Liu (刘洋),” p. 86, 2010.

[29] H. Sasaki and I. Sakata, “Identifying potential technological spin-offs using hierarchical information in international patent classification,” Technovation, vol. 100, no. September 2019, p. 102192, 2021, doi: 10.1016/j.technovation.2020.102192.

[30] J. Read, B. Pfahringer, G. Holmes, and E. Frank, “Classifier chains: A review and perspectives,” J. Artif. Intell. Res., vol. 70, pp. 683–718, 2021, doi: 10.1613/JAIR.1.12376.

[31] W. Chmielnicki and K. Stąpor, “Using the one-versus-rest strategy with samples balancing to improve pairwise coupling classification,” Int. J. Appl. Math. Comput. Sci., vol. 26, no. 1, pp. 191–201, 2016, doi: 10.1515/amcs-2016-0013.

[32] M. Suzgun, L. Melas-kyriazi, and S. K. Sarkar, “The Harvard USPTO Patent Dataset: Corpus of patent applications,” no. Ml, pp. 1–38, 2020.

[33] Z. Wang, T. Wang, B. Wan, and M. Han, “Partial classifier chains with feature selection by exploiting label correlation in multi-label classification,” Entropy, vol. 22, no. 10, pp. 1–22, 2020, doi: 10.3390/e22101143.

[34] M. S. Hajmohammadi, R. Ibrahim, and A. Selamat, “Bi-view semi-supervised active learning for cross-lingual sentiment classification,” Inf. Process. Manag., vol. 50, no. 5, pp. 718–732, 2014, doi: 10.1016/j.ipm.2014.03.005.

[35] R. T. Wahyuni, D. Prastiyanto, and E. Supraptono, “Penerapan Algoritma Cosine Similarity dan Pembobotan TF-IDF pada Sistem Klasifikasi Dokumen Skripsi,” vol. 9, no. 1, 2017.

[36] R. Sharma and P. Shrinath, “Ensemble of weighted code mixed feature engineering and machine learning-based multiclass classification for enhanced opinion mining on unstructured Data,” vol. 15, no. 10, pp. 1220–1230, 2024.

[37] A. Alshammari, F. Alotaibi, and S. Alnafrani, “Prediction of outpatient no-show appointments using machine learning algorithms for pediatric patients in Saudi Arabia,” Int. J. Adv. Comput. Sci. Appl., vol. 15, no. 8, pp. 108–116, 2024, doi: 10.14569/IJACSA.2024.0150812.

[38] Dafid, Ermatita, and Samsuryadi, “A framework for predicting academic success using classification method through filter-based feature selection,” Int. J. Adv. Comput. Sci. Appl., vol. 14, no. 9, pp. 435–444, 2023, doi: 10.14569/IJACSA.2023.0140947.