INTEGRATING NATURAL LANGUAGE PROCESSING WITH EXPERT SYSTEMS FOR STREAMLINED EVALUATION OF APPLICATIONS

ARUP MOHANTY

A THESIS SUBMITTED TO

THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF MASTER OF ARTS

GRADUATE PROGRAM IN INFORMATION SYSTEMS AND TECHNOLOGY

YORK UNIVERSITY

TORONTO, ONTARIO

SEPTEMBER 2025

Abstract

Screening applications for jobs, education, and grants is often slow, subjective, and inefficient. To address these challenges, the Hybrid AI Platform for Streamlining Evaluation (HAIPSE) integrates expert rule-based reasoning with NLP and computer-vision techniques, offering a structured alternative to traditional tracking systems. Its heuristic scoring module, powered by spaCy, extracts key details and grades responses against predefined criteria. To reduce workload and capture context, HAIPSE employs large language models, Meta Llama 3-8B and Mistral 8×7B, to generate concise essay summaries. Group-fairness metrics and built-in debiasing ensure transparency while embedding fairness checks directly into the pipeline. Trained on both sample IDs and 2,000+ applications from the NIB Trust Fund (Canada), HAIPSE improves efficiency and reduces bias compared with manual reviews. A collaborative audit interface bridges AI automation with expert judgment, reinforcing responsible AI that is scalable, interpretable, and equitable.

Acknowledgements

First and foremost, I would like to express my deepest gratitude to Prof. Peter Khaiter for his exceptional supervision, insightful guidance, and unwavering support throughout my Master's studies. His expert advice and constructive feedback have been invaluable to both my academic development and the successful completion of this thesis. I would also like to express my gratitude towards the members of my examination committee, Prof. Ling Jiang and Prof. Jennifer Pybus for their time and expertise. Lastly, I would like to thank my family and friends for their continuous support that allowed me to complete this milestone successfully.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

Evaluating applications for grants, scholarships, and employment is traditionally a time-consuming and subjective process, often vulnerable to implicit bias and inconsistency. In particular, verifying the authenticity of identification documents and supporting IDs typically requires manual inspection. Such visual checks are time-consuming, susceptible to human errors, and difficult to standardise across reviewers, thereby compromising both efficiency and consistency. Demographic identifiers, or institution names can often subconsciously influence evaluators, leading to unequal treatment of applicants from different groups. Without explicit bias-detection mechanisms, such disparities may persist undetected. Existing applicant-tracking platforms were not developed for a large-scale application submission. As application numbers grow, performance degradation (e.g., slower query times, batch-processing delays) can jeopardise user experience and overall process integrity. Organizations following a traditional rule-based scoring framework to evaluate applications often struggle to adapt to evolving priorities (e.g., new diversity goals) or emerging evaluation criteria, necessitating frequent — and often massive — manual recalibration. Modern funding and scholarship, routinely attract thousands of applications, whereas the reviewing staff often consists of only a handful of program evaluators. This volume–capacity mismatch produces prolonged evaluation cycles, substantial backlogs, and scheduling bottlenecks that can delay time-critical funding or hiring decisions.

These challenges are particularly profound in the case of the NIB Trust Fund Canada (https://fgfoundation.smapply.io), our partner organization, which annually receives thousands of individual and group applications aimed at supporting educational and healing initiatives by providing scholarships to the First Nations communities impacted by the Indian Residential School system. Despite its critical mission, the NIB relies on a limited team of 4–7 program officers, making the evaluation process resource-intensive and time-consuming, often taking up to six months to complete the cycle.

To address these issues and limitations, the thesis introduces HAIPSE (Hybrid AI Platform for Streamlining Evaluation), a comprehensive AI-driven framework designed to transform the application review process. The HAIPSE integrates multiple specialized modules, including a User

Interface, Natural Language Processing (NLP), Expert System, Computer Vision, Cohort Fairness Scoring, Large Language Models (LLMs), and Human Feedback. Together, these components work to enhance transparency, efficiency, and fairness in decision-making, while reducing the burden on human reviewers (Dwork et al., 2012; Friedler et al., 2019).

Central to the HAIPSE is the Expert System Module, which houses a knowledge base of domain-specific heuristic rules derived from human experts (Feigenbaum, 1982; Jackson, 1999). An inference engine of the Expert System applies these rules by matching keywords and facts from the application documents, determining eligibility of the applicants and the scoring outcome of their applications (Waterman, 1986). This process is supported by a Heuristic Scoring Module and fuzzy logic, allowing the system to assign partial credits for applicants' qualifications that nearly meet the set up thresholds, thus avoiding rigid binary judgments and improving fairness (Zimmermann, 2001; Mehrabi et al., 2021).

Another key innovation of the HAIPSE is the Total Score measure, which combines structured evaluations derived from expert rules and NLP with scores from the Cohort Fairness Module and LLM-generated summaries (Bansal et al., 2021). Features extracted from application essays and forms are converted into point values using domain-specific logic, and program administrators can fine-tune scoring parameters based on evolving priorities.

To verify the authenticity of identification documents, HAIPSE employs a state-of-the-art Computer Vision module based on the YOLOV11 model, known for its real-time object detection accuracy. This ensures that blurry, tampered, or incomplete IDs are flagged early in the evaluation process (Redmon et al., 2016).

Fairness in HAIPSE means evaluating every applicant with equal dignity and opportunity, ensuring decisions are transparent, unbiased and inclusive, it is not treated as an afterthought but is instead embedded as a core feature of the system (Dwork et al., 2012). The Cohort Fairness Score module assigns the score value consistency across application batches (Friedler et al., 2019). High variance of the scores alerts administrators about potential bias or inconsistencies, prompting retraining or adjustments of the underlying algorithms (Hardt et al., 2016; Raji et al., 2020). As the system learns from human feedback and overrides patterns, its evaluations become more accurate and equitable over time (Holstein et al., 2019).

HAIPSE's evolving hybrid architecture adapts to emerging biases through continuous updates, mainly by the humans who are responsible for the end decision on the applications. Drawing from critique of static AI systems (Crawford and Paglen, 2021), the HAIPSE refines its rules and retrains its models using real-world feedback, such as disparities in cohort evaluations. By grouping applications into batches of 30, it applies comparative fairness principles (Friedler et al., 2019) to minimize contextual biases like reviewer fatigue (Dwork et al., 2012). This continuous refinement ensures that the platform dynamically aligns with evolving societal standards, transitioning from temporary adjustments (Zemel et al., 2013) to a sustainable, adaptive framework.

A fine-tuned NLP pipeline, powered by leading LLMs like Meta LLaMA 3-8B (Meta AI, 2020) and Tiiuae Falcon 7B (Almazrouei et al., 2023), condenses multi-page essays into summaries that are both precise and context-aware (Liu & Lapata, 2019; Celikyilmaz et al., 2020). These generated summaries significantly reduce inter-reviewer variability, allowing evaluators to make faster and more consistent decisions (Bhandari et al., 2020; Fabbri et al., 2021). Standardized outputs ensure that all applications are reviewed through a consistent lens, minimizing subjective bias (Maynez et al., 2020; Zhang et al., 2023).

Importantly, the HAIPSE does not replace human judgment but augments it (Amershi et al., 2019). Reviewers remain embedded in the loop, particularly for complex or ambiguous cases. Overrides and feedback mechanisms are logged and used to continuously refine the system (Holstein et al., 2019; Raji et al., 2020). This collaborative approach enhances both decision quality and trust (Binns et al., 2018). The HAIPSE also operationalizes algorithmic accountability: every automated decision can be traced, and justifications for overrides are documented (Kroll et al., 2017). The framework and its modules are validated using over 2,000 real-world applications from the NIB Trust Fund, demonstrating its scalability and practical impact. The specific features of the HAIPSE can be summarized as follows:

1. The HAIPSE presents a hybrid application evaluation framework that uniquely combines rule-based Expert System, NLP, Large Language Models, and Computer Vision based on the YOLOV11 model.

2. It ensures transparent and structured scoring by integrating group fairness metrics and LLM-generated summaries that capture nuanced application context.
3. It pioneers in bias mitigation through proactive fairness checks and real-time debiasing mechanisms.
4. It promotes a human-AI partnership, bridging automation and judgment through a collaborative auditing process.

Historically, manual screening processes allowed for nuanced human judgment but also introduced inconsistencies and inefficiencies (Cappelli, 2019). As application volumes increase, reviewer fatigue and uneven evaluations become more common (Van Esch et al., 2019). The HAIPSE, grounded in the principles of responsible AI (Barocas & Selbst, 2016), offers a way forward, one that enhances equity, transparency, and efficiency in high-stakes decisions.

The corresponding research contributions are as follows:

RC1: The HAIPSE platform is proposed implementing a hybrid AI methodology that unifies NLP extraction, rule-based reasoning, generative-AI summarisation, and human oversight for fair application evaluation.

RC2: A transparent, adjustable heuristic-plus-fuzzy scoring engine is designed that assigns audit-ready point values and can be retuned instantly as programme priorities evolve.

RC3: YOLOV11 model-based computer vision ID verification is integrated, eliminating manual document checks while preserving security and data integrity.

RC4: A sophisticated NLP pipeline is implemented that converts free-form essays into structured, bias-aware features.

RC5: A human-AI collaboration workflow is formalized that delegates routine analysis to AI while retaining human authority for overrides, boosting speed without sacrificing contextual judgment.

RC6: Seven LLMs for summarising applications are benchmarked whereby Meta-Llama-3-8B and Falcon-7B demonstrated better performance in reducing inter-reviewer variance.

RC7: The Cohort Fairness Score, a variance-based metric is proposed that automatically detects batch-level bias and triggers retraining or rule refinement.

The key research questions addressed in this study are:

[RQ1] How to integrate Natural Language Processing (NLP), rule-based Expert System, Computer Vision, Large Language Models (LLMs) and Cohort Fairness scoring within a comprehensive AI-driven framework to enhance the transparency and precision of application evaluations?

[RQ2] What is the best Large Language Model, generating concise essay summaries, to fit the specific tasks of the HAIPSE?

[RQ3] Which scoring methods are to be applied within the HAIPSE evaluation pipeline to ensure debiasing features, proactive fairness and greater transparency in real time application processing cycle?

[RQ4] How does the HAIPSE's adaptive learning loop improve the accuracy and equity of scoring across successive evaluation cohorts?

[RQ5] How does the human-AI collaborative model implemented in the HAIPSE contribute to more trustworthy, explainable, and context-sensitive decision-making?

The subsequent chapters are organized as follows. Chapter 2 presents a literature review and bibliometric analysis to identify potential gaps in the area of research. Chapter 3 details the methodology we adapted and applied in the HAIPSE development. Chapter 4 describes the experimental results and offers their analysis. Finally, Chapter 5 provides conclusions and outlines directions for future work.

Chapter 2: Literature Review and Bibliometric Analysis

The objectives of this chapter are to define key terms for the research and to provide an overview of the current state-of-the-art within the problem domain in the form of literature review and bibliometric analysis.

The integration of Natural Language Processing (NLP) with Expert Systems presents a significant advancement in the field of information systems and artificial intelligence (AI). This convergence aims to enhance decision-making processes by leveraging the power of language understanding and generation.

The review aims at assessing the required NLP techniques that can automatically extract relevant information from vast amounts of structured text data, helping to organize and process large amounts of data to facilitate decision support systems in different practical applications. The primary goal of the review is to identify the research gap and provide a summary of the findings from the bibliometric analysis and literature review. Figure 1 shows the roadmap of the conducted literature review.

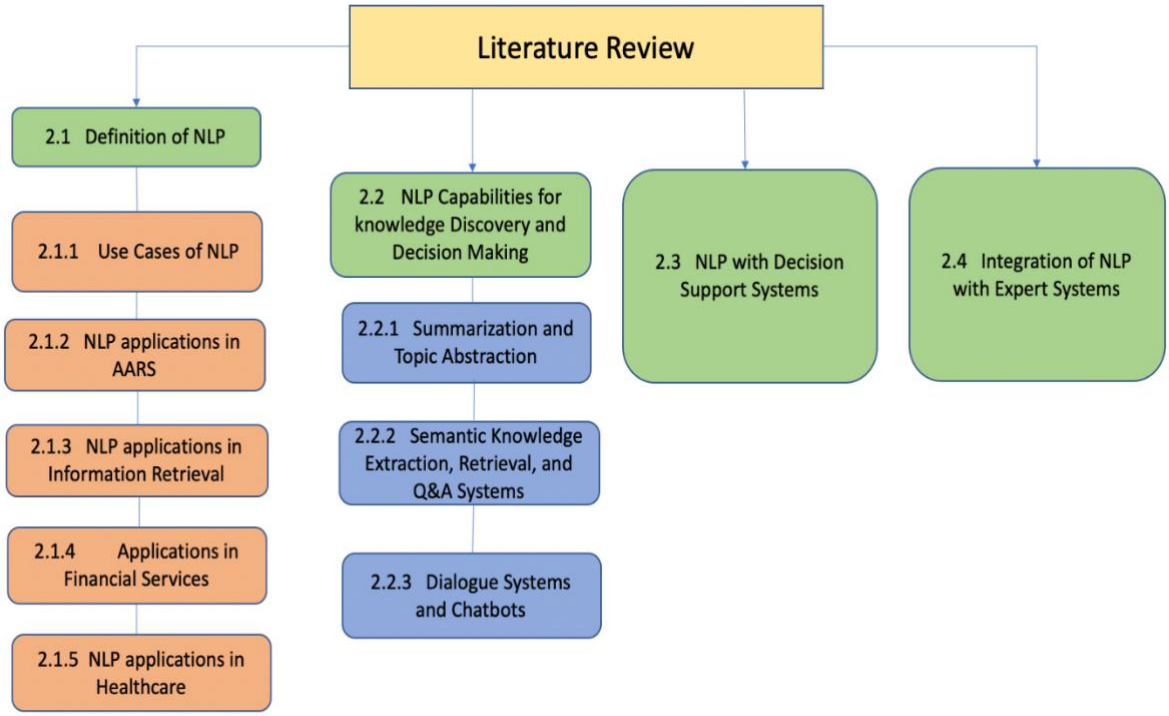


Figure 1. Roadmap for the Literature Review.

2.1 Definition of Natural Language Processing

We start with providing necessary definitions and outlining the roles of natural language processing (NLP). NLP is generally an area within AI and computer science that tends to the connection between computers and human languages (Reshamwala et al., 2013; Agarwal, 2019; Mishra and Kumar, 2020). It is generally focused on making it possible for computers to understand, interpret, and generate natural languages in ways that are meaningful. NLP involves a variety of approaches and tactics for the analysis and manipulation of text and speech. It has many applications in machine translation, text summarization, speech recognition, sentiment analysis, and information retrieval. What made this field possible was combining techniques from different areas: linguistics, statistics, and machine learning (ML).

2.1.1 Use Cases of NLP

NLP has a broad range of real-world applications across various sectors, each offering unique opportunities and challenges. In the logistics sector, NLP can significantly enhance supply chain management by automating and optimizing processes, such as inventory tracking, demand forecasting, and supplier communication. However, the implementation of NLP in this field is not without difficulties. Issues, such as data integration, system compatibility, and the need for high-quality, domain-specific language models, remain challenges that must be addressed for effective deployment (Garg et al., 2021).

In the education sector, NLP applications are increasingly being used to support teacher training and development. These applications include voice emotion analysis, which helps in understanding and responding to students' emotional states; assistive technology, which provides support for diverse learning needs; language learning tools that aid in teaching and learning new languages; and intelligent analysis for educational content. These NLP tools have the potential to greatly enhance the quality of education by providing personalized feedback, improving teaching methods, and supporting professional development for educators (Zhu, 2023).

NLP also plays a valuable role in software development, particularly in textual use case analysis. By automating the analysis of requirements and aiding in the design phase, NLP can save significant time and effort. Despite these benefits, the effectiveness of NLP in this domain heavily

depends on selecting the appropriate processing methods and pipelines tailored to the specific use cases. The variability in use cases across different domains means that the choice of NLP techniques must be carefully considered to achieve optimal results. Evaluations of syntactic and semantic pipelines at theoretical and practical levels have shown promising outcomes, suggesting that further exploration and refinement of task-specific NLP pipelines could enhance their effectiveness (Kulkarni et al., 2012).

NLP offers transformative potential across logistics, education, and software development, having regarded that each sector faces unique implementation challenges. The success of NLP applications in these areas depends on overcoming these challenges and leveraging the right techniques and tools tailored to specific needs. Continued research and practical evaluations will be crucial in realizing the full benefits of NLP in these diverse domains.

2.1.2 NLP Applications in Automated Application Review Systems (AARSs)

NLP has become a crucial technology for automating various facets of application review systems, streamlining tasks that were traditionally manual and time-consuming. Recent research highlights the effectiveness of NLP in several areas, such as analyzing safety occurrence reports, screening resumes, and automating resume parsing and ranking processes (Sinha et al., 2021). By leveraging NLP techniques, organizations can efficiently extract valuable information from unstructured text, including skills, qualifications, and work experience, which is vital for making informed decisions in recruitment and safety management.

ML models, especially deep learning approaches like Bidirectional Transformers, have significantly advanced text classification and entity extraction tasks (Devlin et al., 2019). These models demonstrate high accuracy in parsing and interpreting complex text data, which enhances the ability to identify relevant information quickly and accurately (Ricketts et al., 2023). In the context of resume evaluation, NLP-powered systems can automate the extraction of critical details and rank candidates based on predefined criteria, thus optimizing the recruitment process. Such systems can be seamlessly integrated with existing Applicant Tracking Systems (ATS), providing real-time insights and improving the efficiency of the recruitment pipeline (Nimbekar et al., 2019).

The integration of NLP into automated resume evaluation systems represents a significant step forward in recruitment technology. These systems can handle large volumes of applications by parsing resumes, extracting pertinent information, and ranking candidates with

greater speed and accuracy than traditional methods. However, while computational accuracy is crucial, it is also important to conduct real-world trials to evaluate the practical effectiveness of these NLP applications. Such trials help in assessing how well these systems perform in diverse and dynamic real-world environments, ensuring that they provide meaningful and reliable support in application review processes (Ricketts et al., 2023).

2.1.3 NLP Applications in Information Retrieval

NLP has shown considerable promise in enhancing Information Retrieval (IR) systems, though its impact has varied across different techniques and applications. Historically, simpler NLP techniques, such as stop word removal, which filters out common, less meaningful words, and stemming, which reduces words to their root forms, have led to significant improvements in IR systems (Brants, 2003). These methods have helped refine the search queries and improve the relevance of retrieved documents by addressing common linguistic variations and redundancies. However, more advanced NLP methods, such as parsing, which involves analyzing the grammatical structure of sentences, and word sense disambiguation, which resolves the meaning of words based on context, have had more limited success in enhancing IR systems (Brants, 2003).

While these techniques offer deeper linguistic insights, their practical benefits in improving search results have not always met expectations, often due to their complexity and computational resources required. The application of NLP in IR has evolved significantly from the early reliance on straightforward keyword matching to the adoption of sophisticated ML and deep learning approaches (Wang, 2020).

ML models, particularly those utilizing large-scale neural networks, have improved the ability to understand and retrieve relevant documents by capturing complex patterns and relationships in textual data. This evolution has been instrumental in various domains, including academic research, medical information retrieval, and e-commerce, where nuanced and context-aware search capabilities are crucial. In the quest to enhance IR systems, some researchers have

focused on leveraging NLP resources like WordNet, a lexical database that organizes words into synonym sets and captures semantic relationships (Smeaton, 1999).

By utilizing these resources, researchers aim to improve IR tasks by enriching the understanding of word meanings and relationships beyond mere keyword matching. Other approaches have proposed frameworks that optimize the retrieval process by employing specific NLP techniques. For example, some models consider only nouns and verbs in the Vector Space Model, a technique used to represent text documents as vectors in a multi-dimensional space (Subhashini & Kumar, 2010). This approach aims to increase retrieval effectiveness by focusing on the most semantically significant parts of the text, thereby enhancing the relevance of search results.

Despite these advancements, challenges remain in the field of IR. The integration of NLP techniques continues to face obstacles, such as the complexity of semantic understanding and the need for personalized retrieval that accurately reflects individual user preferences and contexts (Wang, 2020). Nevertheless, ongoing progress in semantic understanding and personalized retrieval mechanisms holds the promise of more sophisticated and accurate search technologies.

2.1.4 NLP Applications in Financial Services

NLP has been evolved as a pivotal technology in financial services, revolutionizing the sector with its wide array of applications. Traditionally, financial analysis and decision-making relied heavily on structured data, but NLP has expanded the horizon by enabling the processing and analysis of unstructured text data. This transition has been marked by a shift from early rule-based systems, which followed predefined patterns and heuristics, to sophisticated deep learning models capable of understanding and generating human-like text (Du et al., 2025). Such advancements have significantly enhanced the ability to predict financial trends, evaluate financial statements, and gauge market sentiments (Gupta et al., 2020).

In the Financial Technology (FinTech) sector, NLP plays a crucial role in several critical processes. For instance, Know Your Customer (KYC) and Know Your Product (KYP) procedures benefit from NLP's ability to analyze vast amounts of textual data, such as customer reviews, social media posts, and regulatory documents. This analysis helps firms gain deeper insights into their

customers and the products they offer, facilitating more informed decision-making (Gupta et al., 2020).

Similarly, the Sentiment Analysis and Customer (SYC) processes leverage NLP to assess customer feedback and market opinions, which can influence product development and marketing strategies. Moreover, NLP has given rise to a specialized field known as Natural Language Financial Forecasting (NLFF). This field focuses on applying NLP techniques to predict financial markets and stock prices by analyzing textual data from news articles, financial reports, and social media (Du et al., 2025). Despite its potential, NLFF faces challenges such as data scarcity, where limited availability of high-quality textual data can hinder model performance, and adversarial examples, where misleading or manipulative text can distort predictions (Gupta et al., 2020).

2.1.5 NLP Applications in Healthcare

NLP has emerged as a transformative tool in healthcare, offering significant applications and benefits by analyzing unstructured medical data. NLP techniques can effectively process and interpret patient records, clinical notes, and other forms of medical text to extract valuable insights that support clinical decision-making and improve patient care (Sánchez et al., 2024). By converting free-text data into structured information, NLP helps in managing and utilizing medical data more efficiently, which can enhance various healthcare functions, including clinical practice, hospital management, personal care, public health, and drug development (Schopow et al., 2023).

One of the key advantages of NLP in healthcare is its ability to handle vast amounts of medical literature and research findings. This capability aids healthcare practitioners in staying up to date with the latest developments, making informed decisions about diagnosis, treatment plans, and patient care strategies (Rekha, 2023). The growing adoption of NLP technologies in the healthcare sector is reflected in the projected expansion of the global market for NLP solutions, underscoring the increasing importance of these technologies in improving healthcare outcomes (Sánchez et al., 2024).

Despite the promising potential of NLP, there are notable challenges in its implementation within healthcare settings. The complexity and heterogeneity of medical data present significant hurdles, as integrating and standardizing diverse data sources can be difficult (Sánchez et al.,

2024). Addressing these challenges is crucial for leveraging NLP's full potential to enhance patient care and advance medical research. NLP plays a crucial role in transforming unstructured text data from electronic health records (EHRs) into actionable insights, improving the usability and effectiveness of EHR systems (Friedman et al., 2013).

In addition to the technical challenges, there is a need for collaborative efforts to advance NLP applications in healthcare. Developing and improving usability in healthcare applications requires open-source software components and infrastructure that support the sharing of annotated data and tools. Collaborative initiatives can accelerate the development of NLP technologies by providing the resources and data needed for continuous improvement and innovation (Ohno-Machado, 2014).

NLP is set to revolutionize healthcare by improving the management and utilization of medical data, aiding in clinical decision-making, and enhancing patient care. While there are challenges related to data complexity and integration, ongoing research and collaborative efforts are essential to overcome these obstacles and fully realize the benefits of NLP in the healthcare sector. The continued advancement of NLP technologies will play a vital role in shaping the future of medical practice and health information management.

2.2 NLP Capabilities for Knowledge Discovery and Decision Making

In healthcare, NLP can structure narrative information from medical reports and case notes, making it more accessible for decision-making (Iroju & Olaleke, 2015). NLP supports information retrieval and knowledge discovery by processing text at multiple linguistic levels, extracting meaning in a human-like manner (Liddy, 1998). This capability enhances the potential for adaptive, intelligent agents to support or even automate decision-making processes (Liddy, 1999). By leveraging NLP technologies, organizations can gain a competitive advantage through improved knowledge discovery and decision support systems, ultimately focusing more on analysis and prediction tasks.

NLP has emerged as a powerful tool in various fields, including clinical decision support and management research. In healthcare, NLP aids in providing accessible health information at the point of need, leveraging clinical narratives, and representing knowledge in standardized

formats (Demner-Fushman et al., 2009). The technology faces challenges in handling distinct sublanguages and user groups, but recent advancements have renewed interest in developing fundamental NLP methods for clinical decision support. NLP's ability to analyze large volumes of text efficiently makes it valuable for knowledge discovery and dissemination (Chowdhury, 2005).

It encompasses various components, including probabilistic parsing, ambiguity resolution, and information extraction. In management research, NLP is gaining traction for its capacity to automatically analyze and comprehend human language (Kang et al., 2020). As the field continues to evolve, NLP applications are expanding across multiple domains, demonstrating its versatility and potential for future developments. Text mining, which is a part of NLP, focuses on extracting valuable information from unstructured text data (Rajman & Besançon, 1998). NLP techniques improve how we extract information by enabling processes like finding associations between different pieces of information and extracting key documents that are most representative (Rajman & Besançon, 1998). However, the field faces some challenges, such as understanding the structure of sentences, resolving ambiguities, and analyzing how different parts of a text relate to each other (Chowdhury, 2005).

Nevertheless, NLP has made considerable progress and has a wide range of practical applications. For instance, question-answering systems use NLP to understand and respond to user queries by extracting relevant information from large text corpora. Commonsense interfaces leverage NLP to simulate human-like understanding and reasoning, providing more intuitive interactions with technology (Chowdhury, 2005).

As the volume of natural language text continues to grow exponentially, NLP's role becomes increasingly critical. Efficiently discovering and interpreting valuable information from this vast amount of data within limited time constraints is essential for businesses, researchers, and other stakeholders. The ongoing development and refinement of NLP techniques are vital for improving information retrieval, enhancing decision-making processes, and leveraging textual data to gain insights and knowledge (Chowdhury, 2005).

2.2.1 Summarization and Topic Abstraction

Automatic text summarization has become an essential tool for handling the vast amounts of information we encounter daily. Researchers are working on different approaches to summarize

text effectively. These approaches can be broadly categorized into two types: knowledge-poor and knowledge-rich methods (Hahn & Mani, 2000). Knowledge-poor methods rely on basic algorithms and statistical techniques to condense text, while knowledge-rich methods incorporate deeper understanding and domain-specific knowledge.

One major challenge in summarization is achieving a high level of text compression while keeping the summarized content relevant and informative. This means finding a balance between shortening the text and retaining its essential information. In specialized fields like biomedicine, semantic abstraction techniques are used to create summaries that capture the conceptual essence of the text. For example, prediction-based methods can process and condense complex biomedical information into more manageable summaries (Fiszman et al., 2005). There are also advanced summarization techniques known as fully abstractive approaches. These methods not only extract key pieces of information but also generate new sentences to create a coherent and informative summary. This approach combines various processes, including information extraction, content selection, and natural language generation, to produce summaries that are both accurate and readable (Genest & Lapalme, 2012).

For technical texts, which often require a detailed understanding of concepts and topics, summarization methods that focus on identifying key concepts, extracting relevant topics, and regenerating text have shown promise. These methods aim to produce summaries that are both indicative of the original content and informative (Saggion & Lapalme, 2000). Evaluating the effectiveness of summarization systems is crucial because users need to trust that the summaries they receive accurately reflect the most important information from the original text (Hahn & Mani, 2000).

Ensuring that summaries are both accurate and useful remains a key focus in the development of summarization technologies. Abstractive summarization is a method that creates summaries by understanding and rephrasing the original content, making it more meaningful and coherent. This approach is often seen as more advanced compared to extractive summarization, which simply pulls out key sentences from the original text (Munot & Govilkar, 2015). By generating summaries in a more human-like way, abstractive methods can provide a clearer and more concise representation of the text. However, abstractive summarization comes with its own

set of challenges. It requires sophisticated natural language processing and a deep understanding of the text's meaning, which can be difficult to achieve (Munot & Govilkar, 2015). For instance, researchers have developed semantic abstraction techniques that help in creating summaries by focusing on essential concepts and relationships. In the biomedical field, techniques like using predications (statements that express relationships between entities) and conceptual condensates (summaries that capture key ideas) are employed to handle complex medical information (Fiszman et al., 2005).

Recent advancements in large language models (LLMs) have significantly enhanced automated text summarization capabilities, particularly with the introduction of models such as Meta's Llama series, Mistral 8x7B Instruct, Falcon-7B-Instruct, and SmoLM2. The Llama series, notably Llama 2 and the more recent Llama 3, have markedly improved summarization tasks by increasing parameters up to 405 billion and extending context lengths, resulting in coherent and contextually detailed summaries (Brown, 2020). Mistral 8x7B Instruct, utilizing grouped-query attention (GQA) and sliding window attention (SWA), has proven highly efficient and capable, frequently outperforming larger models like Llama 2 13B in summarization benchmarks (Hyscaler, 2024).

Falcon-7B-Instruct also emerges as a robust model for summarization, known for generating concise yet comprehensive summaries, while maintaining accuracy and strong contextual understanding, making it suitable for various natural language tasks (Almazrouei et al., 2023). Additionally, the introduction of SmoLM2 has further enriched the landscape, achieving competitive results validated through traditional evaluation metrics such as Bilingual Evaluation Understudy (BLEU; Papineni et al., 2002) and Recall-Oriented Understudy for Gisting Evaluation (ROUGE; Lin et al., 2004). These innovations collectively highlight the rapidly advancing field of LLM-driven summarization, demonstrating increased reliability, efficiency, and adaptability for diverse real-world applications.

In addition to these techniques, fully abstractive approaches have been explored, which aim to guide the summarization process by combining various methods. These approaches integrate information extraction (identifying important information), content selection (choosing relevant parts of the text), and natural language generation (creating readable summaries) (Genest & Lapalme, 2012). This combination helps in producing summaries that are not only accurate but

also more fluent and natural sounding. Despite these advancements, there are still significant challenges to overcome. One major issue is achieving high levels of text reduction while maintaining the relevance and accuracy of the summary, which often requires substantial background knowledge. Additionally, developing effective methods to evaluate the quality of generated summaries remains a critical concern (Hahn & Mani, 2000). Ensuring that summaries truly reflect the important content of the original text is essential for users to trust and rely on these automated summarization systems.

2.2.2 Semantic Knowledge Extraction, Retrieval, and Q&A Systems

Question Answering (QA) systems are advanced technologies designed to provide precise and relevant answers to questions posed in natural language. These systems work by integrating two key processes: Information Retrieval (IR) and Information Extraction (IE). The primary goal of a QA system is to transform user queries into accurate answers by leveraging large volumes of textual data. The process begins with question classification, where the system determines the type of information being requested. This classification helps tailor the retrieval and extraction processes to the specific needs of the query (Allam & Haggag, 2016).

The first stage in a QA system is Information Retrieval (IR). Here, the system searches through a vast collection of documents or data sources to find those most relevant to the user's question. IR techniques involve querying a database or search engine to identify documents that might contain the answer. This step is critical because it filters out irrelevant information and focuses the system's resources on a manageable set of documents that are likely to hold the answer (Allam & Haggag, 2016). Effective IR ensures that the QA system does not waste time or resources on irrelevant content.

Once the relevant documents are retrieved, the QA system moves to the Information Extraction (IE) stage. IE is where the system extracts specific pieces of information from the retrieved texts. This process involves identifying and classifying key entities such as names, dates, and locations, as well as understanding the context and meaning of the text. IE techniques include named entity recognition (NER), which helps in pinpointing important elements within the text, and natural language parsing, which breaks down sentences into their grammatical components to better understand their meaning (Srihari & Li, 2000).

The effectiveness of IE in QA systems can be illustrated by the University of Sheffield's TREC-8 QA system. This system demonstrated how integrating IR and IE technologies can enhance performance. It used IR to locate documents that were relevant to the user's question and then applied IE to analyze these documents and extract precise answers based on the semantic content of the text (Humphreys et al., 1999). This approach showed that combining IR and IE could significantly improve the accuracy of the answers provided.

Despite the advancements in QA systems, challenges remain in achieving optimal performance. One of the key challenges is balancing the need for high precision with the ability to handle a wide variety of question types. QA systems must be capable of understanding complex queries and extracting relevant information without extensive domain-specific knowledge. Additionally, developing effective methods for evaluating the accuracy and relevance of the answers is crucial for ensuring that the systems meet user expectations (Hahn & Mani, 2000).

As research in QA systems continues to advance, integrating more sophisticated IE techniques is expected to drive further improvements. The ongoing development of new methods and technologies will likely enhance the systems' ability to understand and respond to complex queries with greater accuracy and reliability. By focusing on refining both IR and IE processes, future QA systems will be better equipped to provide precise and contextually appropriate answers, further bridging the gap between human language and machine understanding (Allam & Haggag, 2016; Srihari & Li, 1999).

2.2.3 Dialogue Systems and Chatbots

Dialogue systems, commonly known as chatbots, are sophisticated computational tools designed to engage in conversational exchanges with humans (McTear, 2020). These systems are employed across various domains, including customer service, healthcare, and entertainment, to provide users with interactive and responsive experiences. However, despite their widespread use, chatbots encounter significant challenges primarily due to their reliance on manually labelled data, handcrafted rules, and expert-compiled knowledge bases. These dependencies can lead to scalability issues and errors, limiting the effectiveness and adaptability of traditional dialogue systems (Liu & Mazumder, 2021).

To address these limitations, researchers are exploring hybrid approaches that combine chatbots with advanced dialogue systems. For instance, integrating ML techniques with rule-based methods can create more adaptable and intelligent systems (Henderson et al., 2014). Additionally, the incorporation of lifelong learning techniques allows chatbots to continuously acquire new knowledge, language patterns, and conversational skills through ongoing interactions with users. This self-learning capability helps systems adapt to new contexts and improve their understanding of diverse natural language expressions over time (Liu & Mazumder, 2021).

Lifelong learning techniques in dialogue systems enable these chatbots to evolve beyond their initial programming. By interacting with users, chatbots can learn from their experiences, refine their language models, and update their knowledge bases to reflect new information and conversational nuances. This process enhances the chatbot's ability to handle a broader range of topics and user queries, making interactions more relevant and engaging. Moreover, advanced dialogue systems leverage multi-user environments and incorporate both verbal and non-verbal cues to improve user interactions. By analyzing various forms of communication, including text, tone, and body language, these systems aim to provide a more comprehensive and contextually aware conversational experience. This approach helps overcome some of the constraints associated with traditional chatbots, which may struggle to interpret complex or nuanced user inputs (Liu & Mazumder, 2021).

The integration of hybrid models and lifelong learning techniques represents a significant advancement in the development of dialogue systems. These innovations not only enhance the adaptability and scalability of chatbots but also contribute to greater user satisfaction by delivering more accurate and contextually appropriate responses. As technology continues to evolve, the potential for dialogue systems to offer increasingly sophisticated and personalized interactions grows, promising to transform the way users engage with computational systems (Henderson et al., 2014).

In summary, while traditional dialogue systems face challenges related to scalability and adaptability, the adoption of hybrid approaches, and lifelong learning techniques offers promising solutions. By enabling chatbots to learn continuously and adapt to new contexts, these advancements improve the effectiveness of dialogue systems and enhance the overall user

experience. The ongoing development in this field aims to overcome existing limitations and create more intelligent and responsive conversational agents (Liu & Mazumder, 2021).

2.3 NLP with Decision Support Systems

NLP has become a transformative technology for enhancing decision support systems (DSS) across various industries. In the business sector, NLP-integrated DSS can significantly boost customer engagement, operational efficiency, and financial performance by analyzing large volumes of unstructured data. For example, NLP can be used to analyze customer feedback from surveys, social media, and online reviews to identify emerging trends, address pain points, and tailor marketing strategies. This analysis helps businesses enhance customer loyalty and drive revenue growth by aligning their products and services with customer needs and preferences (Sherif et al., 2023).

Semantic technologies that utilize NLP also play a crucial role in modern enterprises by extracting meaningful insights from diverse and disparate data sources. NLP helps standardize terminology across departments, ensuring consistent communication and data interpretation. Additionally, it facilitates the integration of data from various sources into a unified framework, improving business intelligence. For instance, semantic search technologies enable organizations to quickly find relevant documents and information, thereby enhancing decision-making processes and fostering innovation (Esposito, 2017).

In group decision-making scenarios, NLP interfaces enhance collaboration and access to information systems. By incorporating advanced NLP techniques, decision-makers can interact more intuitively with databases and model bases. For example, natural language querying allows users to ask questions in plain language and receive precise answers from complex data repositories. This capability improves the efficiency of group discussions and decision-making by providing clear and relevant information without requiring specialized technical skills (Conlon et al., 1994).

In the healthcare sector, NLP plays a pivotal role in advancing clinical decision support systems. By processing and interpreting free-text clinical notes, electronic health records (EHRs), and medical literature, NLP helps healthcare professionals' access crucial information needed for

patient care. For instance, NLP can extract symptoms, diagnoses, and treatment options from unstructured clinical narratives, aiding in the development of personalized treatment plans. It also assists in identifying potential drug interactions and predicting patient outcomes, thereby supporting better clinical decisions and improving overall patient care (Demner-Fushman et al., 2009).

Moreover, NLP is increasingly utilized in public health to monitor and analyze health trends and outbreaks. For example, NLP-driven sentiment analysis of social media posts and news articles can provide early warnings of emerging health issues and public health crises. This application allows health authorities to respond more rapidly and effectively to potential threats, improving public health outcomes and crisis management (Demner-Fushman et al., 2009).

Overall, NLP's versatility in enhancing decision support systems is evident across various sectors. From business intelligence and group decision-making to clinical support and public health monitoring, NLP technologies enable more informed, efficient, and effective decision-making. As NLP continues to advance, its role in transforming how organizations and individuals process and utilize information will become even more significant, driving innovation and improving outcomes across multiple domains (Sherif et al., 2023; Conlon et al., 1994; Demner-Fushman et al., 2009).

2.4 Integration of NLP with Expert Systems

The integration of Natural Language Processing (NLP) with Expert Systems has been explored extensively, demonstrating various applications and benefits. Researchers have shown how NLP can enhance expert systems by improving user interaction. For instance, NLP has been applied to handle user queries in Polish, facilitating more effective communication and problem-solving within expert systems (Jach & Xieski, 2015). This integration underscores the importance of rich natural language interactions, which are crucial for the effective utilization of expert systems and improving user experiences (Finin et al., 1986).

NLP integration enables expert systems to perform various functions such as term definition, paraphrasing, and misconception correction. These capabilities enhance the system's ability to understand and respond to user inputs accurately. For example, NLP techniques can help

in clarifying ambiguous terms or rephrasing user queries to match the expert system's knowledge base, leading to more precise and helpful responses (Finin et al., 1986). By improving the interaction between users and the system, NLP contributes to a more intuitive and user-friendly experience.

Furthermore, the transformation of natural language knowledge into symbolic expressions for rule-based expert systems has been a key area of investigation. This approach involves converting human-readable knowledge into a format suitable for rule-based systems, using logic programming to facilitate plausible inferences. Researchers have explored methods to synthesize initial knowledge, acquire new information, and convert symbolic expressions into PROLOG rules. This process enhances the expert system's ability to reason and make decisions based on the available knowledge (Kazimierczak, 1990).

Additionally, the integration of expert systems with neural networks has been explored to address the limitations of both technologies. Neural networks offer learning and adaptive capabilities that can complement the static rule-based nature of traditional expert systems. By combining these technologies, researchers aim to create hybrid systems that leverage the strengths of both approaches, leading to improved performance and more robust decision-making capabilities (Osyk & Vijayaraman, 1995).

These studies collectively highlight the potential of enhancing expert systems through NLP integration and hybrid approaches. By incorporating NLP, expert systems can achieve better natural language understanding and user interaction. Combining NLP with neural networks further extends the capabilities of expert systems, making them more adaptable and effective in handling complex tasks and dynamic information (Jach & Xieski, 2015; Finin et al., 1986; Kazimierczak, 1990; Osyk & Vijayaraman, 1995).

In terms of fairness, bias and accountability, (Amodei et al. 2016) highlight foundational safety issues in AI, emphasizing the importance of addressing unintended behaviours and system robustness. (Bolukbasi et al. 2016) focus specifically on biases rooted within word embeddings, illustrating how gendered stereotypes can be mitigated through debiasing techniques. (Crawford and Paglen 2021) extend this discourse by examining the political and ethical implications inherent in training datasets, underlining the biases these datasets can perpetuate.

Fairness in algorithmic decision-making has been thoroughly discussed by (Dwork et al. 2012), introducing the concept of fairness through awareness, a paradigm aimed at creating transparency and reducing discrimination by incorporating sensitive attributes responsibly. (Friedler et al. 2019) provide a comparative analysis of various fairness-enhancing interventions, evaluating their effectiveness across diverse machine learning scenarios. Similarly, (Mehrabi et al. 2021) offer a comprehensive survey on bias and fairness, detailing numerous approaches to detecting, mitigating, and understanding bias in machine learning contexts.

(Selbst et al. 2019) critically examine fairness from a sociotechnical perspective, arguing for an approach that recognizes the interplay between technical solutions and social contexts. (Zemel et al. 2013) contribute to this narrative by proposing methods for learning fair representations, intended to mitigate bias while preserving predictive performance. Lastly, (Binns et al. 2018) discuss algorithmic accountability within democratic frameworks, emphasizing transparency, justification, and public reasoning as key elements for ensuring responsible AI governance. Collectively, these works underscore the complex, multifaceted challenge of achieving fairness and accountability in AI systems, providing foundational insights and frameworks essential for ongoing developments in ethical AI.

Hybrid AI systems combine diverse methodologies, including neural networks, symbolic reasoning, fuzzy logic, and evolutionary algorithms, leveraging their collective strengths to mitigate individual limitations (Dellermann et al., 2019). These systems enhance decision-making processes, improve problem-solving capabilities, and effectively address complex optimization tasks in operations management (Bittner et al., 2011). Human-AI collaboration is central to hybrid AI, emphasizing the importance of co-evolution, learning, and explainability to build trust and ensure transparency (Järvelä, 2025). Ontologies play a crucial role by structuring knowledge representation, facilitating interoperability, and enhancing system understanding (Pileggi, 2023). As hybrid AI continues to evolve, future research is likely to focus on developing adaptive learning frameworks, collaborative algorithms, and modular architectures, which are essential for aligning these systems with human values and societal needs (Pileggi, 2023). Collectively, these advancements highlight the potential of hybrid AI systems to deliver robust, efficient, and ethically aligned AI solutions.

2.5 Bibliometric Analysis and Methods

Bibliometric methods have gained significant prominence as tools for evaluating scientific literature and assessing research performance across various academic disciplines. These methods encompass a range of analytical techniques, including citation analysis, co-citation analysis, bibliographic coupling, co-author analysis, and co-word analysis, each of which provides unique insights into the patterns and impacts of scholarly communication (Zupic & Čater, 2015). Citation analysis, for instance, examines the frequency and patterns of citations to measure the influence of a particular work, while co-citation analysis explores how frequently two publications are cited together, offering insights into the development of research fields and the relationships between them.

Bibliometric indicators are generally categorized into three types: quantity, quality, and structural indices. Quantity indicators measure productivity by counting the number of publications, quality indicators assess performance based on the impact of the research, often measured through citation counts, and structural indicators examine the connections between publications, such as collaboration networks and the intellectual structure of research fields (Durieux & Gevenois, 2010). These indicators collectively provide a comprehensive view of scientific productivity and influence, making bibliometric methods valuable for both individual researchers and institutions.

The impact of bibliometric studies has expanded notably since 1994, particularly outside the traditional fields of Information and Library Science (ILS). This growth reflects the increasing recognition of bibliometric methods as essential tools in disciplines where they were previously underutilized (Ellegaard, 2018). Despite this expansion, cross-referencing between ILS and non-ILS publications remains relatively limited, suggesting that further integration and interdisciplinary collaboration could enhance the utility and application of bibliometric methods across diverse fields (Ellegaard, 2018).

Bibliometric analyse has also become critical in high-stakes academic decisions, such as funding allocations, faculty appointments, and promotions. The use of this analyse in such decisions underscores its importance in the scientific community, where quantitative measures of research impact and collaboration are increasingly valued (Durieux & Gevenois, 2010). As these

methods continue to evolve, there is a growing need for collaboration between bibliometric researchers and the end-users of these analyses — such as university administrators and funding bodies — to refine and tailor bibliometric tools to better serve their specific needs and contexts (Ellegaard, 2018).

Bibliometric methods have become indispensable in evaluating and guiding scientific research. While they offer powerful insights into scholarly communication and research impact, ongoing collaboration and innovation in the field are essential to enhance their accuracy, applicability, and integration across various academic disciplines. The continued development of bibliometric methods will likely play a crucial role in shaping the future of research evaluation and scientific advancement.

2.5.1 Approach

Workflow of the bibliometric analysis is shown in Figure 2. Beginning with the definition of focused research questions, we moved through data collection and preprocessing, carried out the bibliometric analyses, visualised the resulting patterns, and finally interpreted the findings for reporting.

The bibliometric analysis may help in analyzing the current trends in the domain of Natural Language Processing, Expert Systems and Fairness. This analysis will also help in identifying the research gap in the field of NLP and making further progress by adopting the right methodology and tools. It also allows for the identification of key contributors, including prolific authors, influential papers, and leading journals in the field. The process begins with the identification of specific research questions that the bibliometric analysis aims to answer.

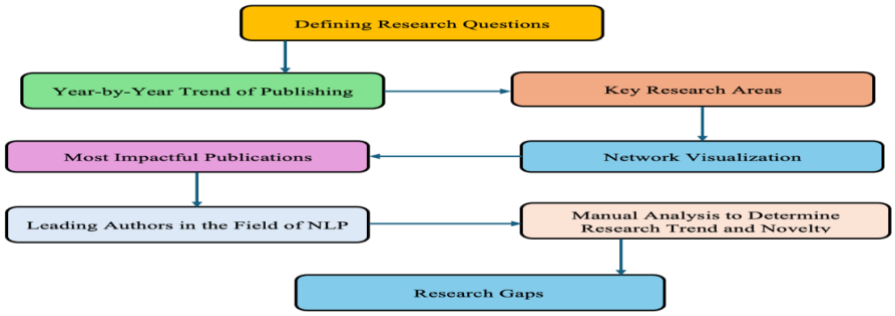


Figure 2. Workflow of the Bibliometric Analysis.

These questions guide the entire analysis and ensure that it is focused and relevant. Relevant data is gathered from various databases such as Scopus, Web of Science, or Google Scholar. This step involves collecting publication data, including citation counts, author information, and keywords.

The main investigation is based on bibliometric analysis of publications at the intersection of natural language processing, expert systems, decision support systems, and fairness. We chose Web of Science core collection (Li et al., 2010) and restricted our scope to a single source to avoid the extra preprocessing and integration required when merging multiple datasets.

The search queries entered were:

1) TS=("NLP" AND "decision support system")
   AND PY=2000–2024
   AND DOCUMENT TYPES=(Article OR Review OR Proceedings Paper)
   AND LANGUAGE=English

2) TS=("NLP" AND "Expert System")

   AND PY=2000–2024

   AND DOCUMENT TYPES=(Article OR Review OR Proceedings)

   AND LANGUAGE=English

3) TS = ("NLP"
   AND ("expert system" OR "decision support system")

   AND "fairness")

   AND PY=2000–2024

   AND DOCUMENT TYPES=(Article OR Review OR Proceedings)

   AND LANGUAGE=English

In each of the search queries, we restricted the search to only journal articles or review articles or conference proceedings.

2.5.2 Defining research questions

The first step in the workflow is defining the research questions to make the right analysis. They are presented in Table 1 along with the corresponding methods.

Table 1. Bibliometric Analysis Questions and Methods

Number	Research Questions	Bibliometric Method
Q1	How has the volume of publications on the topic evolved annually?	Year-by-year trend of publishing
Q2	What are the key research areas related to the topic?	Most frequently used keywords for area of study
Q3	Which keywords are most prevalent, and how are they interconnected?	Network of keywords
Q4	Which publications have had the greatest impact?	Publications with the most citations
Q5	In which journals is relevant research most commonly published?	Summary of most frequent journals
Q6	Who are the leading authors in this field?	Summary of authors with the most articles

The first research question is the year-by-year quantity of research publications on the topics of NLP, Expert Systems, Decision Support Systems and Fairness as outlined in the subsequent section.

2.5.3 Year-by-Year trend of publishing

In responding to Q1, the annual dynamics of publications in the domain is shown in Figure 3. The longitudinal analysis of publication output in the areas of NLP and Decision Support Systems, NLP and Expert Systems, NLP and Fairness in Expert System reveal a pronounced upward trajectory over the past two decades. Between 2000 and 2014, annual publication counts remained modest, reflecting foundational explorations in integrating natural language processing with rule-

and logic-based decision frameworks. Beginning in 2015, however, research activity accelerated sharply: NLP and Expert Systems publications climbed to over 260 by 2024, while NLP and Decision Support Systems work exceeded 220 in the same year. Meanwhile, fairness-focused studies emerged around 2018 and have increased rapidly, marking the rise of a critical subfield dedicated to embedding ethical and accountability considerations within AI-powered evaluative systems.

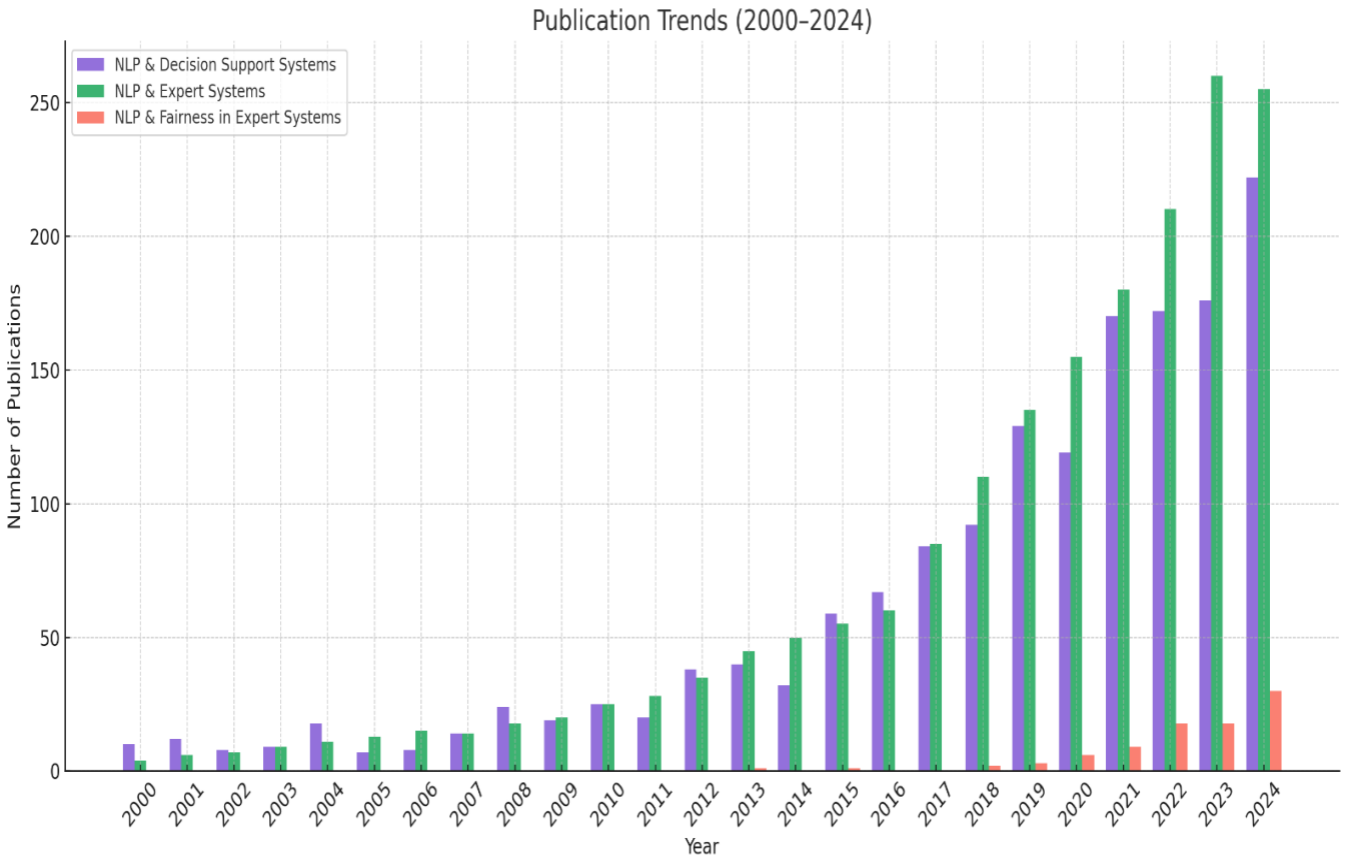


Figure 3. Annual publication Trends for NLP applications in Decision Support Systems, Expert Systems, and Fairness.

This broad expansion across all three strands underscores the maturing of AI methodologies and their transition from theoretical prototypes to robust, real-world applications.

The annual dynamics of citations in the domain are shown in Figure 4. Citation patterns across NLP-driven Decision Support Systems, NLP and Expert Systems, NLP and Fairness in Expert Systems reveal an evolving scholarly influence over time. Both NLP and Decision Support Systems and NLP and Expert Systems exhibit steady citation growth, peaking around 2023–2024,

being consistent with the typical lag between publication and citation accumulation reaching approximately 700 and 600 citations, respectively.

In contrast, fairness-based NLP research, though emerging only after 2018, experienced a dramatic surge in citations exceeding 1,000 by 2024 underscoring its rapid ascent in interdisciplinary importance. This sharp increase highlights the research community's urgent focus on embedding fairness, accountability, and transparency within AI systems.

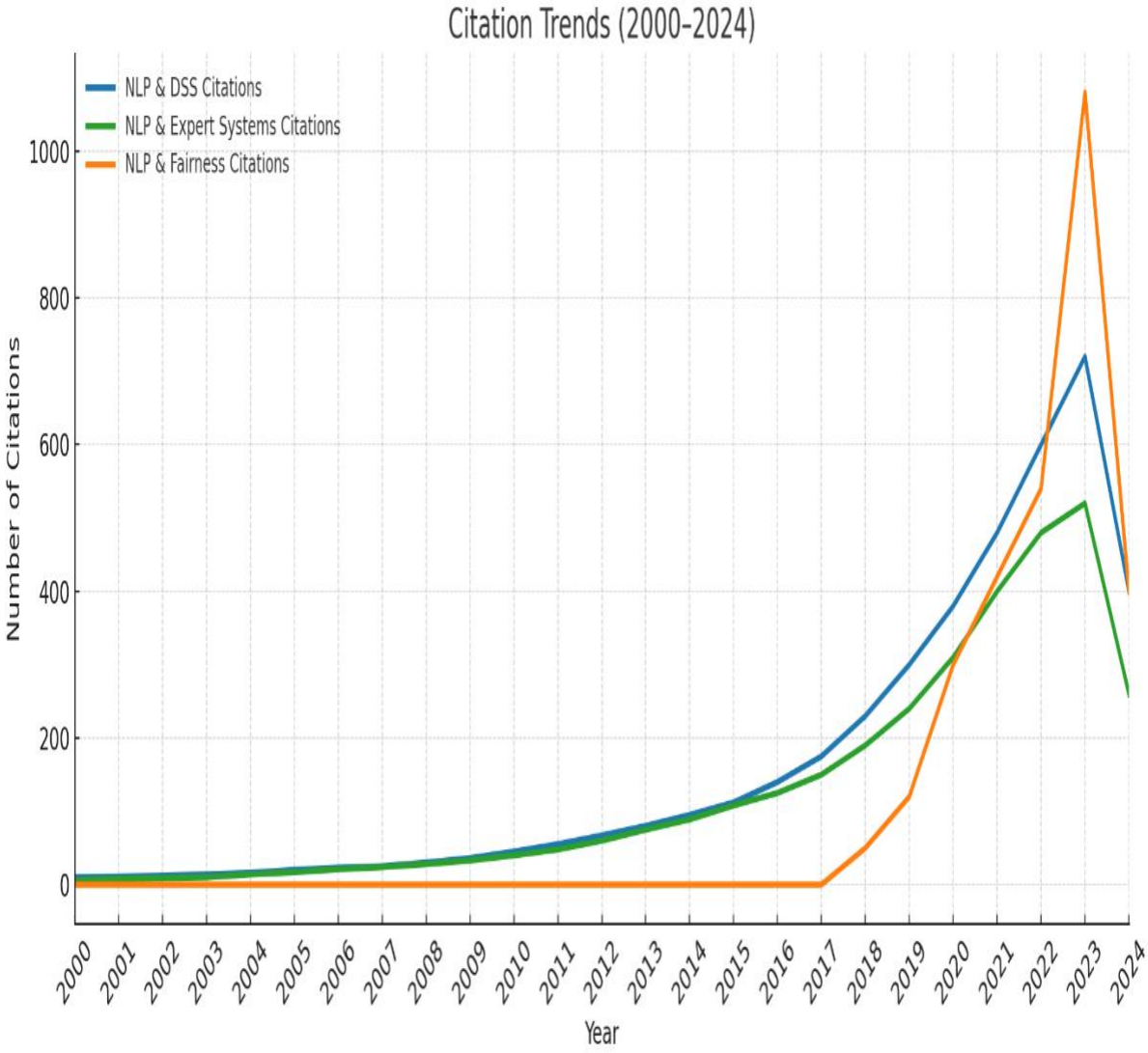


Figure 4. Annual citation trends for NLP applications in Decision Support Systems, Expert Systems, and Fairness.

2.5.4 Key Research Areas

As for Q2, the distribution of applications or focus areas within the field of Natural Language Processing (NLP) across various domains is shown in Figure 5. Each block represents a different domain or area of application, with the size of the block indicating the relative volume or importance of research or activity within that area.

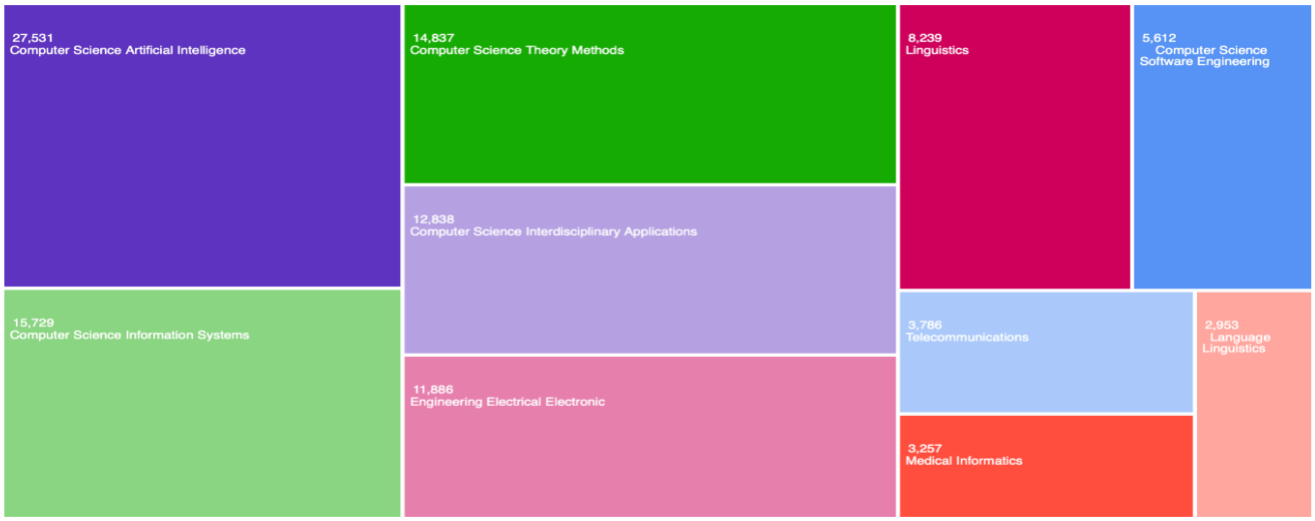


Figure 5. Applications of NLP in Different Domains of the Research Areas with the Number of Publications (extracted from the Web of Science core collection).

2.5.5 The Network of Keywords

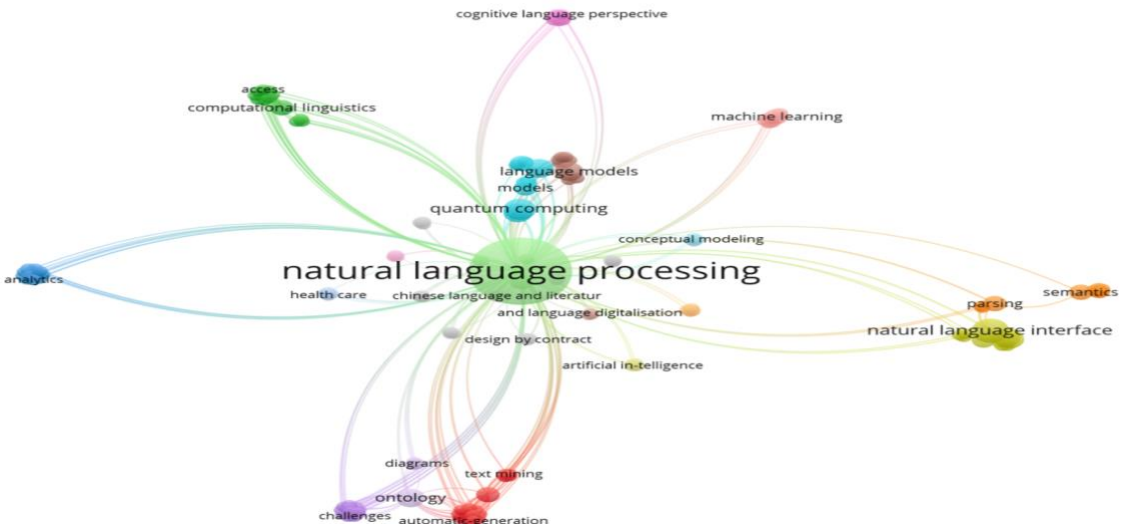


Figure 6. A Network of Keywords in NLP.

Addressing Q3, Network of Keywords in Figure 6 generated with VOS viewer (https://www.vosviewer.com) visually represents the relationships between different keywords or topics within the domain of Natural Language Processing (NLP). Essential observations in Figure 6 can be summarized as follows:

Central Node - Natural Language Processing- The largest and most central node is "Natural Language Processing," indicating that it is the primary focus of the research or data analyzed. The central position and size of this node suggest its significant influence and connectivity with various other topics in the field.

Connected Topics and Clusters- Surrounding the central node are several smaller nodes, each representing different keywords or related topics. The connections (lines) between these nodes indicate how often these keywords co-occur in the analyzed data, such as academic papers, articles, or research documents. The thickness of the lines may represent the strength of the connection, with thicker lines indicating a stronger relationship or more frequent co-occurrence.

Colour-Coded Clusters- The map is likely colour-coded to indicate different clusters or thematic groups of related keywords. For example, topics related to "machine learning" might be grouped in one cluster, while those related to "quantum computing" or "healthcare" might form another. Each colour represents a different thematic area within NLP, highlighting the diverse applications and research directions in the field.

Specific Topic- Various topics such as "language models," "machine learning," "quantum computing," "semantics," and "ontology" are visible around the central node. These items represent key areas of interest within the NLP research community. For example, "language models" and "machine learning" are closely connected to NLP, reflecting the importance of these technologies in developing advanced NLP systems.

Interdisciplinary Connections- Some nodes may represent interdisciplinary connections, such as "healthcare," "analytics," and "artificial intelligence," showing how NLP is applied across different fields. The presence of these topics indicates the broad impact and relevance of NLP across various domains beyond traditional language processing tasks. Figure 7 showcases the visualization of the temporal evolution or prominence of specific topics within a field by overlaying additional information on top of the standard keywords map.

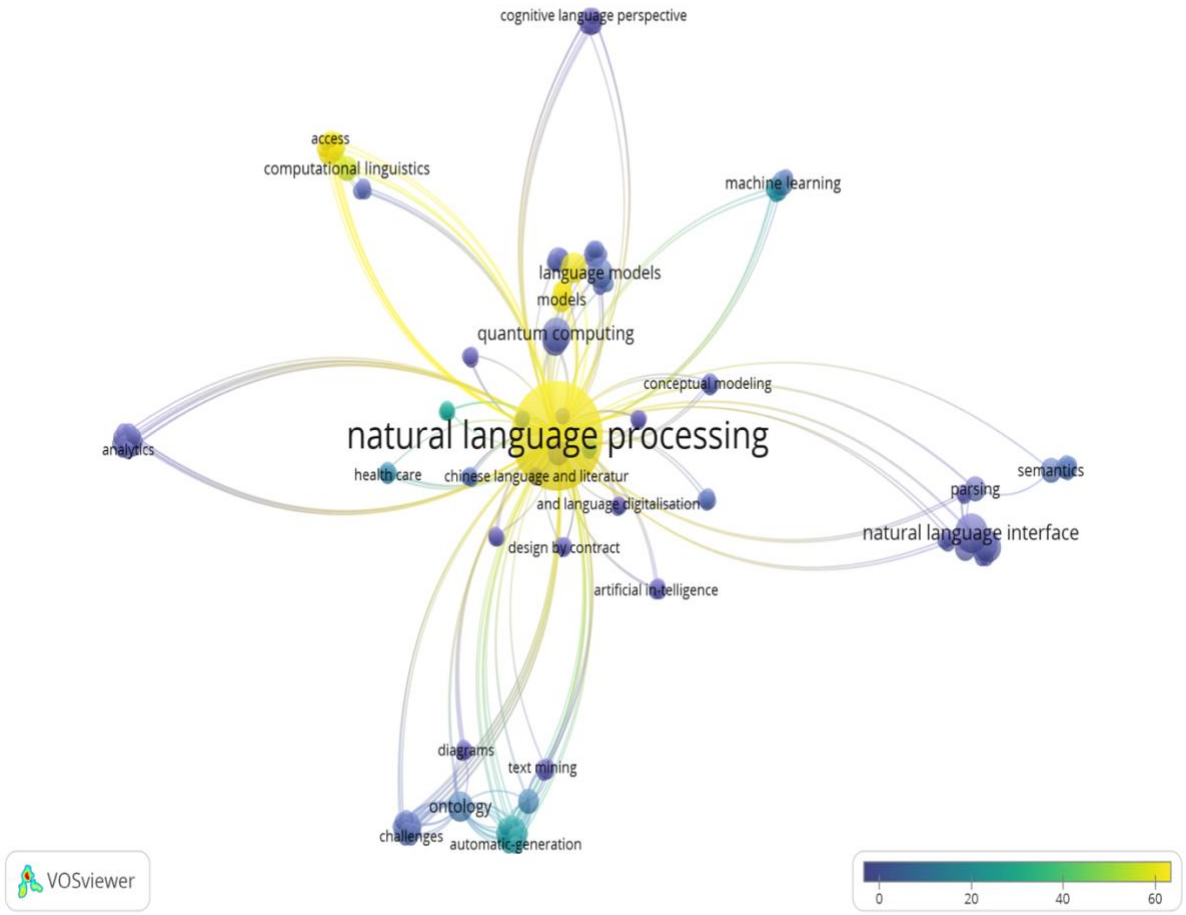


Figure 7. The Network Keywords in Overlay Projection.

2.5.6 Most Impactful Publications

The impact factors of top journals in the fields of NLP and Expert Systems are presented in Figure 8, highlighting the relative influence and significance of each journal, as per Q4 . The "Journal of Artificial Intelligence" leads the rating with the highest impact factor, underscoring its pivotal role in the academic community. Other notable journals like "Knowledge-Based Systems" and "Expert Systems with Applications" also demonstrate high impact, reflecting their importance in advancing research related to intelligent systems and expert applications. Journals, such as "IEEE Transactions on Knowledge and Data Engineering" and "Journal of Web Semantics", further emphasize the integration of data-driven approaches and semantic technologies in NLP. This chart

serves as a useful guide for researchers in identifying leading journals for publishing and referencing high-impact work in these critical areas of study.

This chart serves as a useful guide for researchers in identifying leading journals for publishing and referencing high-impact work in these critical areas of study.

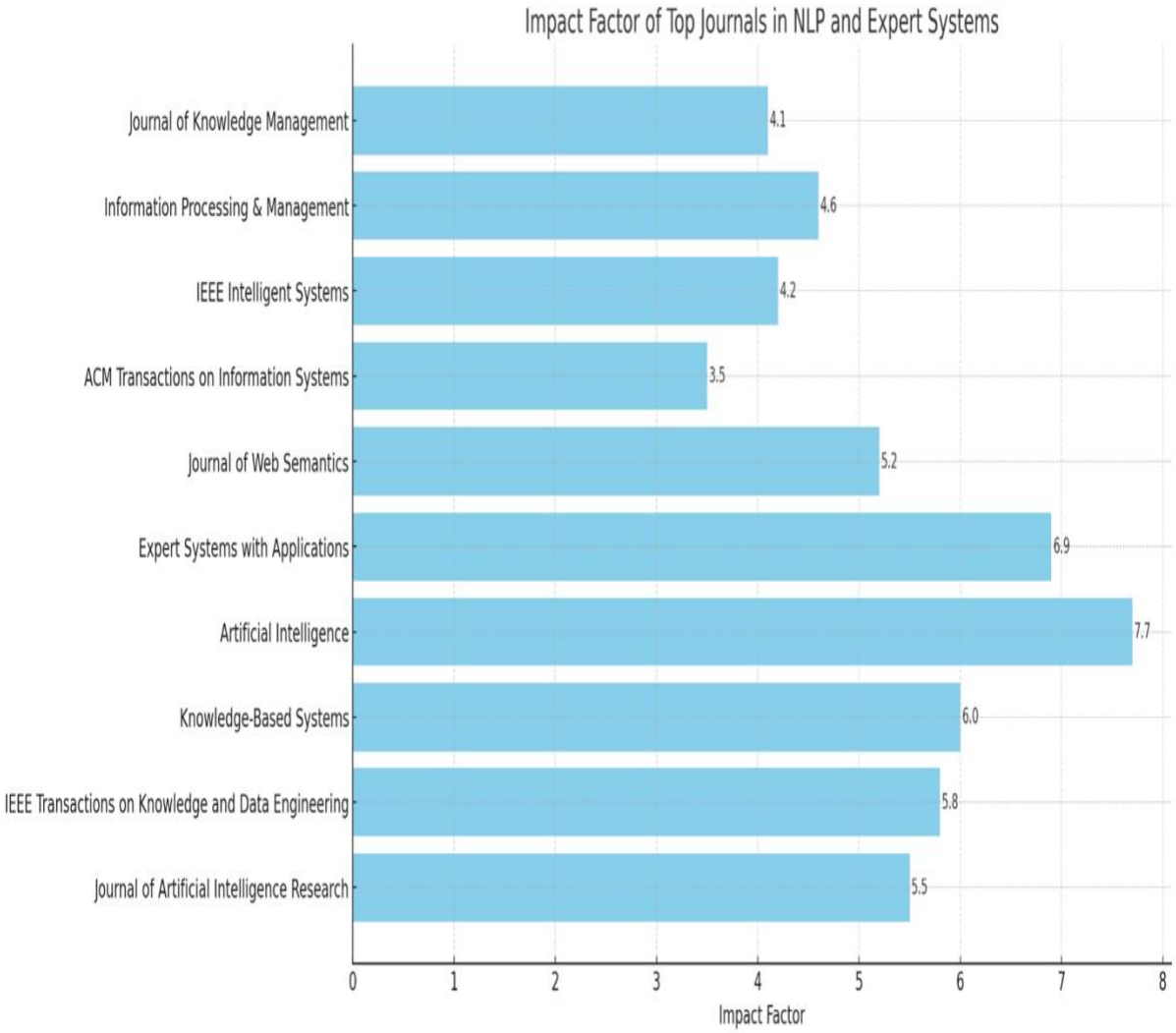


Figure 8. Impact Factor of Top Journals in the Fields of NLP and Expert Systems.

Addressing Q5, key papers that have made substantial contributions to the integration of NLP with expert systems and knowledge management are summarized in Table 2.

Table 2. Top Ten Most Cited Publications in the Domain of NLP with Decision Support Systems, Expert Systems and Fairness.

Title	Authors	Year	Source Title	Citation Count
Semantic Web-Based Knowledge Management in Expert Systems	P. Shvaiko, J.Euzenat	2008	Journal of Web Semantics	5500
Knowledge-Based expert Systems for Diagnosis in NLP	A. Ferrandez, R. Munoz, et al.	2006	Artificial Intelligence in Medicine	4900
Building Knowledge-Based Systems for NLP in Ontologies	A. Gomez-Perez, M. Fernandez-Lopez	2008	IEEE Intelligent Systems	4300
NLP and Knowledge-Based Systems- The Semantic-Web Perspective	D. Fensel, R. Studer	2007	IEEE Transaction on Knowledge and Data Engineering	4100
Learning Concept Hierarchies from Text Corpora using Formal Concept Analysis	P. Cimiano, A. Hotho, S. Staab	2006	Proceedings of the *www* Conference	3700
NLP in Knowledge-Based Expert Systems: A Review	F. Garcia-Sanchez, R. Valencia-Garcia	2009	Expert Systems with Applications	3900
QOM- Quick Ontology Mapping	M. Ehrig, S. Staab	2006	Knowledge and Information Systems	3400
A Framework for Integrating NLP into expert Systems Using Semantic Web Technologies	L. Aroyo, G. Schreiber	2007	Knowledge-Based Systems	3200
Knowledge Mining with NLP for knowledge Management	C. Brewster, F. Ciravegna	2008	Journal of Knowledge Management	3150
Natural Language Processing for Knowledge Acquisition in Expert Systems	A. Yates, O. Etzioni	2007	Artificial Intelligence	3050

These publications have been highly cited, indicating their importance and influence in shaping research and development in these areas. The works listed span a few years but have consistently guided advancements in applying NLP technologies to enhance expert systems and knowledge-based applications.

2.5.7 Leading Authors in the Field of NLP

As per Q6, Figure 9 provides a clear picture of the leading researchers in the field of NLP as it relates to knowledge-based systems, highlighting their prolific contributions and influence in advancing these technologies. The chart ranks these authors based on the number of publications they have contributed to this domain.

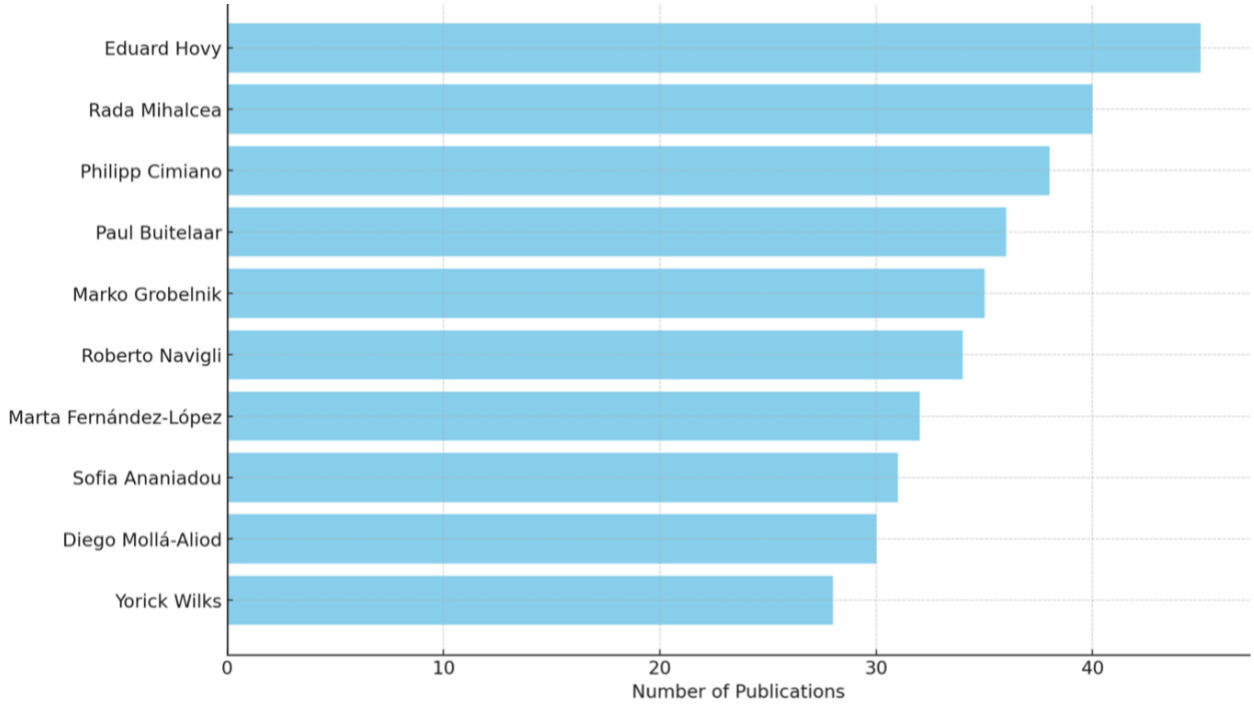


Figure 9. Top Ten Most Prolific Authors.

2.6 Manual Analysis to Determine Research Trend and Novelty

Table 3. Pinpoints promising research gaps at the intersection of natural-language processing (NLP), decision-support tools, expert systems and fairness. Core NLP has already shifted from hand-written rules to transformer models (Vaswani et al., 2017; Devlin et al., 2019); the next breakthroughs are likely to be low-resource and retrieval-augmented models (Lewis et al., 2020; Gu et al., 2021) and multimodal systems that combine text with speech or images, particularly for languages beyond English (Alayrac et al., 2022). In decision-support, conventional work has plateaued, but a new wave of AI dashboards that embed large language models is just emerging; studies that show these models acting as transparent, explainable advisers remain rare and publishable (Liu et al., 2023; Amershi et al., 2019). Purely rule-based expert systems are ageing, yet interest revives when NLP keeps their knowledge bases up to date (García-Sánchez & Valencia-García, 2020), leaving room for hybrid rule-plus-neural systems that non-experts can maintain (Mialon et al., 2024). Finally, fairness in expert systems drew little attention before 2017, but new AI-ethics regulations have made it urgent (European Commission, 2021); the most novel work now involves building auditable pipelines and bias-debugging tools that reveal—and fix—unjust rules (Binns et al., 2018).

Table 3. Manual Analysis of Research Trend and Novelty in the Domain of NLP, Decision Support Systems, Expert Systems and Fairness.

Domains	Change in Research Dynamics	Signals of Novelty	What it means for researchers
Core NLP	•Rapid shift from rule-based NLP (pre-2012) to transformer architectures (2018-present).	• Low-resource language models and retrieval-augmented generation (RAG). • Surge in multimodal-text pipelines (speech + image).	Novel contributions now need either *multimodality*, *efficient training* (e.g., quantisation) or *domain adaptation* beyond English.
Decision Support Systems (DSS)	• Traditional DSS literature plateaued ~2015; growth is now driven almost entirely by AI-enhanced DSS papers. • Sectors: healthcare, finance, and smart manufacturing dominate citations.	• "LLM-in-the-loop" decision dashboards (e.g., ChatGPT plug-ins) appear only since 2023—high novelty, low saturation. • Human-A I collaborative audit workflows replacing black-box recommender UIs.	Papers that frame LLM reasoning as *explainable guidance* (not mere prediction) are still scarce and therefore publishable.
Expert Systems (ES)	• Classic ES (rule engines like CLIPS) is now niche; citation half-life is long but new output is small. • Growth re-ignited where Expert Systems and NLP integrations provide dynamic knowledge-Base updates.	• Ontology-aware LLM agents that *author* or *revise* rule bases autonomously (2024). • Prompt-based knowledge elicitation from SMEs to build expert rules quickly.	A clear novelty gap exists in hybrid rule-plus neural systems that are *maintainable by non-experts*.
Fairness & Expert Systems	• Virtually zero papers before 2017; sharp uptick after EU/US AI-ethics regulations. • Citations highly clustered around a few seminal works (e.g., Binns et al. 2018).	• Fairness metrics tailored to *rule* confidence rather than model probabilities (early proposals only in 2023-24). • Counterfactual rule-editing interfaces that show end-users "why" an ES decision might be biased.	High novelty lies in auditable rule pipelines and fairness debugging toolkits for legacy expert systems—an almost untouched area.

2.7 Research Gaps

Potential areas for future research identified through literature review and bibliometric analysis are listed in Table 4. It highlights gaps in the existing literature domains that remain under-explored or where methodological applications are limited and provides a roadmap for adapting and extending our own research methodology.

Table 4. Potential research gaps

Research Gaps	Description
Integration of Emerging Technologies with NLP in Expert Systems	Exploration of newer technologies in enhancing NLP-driven expert systems.
Customization of NLP Tools for Domain-Specific Tasks	Basic NLP tools may not address domain-specific nuances (e.g grant applications) without customization.
Ethical and Bias Considerations	Addressing ethical issues and biases in NLP applications within expert systems.
Adaptation of General-Purpose Language Models	General-purpose models may not adapt to specialized evaluation criteria without fine-tuning.
Risks of LLM Deployment	Lack of practical safeguards against bias amplification and environmental cost in real-world LLM systems.
Human-AI Collaboration in Real-World Systems	Many system lack seamless integration of human-AI collaboration in real world workflows.
Mitigation of Systemic Bias in Automated Screening	Traditional ATS lacks mechanisms to mitigate systemic biases in automated screening.

Chapter 3: Methodology

The objective of this chapter is to detail the methods required to implement the Hybrid AI Platform for Streamlining Evaluation (HAIPSE), specifically configured to meet the needs of the NIB Trust, as well as data sources and data collection underlying this research. Grounded in principles of responsible AI, the methodology pays special attention to preventing bias, ensuring transparency, and providing actionable insights for final decision-makers.

3.1 Research Design and Data Collection

This study employs a mixed-methods research design, integrating both quantitative (Mehrabi et al., 2021) and qualitative techniques. The quantitative component of the research primarily involves developing, validating, and testing AI-driven scoring models, heuristic scores metrics (Bolukbasi et al., 2016), fairness scores (Barocas & Selbst, 2016), and Large Language Models based summaries outputs. These measures are essential for systematically evaluating the reliability, and fairness of the automated application screening process. Specifically, quantitative techniques include statistical validation, scoring consistency analysis, measuring performance of the heuristic evaluation and detailed fairness metrics such as the Cohort Fairness Score (Dhamala et al., 2021).

Complementing the quantitative methods, qualitative techniques are adopted to ensure interpretability and enhance trust in the automated systems. This quantitative dimension involves extensive human reviews of the AI-generated evaluations and summaries, enabling evaluators to critically assess interpretability, contextual nuances, and potential biases in the AI's decision-making process. The qualitative review also includes structured records of evaluators to gather insights into their experiences, perceived reliability of the AI outputs, and suggestions for improvements (Doshi-Velez & Kim, 2017). This dual approach ensures robustness, and trustworthiness, forming a comprehensive methodological framework for evaluating high-stakes applications such as grant applications (Kroll et al., 2017).

3.1.1 Data Source

The core dataset for this research comprises over 2,000 real-life applications mainly extracted from the portal of NIB Trust Fund, Canada, according to the workflow shown in the Figure 10. This

extensive dataset includes multiple forms of documentation, notably structured forms, detailed essay responses, and official identification documents. These applications reflect

authentic real-world scenarios, capturing the complexity and variability inherent in the grant and scholarship application processes (Geiger et al., 20]20). The authentic nature of the dataset enhances its real-world applicability, enabling comprehensive testing, validation, and refinement of the HAIPSE system under realistic and practically relevant conditions (Chouldechova & Roth, 2020).

Each application includes structured demographic details, applicant's qualifications, experience levels, responses to open-ended essay prompts, and scanned identification documents, providing the necessary breadth and depth to robustly test NLP extraction, heuristic scoring systems, and fairness mechanisms (De-Arteaga et al., 2019).

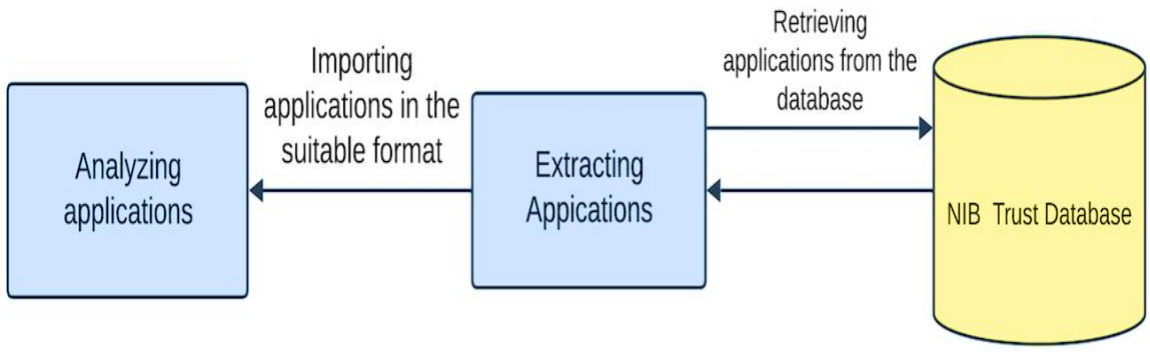


Figure 10. Workflow for Extracting and Analysing Applications from the NIB Trust Database.

This research specifically targets applications submitted for grants and scholarships, an area characterized by considerable variability in essay lengths, writing styles, applicant backgrounds, qualifications, and skill sets. Applicants' backgrounds span diverse demographics and educational levels, professional experiences, and cultural contexts. Essays vary widely in length, complexity, and clarity mainly ranging from concise summaries to extensive narratives detailing life experiences, professional achievements, or motivations. Such broad variability is vital for rigorously evaluating the flexibility, robustness, and fairness of the NLP and scoring mechanisms integrated within the HAIPSE framework. By intentionally incorporating this diversity, the study ensures that the developed methods and systems can effectively generalize

across various applicant scenarios and maintain equitable outcomes irrespective of differences in applicant profiles or essay styles (Binns, 2018).

Through the explicit combination of quantitative and qualitative approaches, reliance on authentic application datasets, and deliberate inclusion of varied application types, the research design provides a strong foundation for comprehensively assessing and validating the HAIPSE system's capabilities (Geiger et al., 2020).

3.1.2 Data Extraction and Preprocessing

Secure Data Handling- One of the first priorities in the study domain is ensuring that the data is handled securely and responsibly by having Multi Factor Authentication (MFA) and strict access protocols (NIST, 2020). Figure 11 shows the setting of strict authentication protocols, so that only authorized users, who have legitimate access, can interact with or view the data. In practice, this means using secure login systems (e.g., passwords, multi-factor authentication) and maintaining detailed logs of who accessed what specific data, and when. Additionally, all collaborators and data handlers are required to sign Non-Disclosure Agreement (NDA) to legally commit to the privacy and security of the information (Swire & Ahmad, 2011). These measures protect sensitive applicant information and ensure the compliant with institutional policies and any regulatory standards.

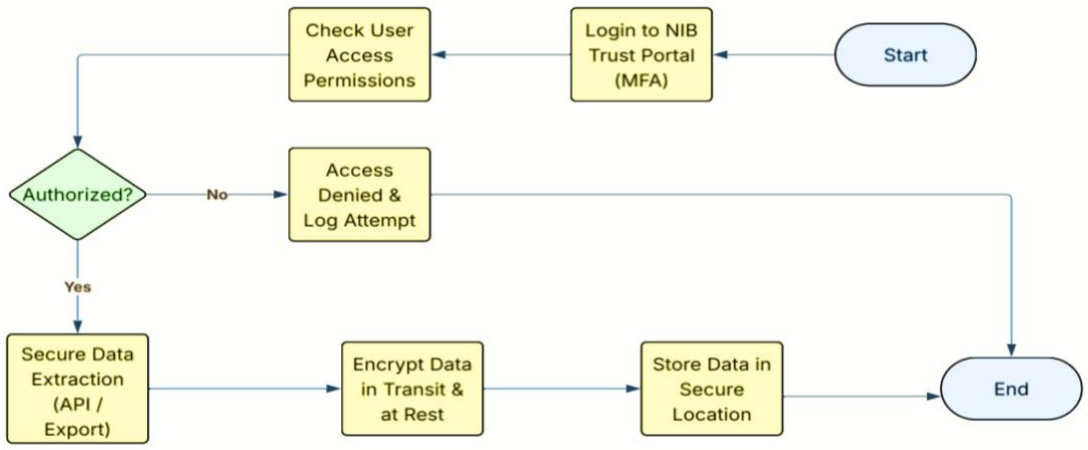


Figure 11. High-Level Flowchart for Secure Data Handling from the User Interface of the NIB Trust Portal.

Supporting Document Formats- The dataset for this study includes different types of documents that applicants submit as part of their applications. These documents come in various formats, such as PDF, JPEG, or PNG for identification documents like ID's or Canadian citizenship card, and text-based formats for essays and other written responses. By accepting multiple file formats, the system accommodates the different ways in which information is stored and shared, ensuring that no data is lost or ignored simply because it was submitted in one format versus another (Weiss et al., 2021). For example, a scanned image of an ID may require conversion into a textual format for further processing, while essays can often be processed directly as digital text. This multi-format approach is critical for building a versatile and inclusive data pipeline capable of handling the diverse inputs that real-world applications provide (Deng & Yu, 2014).

Data Cleaning, Extraction and Preprocessing- Once the data is collected, the first essential task is data cleaning, which ensures the quality and consistency of the dataset before it is subjected to analysis. The initial step involves removing duplicate records to ensure that each application is represented only once. Duplicate entries can distort the analytical outcomes and lead to misleading insights, undermining the reliability of any conclusions drawn from the data (Dasu & Johnson, 2003). Following deduplication, standardizing the textual content becomes a priority. This includes correcting spelling errors, normalizing cases, and ensuring consistency in formatting. Such standardization is critical for downstream processes like keyword extraction or sentiment analysis, which require uniformly structured input data to function effectively (Weiss et al., 2015; Jurafsky & Martin, 2021).

In cases where applicants have submitted scanned documents, Optical Character Recognition (OCR) software is employed to convert image-based data into machine-readable text. This transformation is essential because NLP systems can only process textual inputs, making OCR a vital bridge between visual and textual data (Smith, 2007; Yao et al., 2019). The application of OCR not only expands the types of input data that the system can handle but also ensures that valuable information contained in scanned documents is not excluded from analysis.

Data extraction and preprocessing are further enhanced by incorporating robust security mechanisms that safeguard sensitive applicant data. Ensuring the confidentiality and integrity of the data collected is a foundational requirement in systems that manage personally identifiable

information, as outlined by best practices in digital security (Kent & Souppaya, 2013). In addition to security, the system is designed to accommodate a wide range of file formats, including PDFs, JPEGs, PNGs, and DOCX documents, making it adaptable to the varied formats typically submitted by applicants (Yao et al., 2019).

This careful and inclusive preprocessing stage sets the groundwork for both automated and human-in-the-loop analyses. By ensuring the data is clean, secure, and standardized, the HAIPSE framework is better equipped to perform accurate evaluations and deliver fair and transparent outcomes. Human-in-the-loop approaches are particularly effective in contexts where judgment, interpretability, or nuanced decision-making is required, complementing automated processes to improve reliability and fairness (Holzinger, 2016). As a result, this phase not only prepares the data for technical processing but also reinforces the ethical foundations of the system by promoting accountability, inclusiveness, and trust (Chouldechova & Roth, 2020).

3.1.3 Ethical and Compliance Considerations

When designing and implementing a system like HAIPSE, it is essential to address ethical and compliance issues at every stage. These issues help ensure that the system is used responsibly, respects applicant privacy, and operates fairly.

Bias and Fairness- The system is designed to follow the principles of responsible AI as outlined in leading studies by Barocas & Selbst (2016) and Mehrabi et al. (2021). This involves actively working to minimize any biases in decision-making. For example, the system is equipped with measures to detect and reduce the use of biased language or criteria that could unfairly advantage or disadvantage certain groups of applicants (Raji et al., 2020). By integrating fairness checks into the scoring algorithms and continuously monitoring the outcomes, the system ensures that every applicant is assessed on a level playing field. This means that regardless of personal background, gender, or other factors, each application is evaluated based on merits and relevant qualifications (Hajian et al., 2016).

Data Security- Protecting the personal information of applicants is a critical priority. To ensure data security, all personally identifiable information (PII) is stored in encrypted databases. This encryption ensures that the data is scrambled and unreadable to unauthorized users, and only

those with valid decryption keys can access it (Mell & Grance, 2011; Vogt & Von dem Bussche, 2017). In addition, the system employs role-based access control (RBAC) mechanisms, which allow access permissions to be granted based on a user's role within the organization. This ensures that only specific personnel such as system administrators or authorized reviewers can view or modify sensitive information (Sandhu et al., 1996). These technical and administrative safeguards are designed to prevent unauthorized access, maintain confidentiality, and ensure that all applicant data is used solely for purposes explicitly stated in the application process (Kshetri, 2021). The integration of encryption provides a strong foundation for securing data within AI-enabled evaluation platforms, aligning with best practices in data protection and privacy-by-design approaches (Cavoukian, 2009).

Informed Consent- Transparency is essential for ethical data handling. Before applicants submit their information, it is crucial that they are fully informed about how their data will be used. This includes details on how the system employs automated decision-making processes, the role that human experts play in reviewing these decisions, and the overall scope of data usage (Watcher et al., 2017). Ensuring that applicants have granted informed consent means that they understand and agree to these processes. This is achieved through clear communication typically in the form of privacy notices or consent forms that explains the purpose of data collection, how the information will be processed, and what rights applicants have regarding their data (Cavoukian, 2009). By doing so, the system not only complies with legal and ethical standards but also builds trust between the applicants and the evaluators (Mittelstadt et al., 2016).

Addressing ethical and compliance considerations means designing HAIPSE with robust mechanisms to ensure fairness, protect applicant privacy, and maintain transparency. These measures help to mitigate potential biases, secure sensitive data against unauthorized access, and ensure that all applicants are clearly informed about how their information will be used throughout the application evaluation process (Jobin et al., 2019).

3.2 HAIPSE Framework: Modules and Methods

The HAIPSE architecture includes the following core modules: (1) User Interface Module, (2) Computer Vision Module, (3) NLP Module, (4) Expert System Component, (5) Large Language Model (LLM) Summarization Module, (6) Cohort Fairness Score Module, (7) Human Feedback

Module, as presented in Figure 12. The internal operations of, and the relevant methods used in, these core modules are described in the corresponding sections of this chapter.

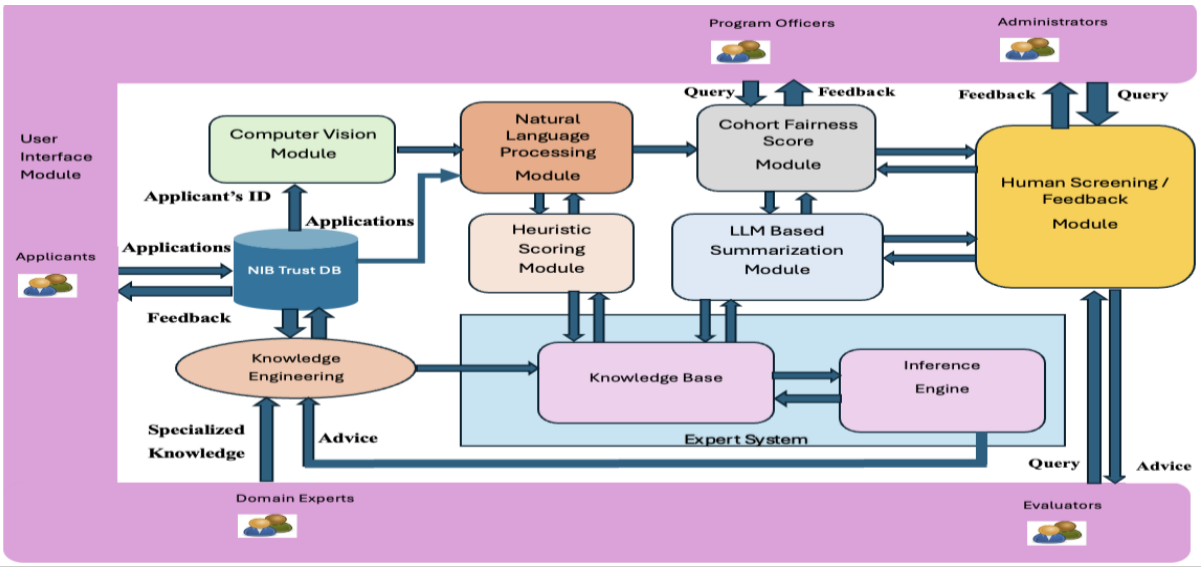


Figure 12. Architecture of the HAIPSE Core Modules.

A high-level workflow of the HAIPSE Framework is shown in Figure 13.

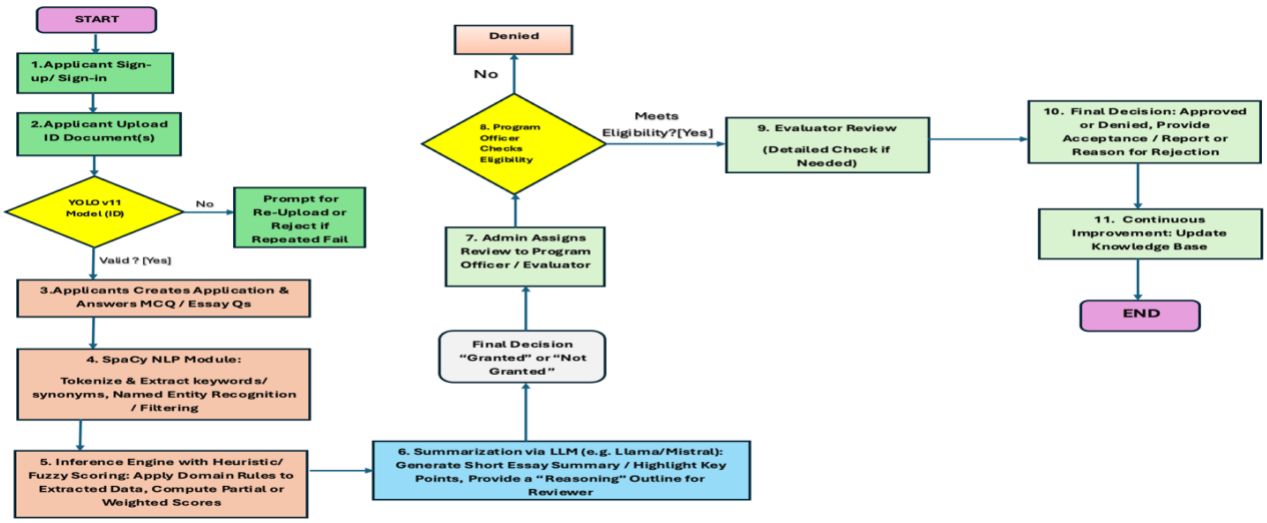


Figure 13. A high-level workflow of the HAIPSE Framework.

It is important to note that the HAIPSE framework integrates a human-in-the-loop ideology, where program officers assess the AI-generated summaries and scores. Evaluators cross-

verify the automated decisions and can adjust the outcomes based on their judgment, particularly in borderline cases. This collaborative approach ensures that any potential biases or errors by the AI are addressed (Buolamwini & Gebru, 2018). Moreover, the framework is designed for continuous improvement, with regular updates to the models and inference rules based on evaluator feedback and new data trends. Overall, the HAIPSE framework balances automation and human oversight to deliver a streamlined, efficient, and fair application evaluation process.

3.2.1 User Interface Module

The User Interface (UI) Module is designed to offer a seamless, transparent, and secure experience for both applicants and evaluators, while orchestrating the complex workflow of the HAIPSE framework.

The UI Module serves as the central hub for all interactions between applicants, evaluators, and the HAIPSE system. Applicants upload their documents, such as IDs and application forms, directly through the UI Module which stores them in the NIB Trust database. From the NIB Trust database, evaluators or admin can query specific applications, override automated scores, or adjust scoring weights, with every activity logged and fed back into the Expert System's Knowledge Base.

This functionality is supported by a security mechanism integrated into the NIB Trust Portal, as demonstrated in Figure 11. It features multi-factor authentication and role-based access controls to ensure that only authorized users can view or modify data, and detailed audit trails capture every upload, query, and override for traceability. The categories of HAIPSE users along with their primary roles are presented in Table 5.

3.2.1.1 Applicant Authentication and Document Validation

In any system handling sensitive and high-stakes applications, establishing robust identity verification protocols is essential (Barocas & Selbst, 2016). For this purpose, the HAIPSE framework implements the security and integrity of applicant data starting with the secure sign-up and login processes, wherein applicants are authenticated using credentials, such as

email/password or Single Sign-On (SSO) mechanisms (Das, 2019). The online Applicant Registration Form is shown in Figure 14.

Table 5. Categories of HAIPSE Users and their Primary Roles.

Category of User	Action(s)
Applicant	Sign-up / Sign-in to NIB Trust Portal Upload ID and supporting documents Complete MCQ's and essay questions
Program Officer	Review automated outputs (ID checks, NLP summaries, heuristic scores) Provide feedbacks on applications Flag complex cases for human evaluation
Administrator	Assign applications to Program Officers Configure heuristic scoring parameters and cohort fairness score triggers Manage access controls (MFA, role permissions)
Evaluator	Perform detailed review of flagged cases Contribute edge-case feedback to refine the knowledge base of the system Verify LLM-generated summaries of the applications
Domain Expert	Provide the rules and edge-case insights that capture real-world expertise in the problem domain Review system rule sets and test cases Ensure the knowledge base stays accurate and relevant

The system is designed to accept a particular piece of government-issued ID, such as national identity card, adhering to standardized documents formats (Shrestha et al., 2021) in a

variety of file formats while implementing automated checks to detect corrupted or incomplete uploads (Jain & Bolle, 2006). If a file does not meet the required standards (e.g., being blurry, incomplete or poorly scanned), the system immediately prompts the applicant to resubmit a clearer, complete version of the document. This required step helps prevent errors or delays in the later stages of processing, where accurate and reliable data is crucial for further evaluation within the HAIPSE framework (Yousef et al., 2020). This approach was adopted to preserve data privacy and avoid exposure of personally identifiable information, which is consistent with the best practices in privacy-preserving machine learning and ethical AI research (Veale et al., 2018).

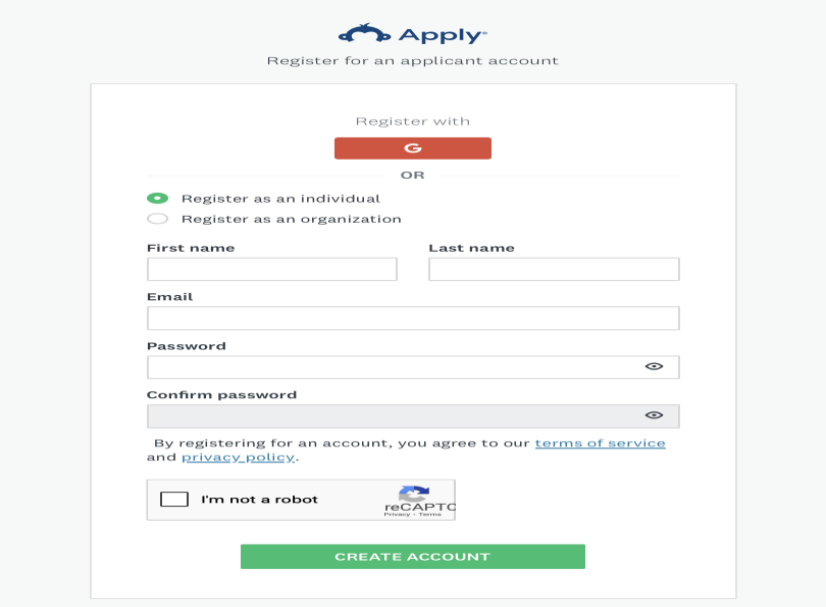


Figure 14. Online Applicant Registration Form.

3.2.1.2 Application Submission

The UI module captures structured and unstructured applicant information (Cappelli, 2019; Rivera, 2012) through Essay and Multiple-Choice Questions (MCQs). MCQs are used to gather quantifiable data in a standardized format, allowing for efficient screening and the inclusion of critical filter criteria (Smith & Martin, 2019). A sample MCQs and Open-Ended Responses recorded on the NIB Trust portal is shown in Figure 15.

Essay responses provide a richer, qualitative understanding of each applicant's motivations, experiences, and personal insights (Kahneman, 2011). A sample Essay response on the NIB Trust

portal is presented in Figure 16. Together, these two formats create a comprehensive foundation for robust application's evaluation within the HAIPSE framework (Mehrabi et al., 2021).

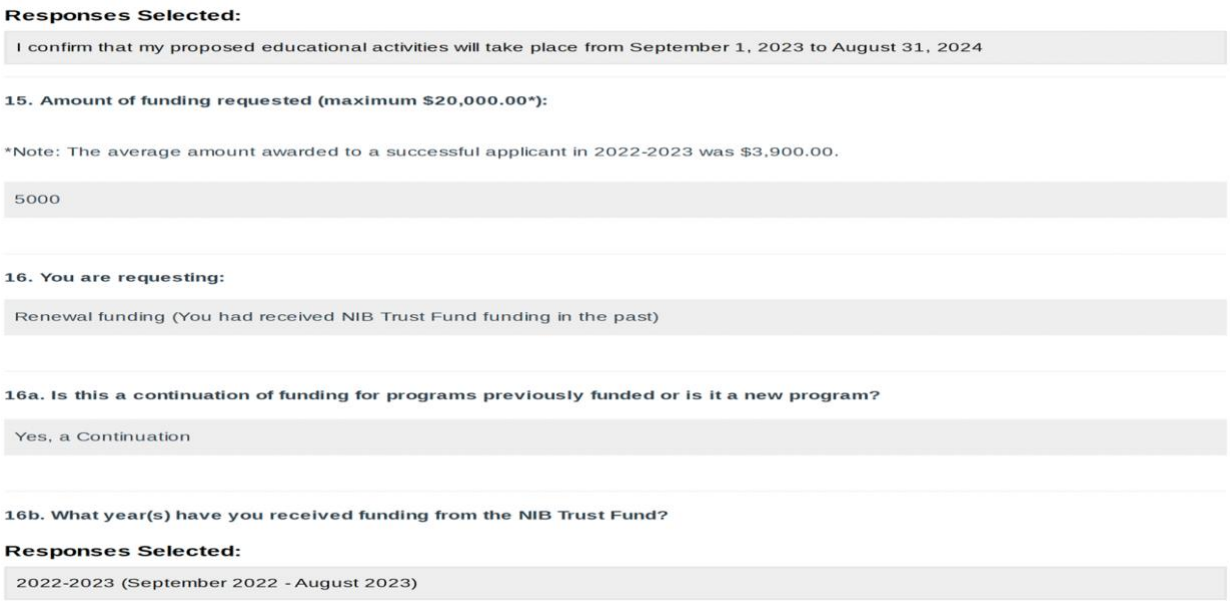


Figure 15. Sample Multiple Choice Questions (MCQs) and Open-Ended Responses on the NIB Trust Portal.

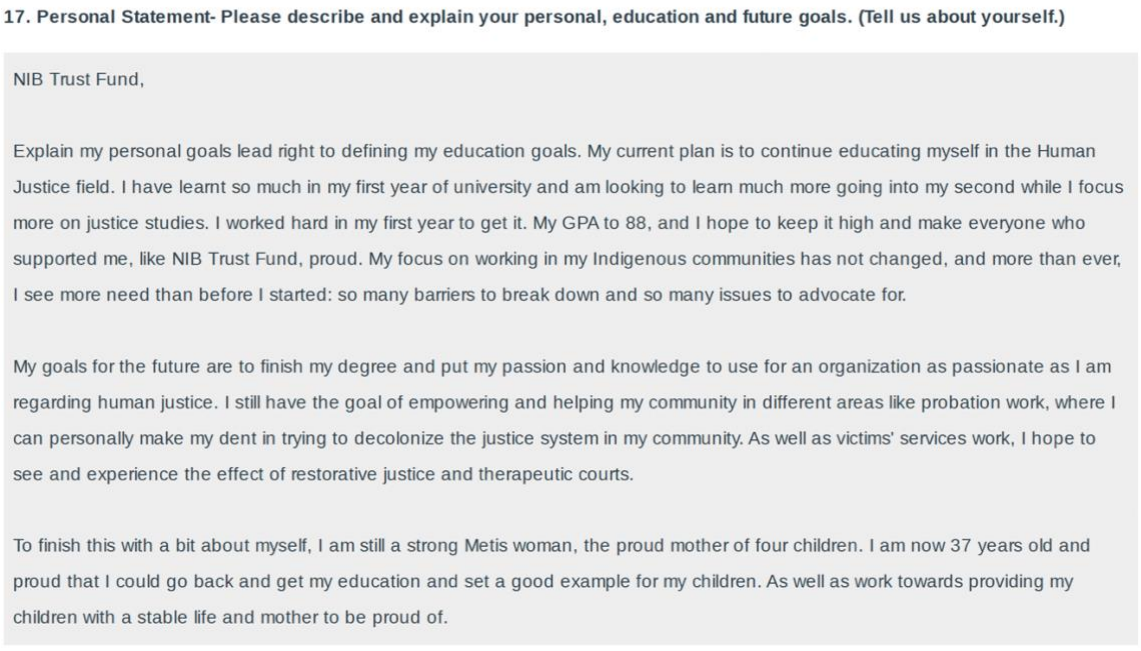


Figure 16. Sample Essay Response on the NIB Trust portal.

3.2.2 Computer Vision Module

To ensure the authenticity of the identification documents submitted by applicants, the HAIPSE framework incorporates a state-of-the-art Computer Vision solution based on the YOLOV11 model (Redmon et al., 2016). YOLO (You Only Look Once Version) is a deep learning model widely recognized for its rapid and accurate object detection capabilities. The architecture of the YOLO V11 largely includes three components: 1) Backbone; 2) Neck; and 3) Head, as demonstrated in Figure 17.

The Backbone of YOLO V11 converts raw images into rich, multi-scale feature maps. It builds on a Cross-Stage Partial (CSP) design, using compact C3k2 residual blocks to reduce parameters and accelerate computation while preserving gradient flow (Wang et al., 2020; Khanam & Hussain, 2024). A streamlined Spatial Pyramid Pooling-Fast (SPPF) layer then pools information at several receptive-field sizes, giving the network global context without heavy cost (He et al., 2014). Finally, a C2PSA attention module re-weights spatial regions and channels so that salient object cues are emphasised and background noise is suppressed (Woo et al., 2018). These three elements together yield lightweight yet expressive features for downstream detection.

The Neck fuses those backbone features across scales to ensure that both fine detail and high-level semantics reach the detector. YOLO V11 stacks a top-down Feature Pyramid Network (FPN) path with a bottom-up Path Aggregation Network (PANet) path, so information flows bidirectionally (Lin et al., 2017; Liu et al., 2018). After each up- or down-sampling step, another C3k2 block refines the concatenated maps with minimal overhead, while the attention maps propagated from C2PSA keep the focus on object regions. The result is a trilogy of feature maps (at strides 8, 16, 32) that are simultaneously rich in context and sharp in localisation, improving detection of small, medium, and large objects.

The Head turns these multi-scale maps into final predictions. YOLO V11 adopts an anchor-free, decoupled head: one branch regresses bounding-box offsets and another predicts abjectness plus class scores, allowing each task to specialise (Ge et al., 2021). A final pass through lightweight C3k2 blocks further polishes the features before a convolution output detection at the three scales. This anchor-free design removes hand-tuned anchor boxes and, together with the

decoupled branches, simplifies training while yielding accurate, real-time results across diverse object sizes.

The YOLO V11 object detection head is shown in Figure 18. In the context of ID validation, the model is trained on a diverse dataset of ID images so that it can recognize and scan critical regions of government-issued documents such as the Machine-Readable Zone (MRZ) found on passports or other ID's which contains standardized personal details essential for verification. During the scanning process, the YOLO V11 model processes each ID image in a single forward pass, swiftly detecting whether the critical text zones are present and legible (Redmon et al., 2016).

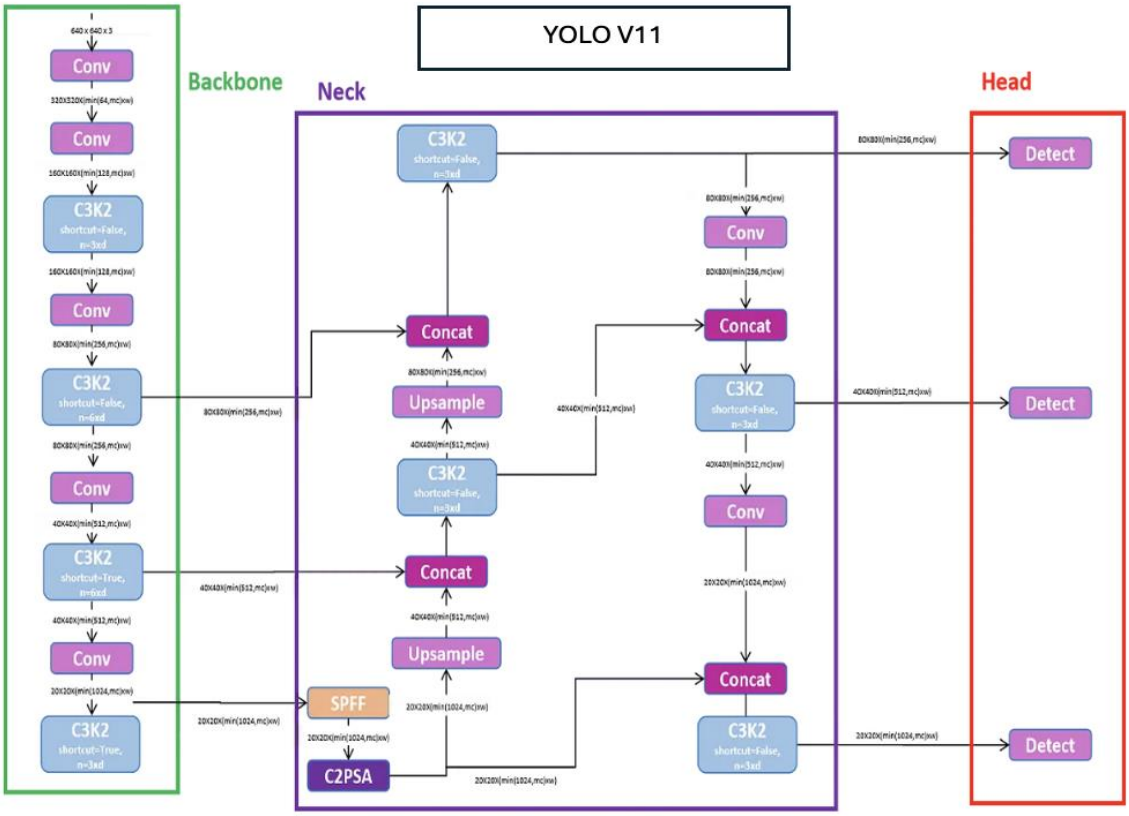


Figure 17. Architecture of the YOLO V11 (From: Rao, 2024).

The YOLO V11 model is specifically fine-tuned to differentiate between acceptable variations in document appearance and significant deviations that could indicate tampering,

degradation, or poor-quality uploads (Jain, 2006). These capabilities are critical for ensuring that only clear and verifiable documents are used in subsequent steps of the evaluation process.

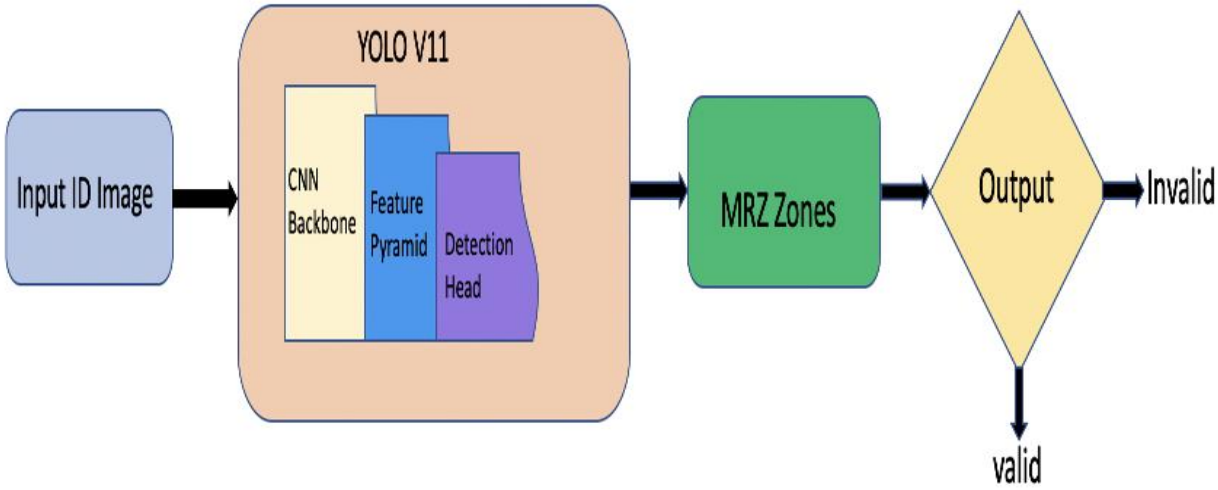


Figure 18. YOLO V11 Model ID Validation Pipeline.

When the applicants upload a picture of their ID, YOLO V11 scans the picture in one quick look and points out the important parts of the document: the photo, the two lines of numbers and letters at the bottom (the MRZ), date of birth, and other key spots.

Next, the text inside those boxed-off areas is read by a small text-reading tool. The system checks that the MRZ follows the official passport format and that the numbers inside it add up correctly (they have built-in checksums, like mini math problems) (Shrestha et al., 2021). It also makes sure dates and names match in different places. If everything is present and easy to read, the ID is marked “Valid” and the applicant can move on. If something is missing or blurry, the ID is marked “Invalid” and the applicant is prompted to re-try.

Based on the outcome of the computer vision analysis, the system makes a definitive determination. If the ID is successfully validated (i.e., all critical text zones are clear and correctly formatted), the application proceeds to the main review process, allowing the applicant to continue smoothly. However, if the model flags significant issues, such as blurry text, incomplete sections,

or other discrepancies, the system immediately initiates a feedback loop by prompting the applicant to re-upload a corrected version of the document. Should repeated validation failures occur, the application may be rejected to maintain the integrity of the data (Mehrabi et al., 2021).

3.2.3 Natural Language Processing Module

The NLP Module is a central component of the HAIPSE framework, designed to transform unstructured applicant text into structured data that can be analyzed. This module leverages advanced natural language processing techniques to extract meaningful features from essays and other narrative responses following the workflow demonstrated in Figure 19. The processing pipeline begins with tokenization and parsing using the spaCy library (Honnibal & Montani, 2017), which splits the text into tokens, segments sentences, and normalizes the language for uniformity. By removing trivial words through stop word elimination, the system ensures a sharper focus on the critical contents.

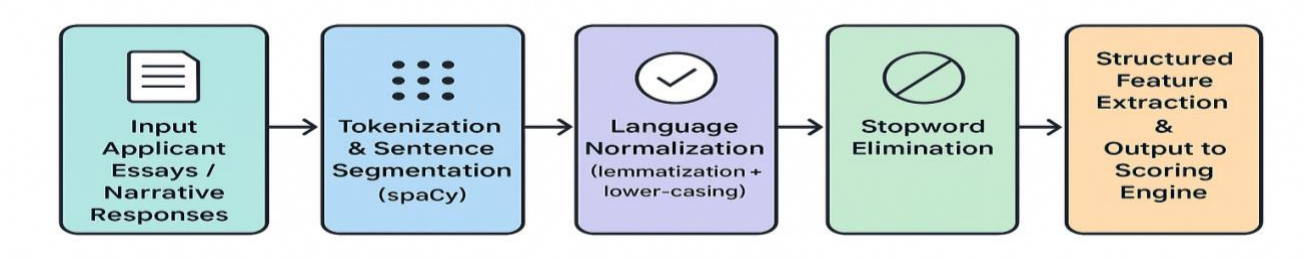


Figure 19. Natural Language Processing Module Workflow.

3.2.3.1 Tokenization and Parsing

The first step in processing raw text data is known as tokenization and parsing. This step involves breaking down the text into smaller units, or tokens, which include individual words, punctuation marks, and other symbols. By using the SpaCy library (Honnibal & Montani, 2017), the system effectively splits the text into these meaningful building blocks, which are essential for any further analysis. Additionally, SpaCy also performs sentence segmentation, which helps to divide the text into complete sentences. This preserves the grammatical structure of the content, making it easier for the system to understand the context and relationships between words and phrases.

Normalization is another vital part of this step. It involves converting all characters to lowercase and standardizing text formats, which helps to reduce inconsistencies that might arise from different text styles or formats. An important aspect of this normalization process is stop word removal. Common words such as “the,” “and,” or “but” are filtered out because they typically do not add significant meaning to the text, thus allowing the analysis to focus on more critical content. Together, these processes ensure that the text is cleaned, uniformly formatted, and reduced to its core components, setting a solid foundation for deeper language analysis in subsequent stages of processing (Bird et al., 2009).

3.2.3.2 Domain Adaptation

Domain adaptation is the process of adjusting a pre-trained NLP model so it can better understand text specific to a given field or context (Ruder, 2019) following the workflow shown in Figure 20. In the case of SpaCy, this involves fine-tuning the model or extending it with specialized lexicons that capture the terminology, phrasing, and conventions unique to a particular domain such as the nonprofit or educational sector (Honnibal & Montani, 2017). By integrating additional vocabulary and custom entity types into SpaCy, the model can accurately recognize terms and references that would otherwise remain unrecognized or misclassified if it only relied on a generic, broad-coverage language model (Mehrabi et al., 2021).

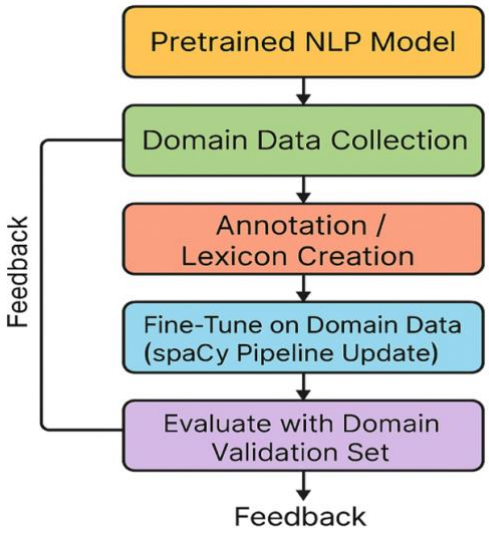


Figure 20. Workflow of the Domain Adaptation.

3.2.3.3 Keyword Extraction

Keyword Extraction refers to the process of identifying and highlighting the most important words or phrases within a text, typically those that contribute significantly to the document's thematic or conceptual content (Weiss et al., 2015). In the HAIPSE framework, this step is crucial for summarizing the essence of an applicant's submission and making it readily usable in subsequent scoring and evaluation stages. Two key practices – "mapping to an ontology" and "filtering irrelevant content" – ensure consistency and precision during keyword extraction.

Mapping keywords to a predefined ontology involves linking or replacing synonyms and semantically related terms so they can be treated uniformly (Wang & Zhang, 2020). For instance, the terms "leadership" and "team management" might have distinct literal meanings yet represent similar concepts within the context of an application. By establishing these relationships in an ontology, the system ensures that references to leadership are scored or interpreted consistently, regardless of how they are phrased. This approach is especially helpful in large-scale or domain-specific contexts, where different applicants might use varied terminology to describe the same skill or experience (Zhang & Zhao, 2019). Through ontology mapping, the HAIPSE framework achieves a more comprehensive and fair assessment of each candidate's background and qualifications.

Once the relevant keywords have been identified and unified via ontology mapping, the next step involves discarding extraneous statements or off-topic segments of text (Weiss et al., 2015). This filtration mechanism prevents noise such as tangential anecdotes from diluting the focus of the analysis. In practice, machine learning classifiers or rule-based systems can flag and remove these non-contributory elements. The objective is to isolate only those pieces of information that add genuine value to the assessment process (Barocas & Selbst, 2016). By refining the textual dataset in this manner, the HAIPSE framework enhances the reliability of its evaluations, ensuring that subsequent metrics and decisions are based on content directly relevant to the applicant's skills and qualifications.

3.2.4 Expert System Component

The core element of the Expert System is the Knowledge Base which stores domain-specific information acquired from the human experts ("Knowledge from an expert"), typically encoded as heuristic rules and facts relevant to the specific problem domain. Each rule is structured in the form:

IF (condition or premise) THEN (action or conclusion).

Sample Eligibility Rules are given in Box 1.

IF ("age" < 18) THEN (flag application as "ineligible")

IF ("citizenship" is not Canadian) THEN (flag application as "ineligible")

IF ("indigenous ID" is not present) THEN (flag application as "ineligible")

IF ("school transcript" is not present) THEN (flag application as "ineligible")

IF ("age" ≥ 18) AND ("citizenship" is Canadian) AND ("indigenous ID" is present) AND ("school transcript" is present) THEN (flag application as "eligible")

Box 1. Sample Eligibility Rules.

The Inference Engine extract facts from the application and applies a keyword-matching algorithm to the documents submitted with the application to detect which rule's premise is satisfied. If it is fully met, the Engine applies the rule, assigning the resulting status to the application (i.e., eligible or ineligible).

In general, the users engage the Inference Engine through the UI module by placing a specific problem or question ("the Query") processed against the rules and facts in the Knowledge Base using its heuristic reasoning capabilities. Here, the Inference Engine acts as the problem-solving mechanism. The resulting "Advice" is delivered back to the user through the UI (Waterman, 1986). Thus, the system effectively captures and replicates human expertise, sometimes called knowledge engineering, making it accessible to non-experts for decision support

or problem-solving within its specialized domain via a structured query-advice cycle (Jackson, 1999). For example, the Heuristic Scoring module interacts with the Inference Engine using rule-based methods, adjusting weights of factors contributing to application's evaluation depending on the scoring outcomes and based on the embedded expert knowledge (i.e., the rules) to fine-tune the HAIPSE operations (Buchanan & Shortliffe, 1984).

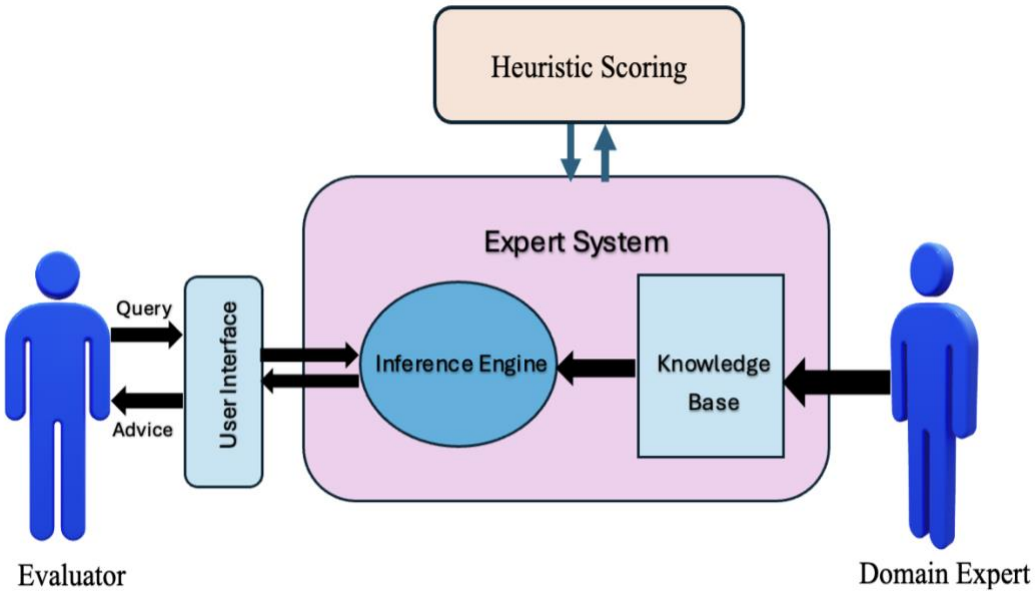


Figure 21. Interaction of the Heuristic Scoring Module with the Expert System Component.

3.2.5 Heuristic Scoring Module

In the HAIPSE framework, the Heuristic Scoring Module and Inference Engine**,** translating raw applicant data into structured evaluations, serve as the decision-making core. According to Mehrabi et al. (2021), one of the key implementation mechanisms involves combining heuristic weighting approach with fuzzy logic to ensure that a broad range of applicant scenarios are scored consistently and fairly.

In the context, the term "heuristic" refers to rules of thumb or simplified decision criteria that enable quick, yet structured judgments in high-volume application processes (Kahneman, 2011). For instance, the Engine might assign to the application "+10 points for 3+ years of relevant

experience in the academics," reflecting the assumption that a certain duration of experience is critically valuable to the position or program in question.

These rules are often built upon domain expertise, making them tailored and transparent. By specifying clear thresholds (e.g., minimum years of experience or must-have certifications), the system streamlines the initial stages of evaluation. However, the exact application of these heuristics can be adjusted for near-threshold values, for example, awarding partial points if an applicant has 2.5 years of experience instead of the full 3 years. This graded approach keeps the scoring process from becoming overly rigid while still respecting the importance of certain key criteria (Mehrabi et al., 2021). Sample Heuristic Scoring Rules are given in Box 2.

IF ("years of relevant experience" ≥ 3) THEN (add 10 points to the application score)

IF ($2.5 \leq$ "years of relevant experience" < 3)
THEN (add 10 * (years of relevant experience / 3) points to the application score)

IF ("years of relevant experience" < 2.5) THEN (add 5 points to the application score)

Box 2. Sample Heuristic Scoring Rules for Relevant Experience.

Fuzzy logic extends these heuristic rules to better handle borderline cases, where rigid cutoffs may inadvertently penalize worthy applicants. Instead of binary decisions fully meeting or not meeting a threshold, fuzzy logic assigns degrees of membership, allowing for partial satisfaction (Zimmermann, 2001).

In practical terms, if an applicant has, for example, 2.5 years of experience, the scoring model would grant them a partial credit rather than a complete denial of points. Through fuzzy logic, the HAIPSE system ensures that, otherwise competitive, candidates are not rejected merely because they fall fractionally short of a prescribed threshold. By combining heuristic weighting assignment with fuzzy logic, the scoring engine achieves a balanced, context-sensitive level, allowing for a more holistic and equitable evaluation of applicant qualifications.

The implementation of the heuristic scoring in the framework is designed to balance structured NLP expert rule-based Heuristic evaluations, Cohort Level Fairness Score and Large Language Models generated summaries in the HAIPSE Total Score, as represented by formula (1):

$$F(A) = \alpha * HS(A) + \beta * CF(A) + \gamma * LLM(A). \quad (1)$$

Here:

F(*A*) is the HAIPSE Total Score.

HS(*A*) is the Heuristic Score for application *A*.

CF(*A*) is the Cohort Fairness Score for application *A*.

LLM(*A*) is the Large Language Models generated summary score for application *A*.

α, β, γ are the weighting factors for Heuristic, Cohort Fairness and LLM Scores, respectively, where $\alpha, \beta, \gamma \in [0, 1]$ and $\alpha + \beta + \gamma = 1$.

In its turn,

$$HS(A) = \sum_{i=1}^{n} [s_i * K_i(A)], \quad (2)$$

where

s_i is the domain specific scores assigned to *i*-th keyword.

K_i is the *i*-th keyword match in the application.

n is the number of keywords in the application *A*.

The computational formulae for CF(*A*) and LLM(*A*) are given in section 3.2.6 and 3.2.7.3.The Heuristic Scoring quantifies the alignment of an application *A* with predefined, domain-specific criteria through NLP module and Expert System component. Parameter α in Formula (1) controls the contribution of the heuristic score to the HAIPSE Total Score F(*A*). Lower α reduces reliance on keyword-based scoring, prioritizing other components of the HAIPSE Total

Score, and vice-versa (Bansal et al., 2021). Adjusting α and s_i dynamically can mitigate biases in the application, as demonstrated in debiasing techniques for rule-based systems (Huang, 2016).

3.2.6 Cohort Fairness Score Module

Within the HAIPSE framework, the Cohort Fairness metric is designed to quantify how consistently a group or a cohort of applications is scored. By grouping applications for instance, into batches of 30 and then examining the spread of scores within each group, the HAIPSE can detect potential bias or systematic inconsistencies in the evaluation process (Dwork et al., 2012; Friedler et al., 2019). A Cohort Fairness Score, CF(*A*), bridges automated and human judgment:

$$CF(A) = \frac{1}{N}\sum_{i=1}^{N}(G(A_i) - \mu_{cohort})^2, \quad (3)$$

where

$G(A_i)$ is the Fairness-adjusted score of the *i*-th application in the cohort.

$\mu cohort$ is the Mean Cohort Fairness Score of all applications in the cohort.

N is the number of applications in the cohort.

Formula (3) captures the variance of the Cohort Fairness Scores around the mean value ($\mu cohort$) of N applications in the cohort. The value of $G(A_i)$ is accounted for factors such as biased language or heuristic weights.

Lower *Cohort Fairness* values indicate that application scores cluster closely around the Cohort's average (Srebro, 2016), suggesting that the evaluation process is relatively consistent. In other words, no individual applicant's score is drastically higher or lower than the mean which indicates the more equitable evaluation (Mehrabi et. al. 2021) within the Cohort.

Higher *Cohort Fairness* values suggest greater variance in scores, potentially indicating the outlier scores or inconsistencies across the applications within the cohort. Such fluctuations might indicate that certain subgroups of applications are being systematically evaluated differently, on the grounds of skill sets, writing style, or other factors, thus signalling about potential bias or uneven standards (Zemel et al., 2013).

The weighting factor β in Formula (1) determines how heavily the variance of the cohort fairness scores impacts the total HAIPSE metric. In particular, β = 1 signifies the strict fairness enforcement while β = 0.3 implies certain tolerance for minor deviations. It can be tuned dynamically to enforce equity in response to bias (Bolukbasi et al., 2016).

Adaptive Refinement- If a Cohort Fairness score is notably high, the framework triggers a review of both the scoring model (e.g, heuristic rules) and the evaluator's decisions. This review may involve retraining or adjusting the scoring algorithm, refining the fairness metrics, or providing additional guidance to evaluators. Sample threshold based decision rules for Cohort Fairness Scores are given in the Box 3.

1. IF ("Cohort Fairness Score" ≥ 30) THEN (trigger "review")
2. IF ("Cohort Fairness Score" < 30) THEN (flagged "fair scoring")

Box 3. Threshold-Based Decision Rules for Cohort Fairness Scores.

If the Cohort Fairness indicator for applications surpasses a pre-defined threshold, the HAIPSE treats the spike as a possible warning that the screening pipeline is drifting toward systematic bias. In such a case, the first remediation step is to retrain the heuristic model on a bias-balanced sample or recalibrate thresholds if the scoring curve has become skewed (Raji et al., 2020; Mitchell et al., 2019).

The second remediation step focuses on the human reviewers. Decision logs are sampled to check whether evaluators systematically up- or down-grade certain profiles after the automated score, a practice that may reintroduce bias despite a neutral model (Holstein et al., 2019). If inconsistent overrides are found, the framework issues targeted guidance or re-training to those evaluators, reinforcing calibrated scoring behaviour and clarifying fairness policies (Binns, 2018). By coupling algorithmic retraining with human-process adjustments, the HAIPSE uses a closed-loop audit to keep both machine and human decision-making aligned with the fairness goals.

3.2.7 Large Language Models Based Summarization Module

The HAIPSE framework leverages a set of fine-tuned Large Language Models (LLMs), such as Mixtral-8×7B-Instruct-v0.1 (Jiang et al., 2023), Meta-Llama-3-8B-Instruct (Meta AI, 2024), Tiiuae/falcon-7b-instruct (Almazrouei et al., 2023) , Databricks' dolly-v2-3b (Conover et al., 2023), and SmolLM2-1.7B-Instruct (Shen et al., 2024), to generate concise summaries of applicant essays. Each model is evaluated for its ability to transform lengthy responses into bullet points or short executive overviews that highlight an applicant's main qualifications, experiences, and achievements. To ensure these summaries are both context-relevant and equitable, a custom fine-tuning step exposes the models to domain-specific essay samples while incorporating fairness checks to guard against omissions or biases Ultimately, these AI-generated summaries reduce the cognitive load on human evaluators, enabling them to quickly grasp each candidate's core strengths without wading through extensive narratives (Wu et al., 2023), while still maintaining transparency and human oversight. The working of the LLM Summarization Module is presented in Figure 22.

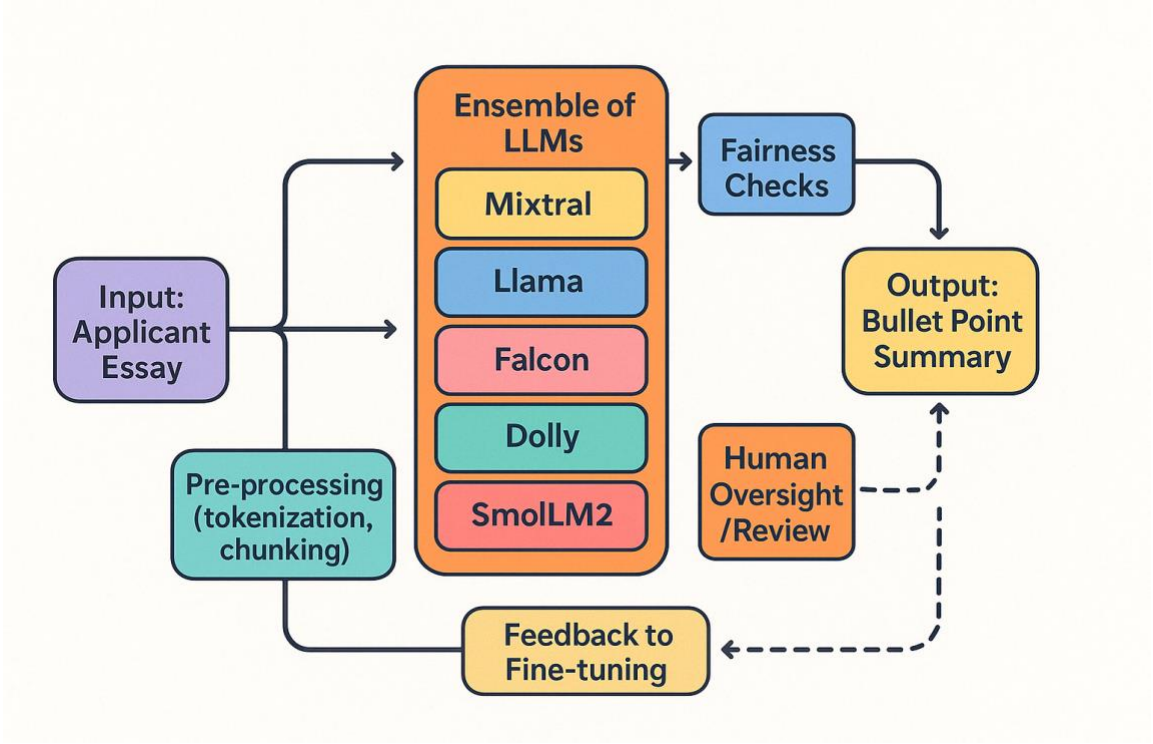


Figure 22. Working of the LLM Summarization Module.

3.2.7.1 Experiments and Model Selection

The first step in building the summarization pipeline is to select a suitable Large Language Model (LLM). In this experiment, several models are tested to evaluate their performance, including Mixtral-8×7B-Instruct-v0.1, Meta-Llama-3-8B-Instruct, tiiuae/falcon-7b-instruct, Databricks' dolly-v2-3b, and SmolLM2-1.7B-Instruct. These LLMs vary in architecture size, instruction-tuning approach, and intended application domain. To serve the needs of an application evaluation workflow, the chosen model(s) must reliably produce concise bullet points or executive summaries that capture each applicant's main arguments, experiences, and achievements. By condensing lengthy essay responses into a more accessible format, the system eases the cognitive load on human evaluators, allowing them to focus on critical decision-making task. Among all the tested LLMs, the Meta Llama transformer model demonstrated the best performance and, therefore, is used in the HAIPSE framework.

The architecture of the Meta-Llama transformer model is shown in the Figure 23 representing one repeating block in the Meta-LLaMA decoder-only Transformer, followed by the final output layer. All blocks are stacked *N* times to build the full model (Touvron et al., 2023). Tokens are first mapped to learned embedding vectors. Rotary positional encodings (RoPE) are then applied in-place, rotating each query/key in the complex plane so that relative positions are preserved even for sequences longer than those seen in training (Su et al., 2021).

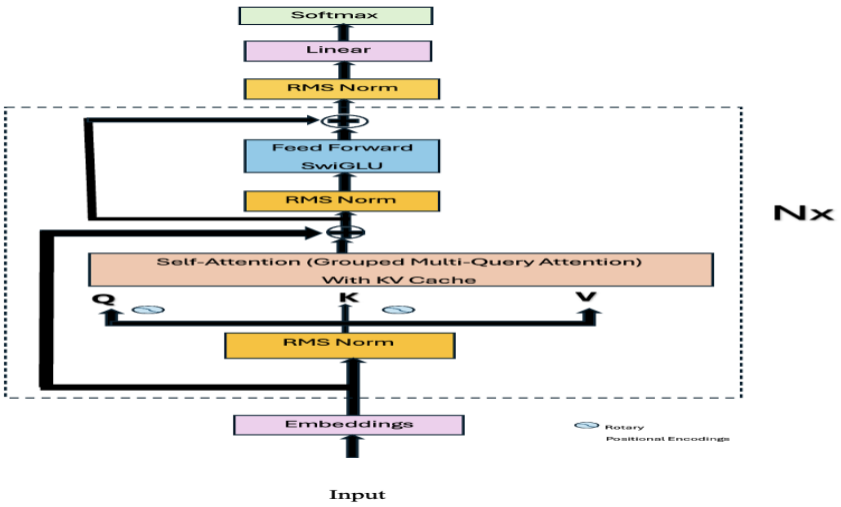


Figure 23. Architecture of Meta-Llama Transformer Model.

Before self-attention, activations pass through RMSNorm, which rescales each hidden vector by its root-mean-square value. Because it skips the mean-centering term of Layer Norm, RMSNorm reduces arithmetic and speeds up both training and inference while providing comparable stabilisation in deep stacks (Zhang & Sennrich, 2019).

LLaMA replaces classic multi-head attention with grouped multi-query attention (MQA). Many query projections are maintained for expressiveness, but a much smaller set of shared key/value (KV) projections is used, cutting the KV-cache memory footprint by an order of magnitude during autoregressive decoding (Shazeer, 2019). RoPE-rotated QKV tensors allow each token to attend to all previous tokens in the sequence, yielding efficient long-context generation (Touvron et al., 2023). The attention output is added back to the layer input (residual-skip) and normalised again with RMSNorm. This is followed by a two-layer feed-forward network (FFN), consisting of two positions-wise linear transformation. The activation is SwiGLU, a gated linear unit that multiplies a SiLU-activated gate by a parallel linear branch improving optimisation stability and running ≈10% faster than GELU while matching or surpassing its accuracy (Shazeer, 2020).

After the last decoder block, one more RMSNorm is applied. The sequence then passes through a linear projection to vocabulary logits, and a soft-max function transforms these logits into token probabilities for sampling or beam search (Touvron et al., 2023). By combining RMSNorm, grouped MQA, RoPE, and SwiGLU, LLaMA achieves GPT-class performance with lower memory use and faster throughput, especially important for very long contexts or resource-constrained deployments.

3.2.7.2 Custom Fine-Tuning and Summarization Outputs

Once the preferred LLM is chosen, it undergoes an additional phase of fine-tuning. This process adapts the model to the specific demands of grant or hiring contexts, exposing it to domain-relevant essay samples and ensuring that it learns the nuances of evaluated project proposals, and personal statements (Howard & Ruder, 2018). Fine-tuning improves both the quality and conciseness of the summarization outputs. However, the system must also safeguard against biases that might be present in either the pre-trained model or the fine-tuned dataset. As a result, the summarization

workflow includes regular checks for fairness and omissions of key details. This involves ongoing reviews of the LLM's outputs, adjustments to the training data, or modifications to the instruction prompts. By proactively monitoring and mitigating biases, the platform strives to maintain equitable, high-quality summaries for every application (Gehman et al., 2020).

The final output takes the form of evaluator-facing summaries, offered in bullet-point or short-paragraph format. These summaries highlight the key contributions and qualifications of each application, highlighting crucial experiences, such as relevant work history, leadership roles, or academic qualifications. To uphold transparency, the system clearly indicates that these summaries are generated by an AI model and are subject to human verification (Buolamwini, 2018), as shown in Figure 24. This disclaimer reminds evaluators that, while the summaries can streamline their review process, ultimate responsibility lies with human decision-makers. Together, these protocols in model selection, custom fine-tuning, and transparent output design create a robust and ethically guided summarization framework (Doshi-Velez & Kim, 2017) that caters to high-stakes application screening.

Figure 24. Summaries Generated by an AI Model Subject to Human Verification.

3.2.7.3 LLM Summary Score

The LLM(A) score in Formula (1) within the HAIPSE Total Score is derived from mean (μ) and variance (σ^2) of the LLM generated summary scores for application A. Formula (4) penalizes inconsistency in the LLM's summary outputs, ensuring reliability and mitigating bias:

$$\text{LLM}(A) = (\mu * (1 - k \cdot \frac{\sigma^2}{\mu^2})), \quad (4)$$

where

k is the penalty factor ($0 \leq k \leq 1$), a constant that controls the strength of the penalty for high variance. In particular, $k = 0$ means that no penalty is applied (the LLM score equals the mean) while $k = 1$ implies the full penalty whereby the score is reduced proportionally to normalized variance.

μ is the average mean score of the LLM-generated summary.

σ^2 is the variance of the LLM-generated summary score.

In Formula (4), the term $\frac{\sigma^2}{\mu^2}$ (squared coefficient of variation) ensures the scale-invariant adjustments. The k values within the range (0.1 – 0.3) are used to prevent the excessive penalties.

The average mean score of the LLM-generated summary for application A, μ, is computed as:

$$\mu = \sum_{i=1}^{M} l_i , \quad (5)$$

where

M is the number of summaries.

l_i is the LLM-generated summary for i-th application.

The variance σ^2 of the LLM-generated summary score is given by the formula:

$$\sigma^2 = \frac{1}{M} \sum_{i=1}^{M} (l_i - \mu)^2 . \quad (6)$$

A higher value of γ in Formula (1) increases the weight of the LLM's output relative to other contributing components, whereas a lower γ shifts more decision-making influence on the human or heuristic side (Choudhary *et al.*, 2023). For example, fairness audits can be used to detect systemic biases in the LLM's behaviour (Gallegos *et al.*, 2024). If such bias is identified in the system, human can proactively reduce γ to decrease reliance on the biased model output, thereby mitigating disparate impacts (Li *et al.*, 2023). Overall, adaptive ensemble weighting combined

with human-in-the-loop oversight has been shown to improve fairness and equity in algorithmic decision-making systems (Kenfack *et al.*, 2021).

3.2.8 Human Feedback Module

After the automated evaluation phases, such as identity validation, NLP-based text analysis and heuristic scoring and LLM summarization have been completed, the HAIPSE framework subjects applications to a structured, human-in-the-loop review (Kleinberg et al., 2018). This review begins with assigning applications to the appropriate program administrators, who conduct an initial screen to filter out obviously ineligible submissions. Next, borderline or complex cases undergo a detailed examination by evaluators, who verify LLM summaries and scores, adjusting for overlooked qualities or biases, when necessary. Through this layered approach, the HAIPSE blends the efficiency of AI with the nuanced judgment of human experts, aiming for both bias elimination/reduction (Mehrabi et al., 2021) and fairness improvement in high-stakes application evaluations (Amershi et al., 2019), as shown in Figure 25.

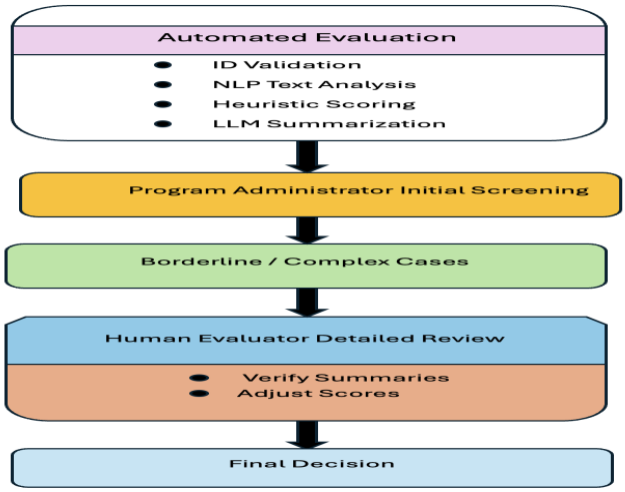


Figure 25. Human-AI Collaborative Application Evaluation Workflow.

3.2.8.1 Assignment to Program Administrators

A Program Administrator can provide feedback on each completed application based on the score evaluated by the system, as shown in Figure 26. By centralizing and organizing this process, the

platform ensures efficiency and reduces the risk of applications getting overlooked. Additionally, the system applies a batch evaluation approach, wherein applications are grouped into sets (e.g., of 30-50 applications) at a time to facilitate comparative fairness checks (Friedler et al., 2019). By reviewing aggregated batches, Program Administrator can quickly identify scoring inconsistencies or emerging biases.

Figure 26. Program Administrator Sample Feedback on an Application.

3.2.8.2 Screening by Program Officer

Program officers conduct a high-level review of each assigned application (Cappelli, 2019). This stage involves an initial filter**,** in which any submissions that fail to meet mandatory prerequisites, such as valid identification or minimum educational criteria, get rejected immediately. Borderline cases, however, undergo a deeper assessment, especially if the LLM summaries suggest potential merit despite not strictly meeting every stated requirement (Van Esch et al., 2019). This targeted approach confers efficiency gains, as Program Officers can dedicate more time and attention to nuanced applications rather than exhaustively reading every single submission. Consequently, human expertise is concentrated on the most complex or uncertain cases, creating a more balanced and effective evaluation process.

3.2.8.3 Detailed Evaluator Review and Final Decision

For all the evaluated applications, the system calls for a more thorough detailed evaluator review (Mehrabi et al., 2021). Here, evaluators cross-check LLM summaries against the applicants' original essays to confirm consistency and completeness. If the automated scores appear to

misjudge certain experiences or if the system has inadvertently penalized nontraditional backgrounds, evaluators perform a score adjustment by updating the final score to better reflect the applicant's true qualifications (Binns *et al*., 2018). By actively monitoring for anomalies and patterns of disparity, the process seeks to maintain integrity and impartiality in the evaluation pipeline.

At the end of the review process, each application receives a final decision. If approved, the system generates an acceptance letter that includes key details like funding or admission information (Strohmeier & Piazza, 2015). If denied, the platform provides clear explanations, for example, noting that a requirement was not met, so that the applicants understand the specific reason for rejection (Brown et al., 2020). During this last stage, every decision and its justification are documented for future audits and stored in the NIB Trust database. This record-keeping not only supports accountability but also allows the evaluation framework to be updated as needed, making the overall process fairer and more transparent (Amershi et al., 2019).

3.2.9 Adaptive Knowledge Base and Model Fine-Tuning

Evaluators and reviewers who interact with the system provide ongoing feedback, such as flagging unfamiliar terms, identifying edge cases, or suggesting more appropriate scores for unique applicant profiles. This human-in-the-loop input is used to refine NLP models and update scoring heuristics, ensuring that the system's interpretative capacity and fairness continually improve (Amershi et al., 2014; Holzinger, 2016).

By regularly incorporating human insights, the HAIPSE framework becomes more susceptible to capturing the nuances in applicant language, reducing potential biases, and adapting to emerging trends or terminologies. As a result, the adaptive Knowledge Base (Amershi et al., 2019) contributes significantly to maintaining a high level of fairness, ensuring that the system evolves in line with organization-specific demands (Barocas & Selbst, 2016; Mehrabi et al., 2021).

Model retraining and fine-tuning means updating the system's AI models regularly so that they continue to perform well as new data or changes occur. Over time, the language and information that applicants provide may evolve, and new kinds of information might need to be taken into account. By retraining, the system is fed new examples and data so it can learn these

changes and adjust its internal patterns (Raffel et al., 2020). Fine-tuning, on the other hand, involves making smaller, targeted adjustments to improve the model's performance on specific tasks without starting the model building from scratch (Ruder, 2018). This regular upkeeping helps maintain the evaluation process accurate and fair, ensuring that the system stays aligned with current realities and emerging trends (Honnibal, 2017; Mehrabi et al., 2021). For example, if the accuracy of ID checks slips because new ID formats were introduced or image quality changed, the YOLO model is retrained with updated examples. This feature helps the system remain efficient at spotting valid and invalid documents.

However, if the model's performance begins to decline, assume, due to changes in ID design or variations in image quality, the system initiates a retraining process. This procedure involves gathering updated samples of the new ID formats or images with different quality levels and then re-labelling these images to serve as fresh training data. By updating the training set, the YOLO model learns to adjust to the new conditions, thereby maintaining its effectiveness in distinguishing between valid and invalid ID's (Redmon *et al*., 2016; Jain & Bolle, 2006).

3.2.10 Maintenance

By systematically tracking system performance and incorporating user and evaluator feedback, the maintenance process not only identifies and remedies issues early but also supports continuous improvement and transparency throughout the application review cycle.

By routinely reviewing the metrics, administrators can pinpoint when the system is not performing as expected and investigate possible reasons behind delays or disparities. This enables them to implement timely improvements, whether that means recalibrating algorithms, adjusting resource allocation, or updating system protocols to better handle current data loads. Overall, continuous monitoring and logging help maintain a transparent evaluation process, ensuring that the HAIPSE framework remains both high-performing and equitable, aligning with best practices in responsible AI deployment (Binns et al., 2018).

Chapter 4: Results

The Chapter presents the results of the experiments conducted using the methodology described in Chapter 3.

4.1 Performance of the YOLOV11 Model in ID Detection

For training and testing of the YOLOV11 Model and its performance evaluation, a sample set of IDs similar to the original applicants' IDs was used (Figure 27).

The operation of an image recognition pipeline for ID detection in Computer Vision module using YOLOV11 model requires specific steps of dataset split, preprocessing, and augmentation. The dataset is divided into three subsets: Training Set (87%) with 60 images, Validation Set (7%) with 5 images, and Test Set (6%) with 4 images. This kind of split is typical in small-scale training environments where the bulk of the data is reserved for model learning, while smaller portions are used to evaluate performance during and after training (Sammut, 2011). The validation set helps fine-tune model parameters without overfitting (Kuhn & Johnson, 2013), and the test set is held back for final performance assessment (Dietterich, 1998).

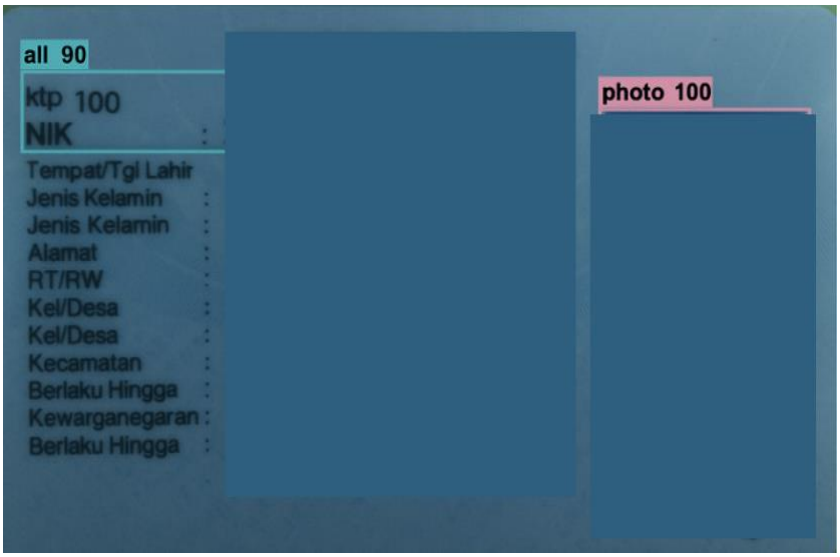


Figure 27. Sample ID for YOLOV11 Model Image Recognition Pipeline.

In the preprocessing phase, two techniques are applied. First, Auto-Orient is used to correct any orientation issues in the input images, this is particularly useful for scanned documents or ID photos that may be upside down or sideways (Krizhevsky et al., 2012). Next, all images are resized to 640x640 pixels, a common resolution in object detection tasks that balances clarity and computational efficiency. Standardizing image size ensures that the YOLO-based model receives consistent input dimensions, which is critical for maintaining detection accuracy (Jocher et al., 2020).

To improve the robustness and generalization of the model, a set of augmentation techniques is applied, such as Grayscale, Saturation, Exposure and Brightness adjustment. Specifically, Grayscale involves randomly turning 15% of images into shades of grey and removing all colour information. This forces the model to focus on shapes, edges, and texture, making it robust to scan photos that come in black-and-white or with strong colour casts (Greene, 2025).

Saturation technique requires to multiply an image's colour intensity by a random factor between 0.75 and 1.25 simulating the documents that appear overly vivid or washed-out (Gallagher, 2023). This "teaches" the network not to rely on exact hues and to detect features even when colour richness varies. Brightness adjustment scales pixel values by a random factor in the range [0.85, 1.15] to ensure that the model can still detect the text and boundaries when overall lighting fluctuates within the range (Greene, 2025). Exposure technique applies ± 10 % compression of the dark and light image pixels simulating low-and high-contrast scans. This processing step ensures that the computer vision model can pick out ID card elements even when the original image contrast is poor or excessively sharp (Buslaev, 2020).

The validation performance of the YOLOV11 model in processing the ID cards from the sample set is assessed by the standard ML metrics of precision, recall and mean average precision as well as the following specific metrics or distinct "classes", each related to a specific region on an ID card: *ktp* representing the entire card boundary, *photo* meaning the applicant's portrait area, *nik* specifying the National Identity Number (NIK) field, *nik_start_point* corresponding to the top-left anchor of the NIK field (helping downstream text extraction), and *all* which is an aggregate measure combining model performance on all four specific classes. These classes are chosen

because detecting both broad, visually obvious regions (ktp, photo) and finer text regions (nik, nik_start_point) is essential for a complete ID-verification pipeline. The five specific metrics measure how accurately the model detects and localizes objects, such as text fields or photos, within an ID document, where a higher score closer to 100% indicates more precise image recognition.

The key performance metrics are calculated according to Formulas (7), (8), (9) and (10). In application to image recognition, precision measures the fraction of bounding boxes that are correctly detected (true positives) out of all boxes the model predicted as positive:

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}}, \quad (7)$$

where

True Positive (TP) is an instance for which the model predicts the positive class, and the ground-truth label is also positive. In object detection, a TP is a predicted bounding box whose intersection-over-union (IoU) with a ground-truth box is ≥ the chosen threshold (e.g., 0.5).

False Positive (FP) is an instance that the model classifies as positive even though its ground-truth label is negative. In object detection, an FP is a predicted bounding box that does not match any ground-truth box at or above the IoU threshold.

Recall measures the fraction of ground-truth objects that the model successfully detected out of all ground-truth instances:

$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}}, \quad (8)$$

Where

False Negative (FN) is an instance whose true label is positive but the model predicts it as negative. In object detection, it corresponds to the ground-truth boxes for which the model produced no detection above the IoU threshold.

Average Precision (AP) for a Class summarizes the Precision-Recall curve for a single class by computing the area under that curve. In practice, we approximate the integral via a discrete sum over *G* recall levels:

$$AP = \int_0^1 P(r)dr \approx \sum_{k=1}^{G} P_k \cdot \Delta R_k, \quad (9)$$

where

P_k is the precision at the *k*-th confidence threshold,

$\Delta R_k = R_k - R_{k-1}$ is the change in recall between thresholds *k*-1 and *k*.

Recall values R_0, R_1,.....,R_G range from 0 to 1.

A common implementation of AP uses the set of all distinct recall levels achieved by the algorithm and totals the products $P_k \times (R_k - R_{k-1})$, $k = 1,..., G$.

Mean Average Precision (mAP) averages individual per-class AP_c over all *C* object categories:

$$mAP = \frac{1}{C} \cdot \sum_{c=1}^{C} AP_c \ . \quad (10)$$

For mAP@50, each AP_c is computed at a fixed IoU threshold of 0.50.

For mAP@[0.50:0.95], each AP_c is computed at multiple IoU thresholds $t \in \{0.50, 0.55,.....,0.95\}$, then averaged over all thresholds and classes:

$$mAP_{[0.50:0.95]} = \frac{1}{T \times C} \cdot \sum_{t=1}^{T} \sum_{c=1}^{C} AP_c^{IoU_t} \ .$$

The YOLOV11 model was trained, tested and validated using Roboflow (https://roboflow.com), an end-to-end computer-vision platform, and the performance results for each class are shown in Figure 28.

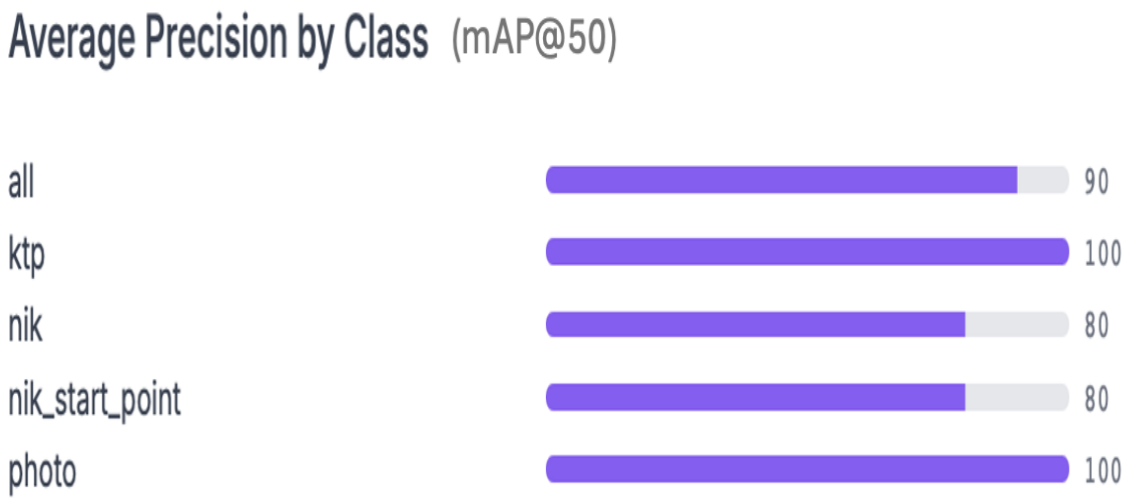


Figure 28. Average Precision Measures by Class.

Overall, these validation metrics indicate that the model performs exceptionally well in detecting broad, visually distinct features, such as the ID card and photo, while maintaining acceptable precision for more specific textual components. Continued fine-tuning or expanding the training dataset with more annotated examples of the NIK region could help close the remaining performance gap (Shrestha et al., 2021).

Figure 29 presents a comprehensive set of training and validation graphs that illustrate the performance of the ID-detection model over 140 epochs. An epoch denotes a single complete pass of the YOLOv11 network over the entire training corpus. Consequently, training for 140 epochs entails 140 sequential traversals of the ID-image dataset, with model parameters updated after every mini-batch gradient step. We determine how many epochs to run by monitoring the training and validation curves: if both continue to improve, training continues; if the validation curve declines, we stop to avoid overfitting. It allow us to observe how loss and accuracy evolve as the model learns from successive iterations. Section (A) in Figure 29 shows metrics related to the training set, while section (B) reflects validation performance. The specific performance metrics are explained in Table 6.

The training losses metrics, train/box_loss , train/cls_loss in section (A) demonstrate a consistent downward trend, indicating that the model is successfully learning to localize objects

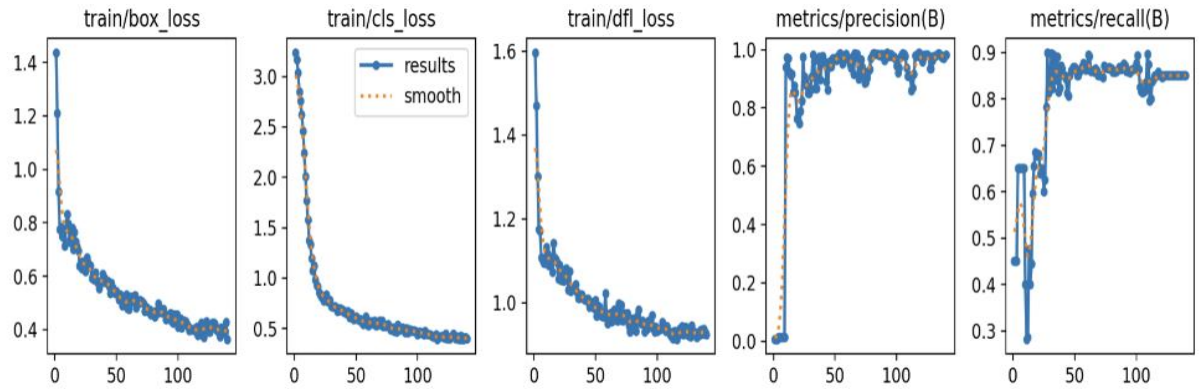
(Li et al., 2020), classify them, and refine bounding box predictions. For example, the train/box_loss and train/cls_loss metrics both start relatively high and decrease steadily as training progresses, suggesting that the model improves its accuracy in both detecting and categorizing objects in ID images.

Looking at the validation results in section (B), we see more variable patterns, particularly in the val/box_loss and val/dfl_loss metrics, which show noticeable fluctuations. This variability is common, especially when working with small validation datasets, as even minor differences in the input set can significantly affect the computed loss. Nonetheless, both val/cls_loss and val/box_loss metrics show an overall decreasing trend, which supports the conclusion that the model is generalizing reasonably well and not overfitted by the training data (Goodfellow et al., 2016). The presence of some spikes suggests potential areas for further optimization, such as data augmentation, learning rate adjustment, or expanding the validation dataset to better stabilize the performance (Li et al., 2020).

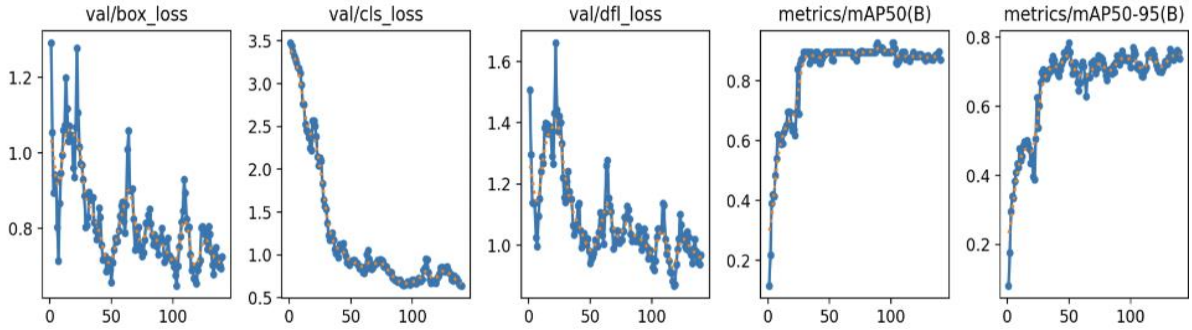
Table 6. YOLOV11 Key Performance Metrics and their Descriptions.

Performance Metric	Description
train/box_loss	Training-set bounding-box regression loss; measures how far the predicted boxes deviate from ground-truth boxes
train/cls_loss	Training-set classification loss; quantifies mis-classifications of detected objects
train/dfl_loss	Training-set Distribution Focal Loss; a refined localization loss that improves bounding-box quality, especially near edges
metrics/precision (B)	Precision on the current training/validation batch B; proportion of predicted boxes that are true positives
metrics/recall (B)	Recall on batch B; proportion of ground-truth objects that were correctly detected
val/box_loss	Validation-set bounding-box regression loss; mirrors train/box_loss but on unseen data to gauge generalization
val/cls_loss	Validation-set classification loss; mirrors train/cls_loss on unseen data
val/dfl_loss	Validation-set Distribution Focal Loss; mirrors train/dfl_loss on unseen data
metrics/mAP50 (B)	Mean Average Precision at IoU 0.50 on batch B; harmonic summary of precision-recall across all classes when predicted boxes overlap ground truth by ≥ 50 %
metrics/mAP50-95 (B)	Mean Average Precision averaged over IoU on batch B; stricter, more comprehensive gauge of detection quality

The metrics/precision(B) and metrics/recall(B) graphs demonstrate that the model quickly learns to correctly identify ID regions with high accuracy, stabilizing above 0.85 and 0.9, respectively. Most notably, the metrics/mAP50(B) score rises sharply early in training experiment and stabilizes near 0.9, which is a strong indicator that the model is reliably predicting bounding boxes with at least 50% overlap with ground truth ID samples (Everingham et al., 2010; Padilla et al., 2020). Additionally, the metrics/mAP50-95(B), a stricter and more nuanced performance metric, improves steadily and approaches 0.75 by the end of training exercise. Together, these results confirm that the model has effectively learned the key tasks for ID detection identifying document elements with precision and consistency and is ready for deployment on HAIPSE platform.



(A)



(B)

Figure 29. Training (A) and Validation (B) Graphs for ID Detection.

The graph in Figure 30 provides a visual summary of the training performance of an ID detection model. Section (A) shows the mean Average Precision (mAP) curves, which measure the

model's ability to correctly detect and classify objects. The dark purple line represents mAP at an IoU threshold of 0.5 meaning that a detection is counted as correct only if the predicted box overlaps the true box by at least 50%, ensuring that the model not only recognizes objects but also localizes them with sufficient accuracy, which rises quickly and stabilizes around 0.9, indicating that the model consistently detects ID components accurately (Everingham et al., 2010). The lighter purple line, representing mAP@50:95, having average precision over multiple IoU thresholds ranging from 0.50 to 0.95 (in increments of 0.05). As a stricter metric, it climbs more gradually and levels out near 0.75, demonstrating that the model maintains solid precision even under tighter bounding-box matching criteria.

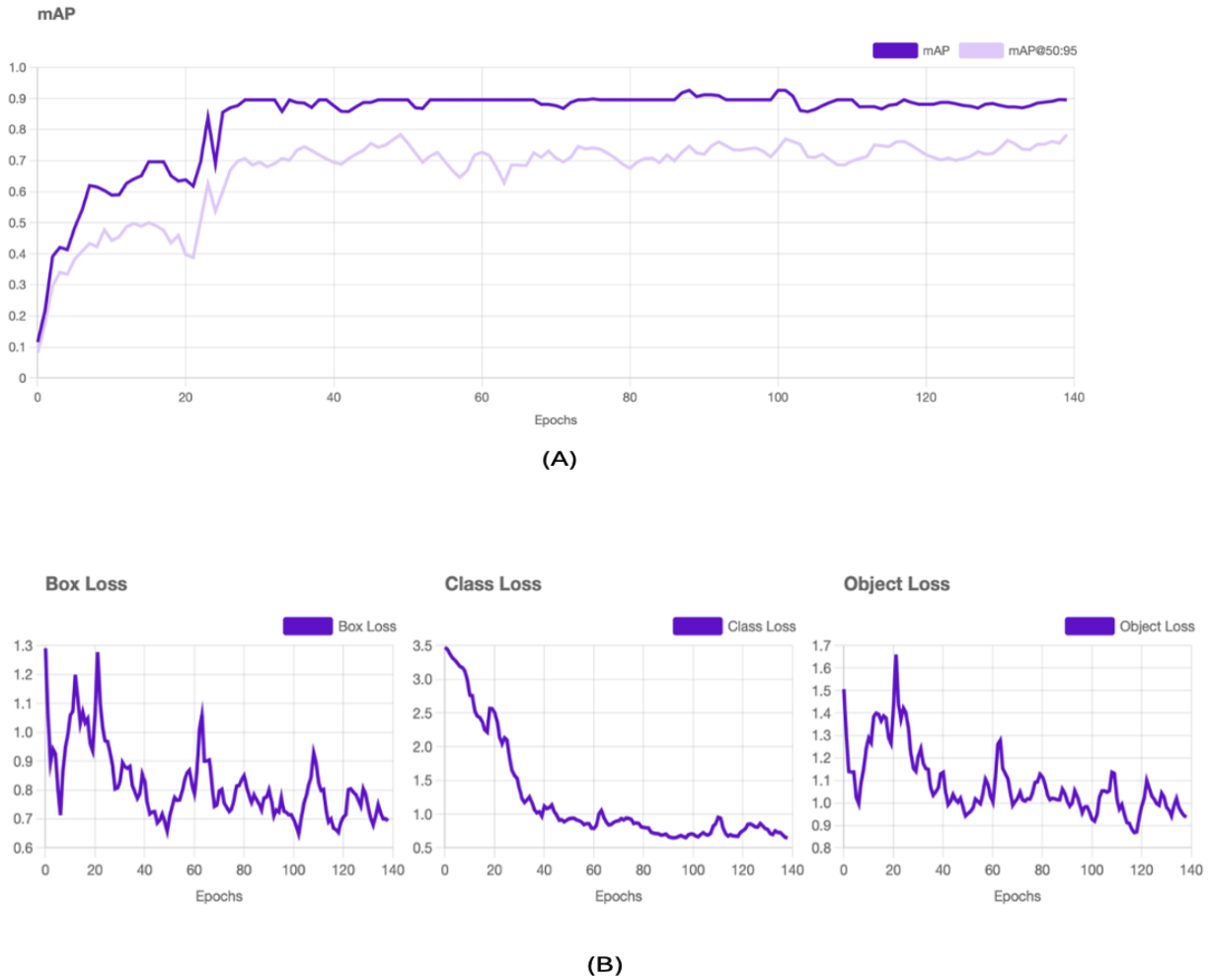


Figure 30. (A) mAP and mAP@50:95 progression improved model accuracy over training epochs. (B) Box Loss, Class Loss, and Object losses showing effective convergence of the YOLOV11 model during training.

Section (B) reflects different loss components: box loss, class loss, and object. These metrics measure how far off the model's predictions are during training (Redmon & Farhadi, 2018). The box loss, which relates to bounding box accuracy, decreases steadily and stabilizes below 0.7, indicating improving localization of ID features. The class loss begins high, above 3.0, but drops significantly below 1.0, showing that the model is learning to correctly classify parts of the ID components, such as photos, NIK numbers, etc. (Zhang et al., 2021). The object loss, which measures the model's confidence in detecting object vs. its background, also trends downward despite some fluctuations. Together, the high mAP scores and declining loss values confirm that the model is learning effectively, becoming both accurate and confident in detecting various ID components.

4.2 Cohort Fairness Score

The Cohort Fairness Score derived from the application heuristic score is a numerical metric designed to measure the variance of application scores within a given batch (or cohort), which is explained in detail in Section 3.2.6 and computed using Formula (3) (Hardt et al., 2016). A lower score value indicates higher fairness, meaning that most applications in the cohort were evaluated with similar criteria, and their scores clustered closely around the mean. A higher score value suggests greater disparity, possibly indicating inconsistency in evaluation standards or inherent bias in the scoring algorithm (Kleinberg et al., 2017).

Cohort Fairness Scores of four application cohorts are shown in Figure 31. For instance, Cohort 1, though having the mean score of 34.00 demonstrates the fairness score of 24.56, which is quite high. This result suggests that, while some applications may have scored very low, others scored very high. This inconsistency can be explained by the unequal treatment of certain applications or scoring criteria not being applied uniformly (Corbett-Davies et al., 2017).

In contrast**,** Cohort 3 has a similar mean score of 38.72, but a fairness score of 1.43, indicating that most applicants received similar scores. This outcome consistency reflects a fairer and more balanced evaluation process, with less risk of systemic bias (Verma & Rubin, 2018). Maintaining a low Cohort Fairness Score is crucial in systems like HAIPSE because it ensures equity, particularly when AI models are involved in scoring evaluation of human applications.

High Fairness Scores can signal areas where the system might be failing to account for demographic or contextual fairness, which may require human audit or model adjustment. It also enhances transparency and trust, as decision-makers can clearly identify which Cohorts were scored fairly and which may require further evaluation (Friedler et al., 2019).

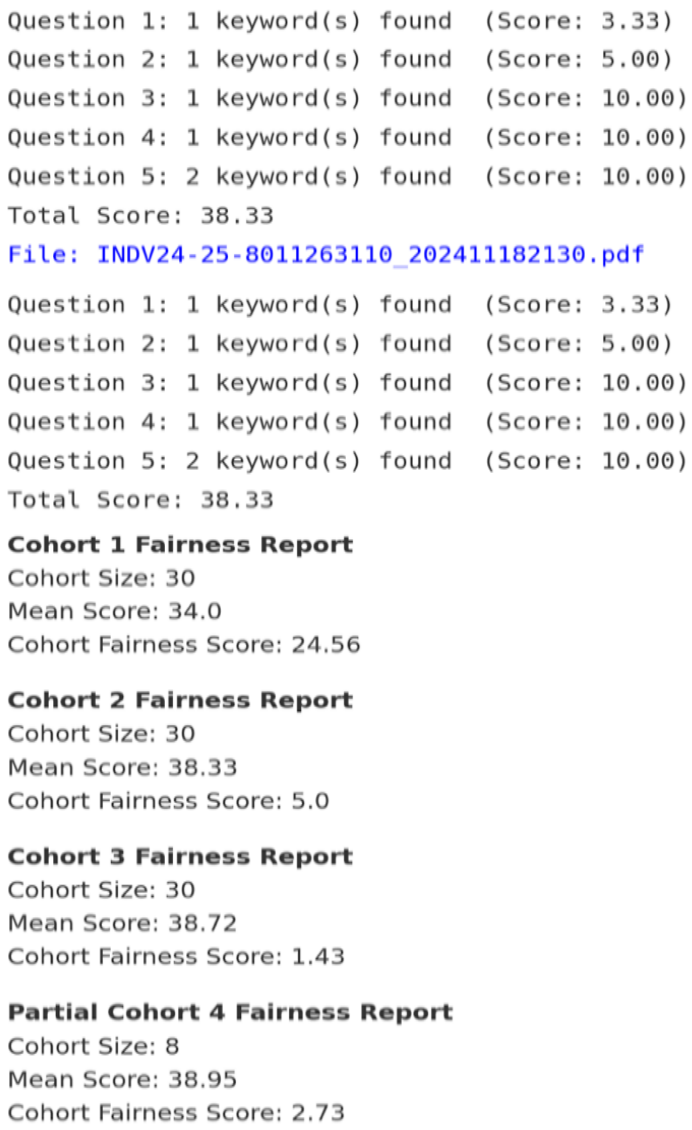


Figure 31. Heuristic Scoring and Cohort Fairness Score of Applications.

The dynamics of the mean values of Cohort Fairness scores across application batches over time are illustrated in Figure 32. Each point on the line represents the mean fairness score for a

specific cohort number which is calculated by averaging each application's individual fairness metric indicating how application evaluations trended throughout the processing cycle. Initially, in Cohort 1, the average score is low, but it gradually increases, peaking at around 39 in Cohort 4, and then stabilizes with minimal variation through Cohorts 5 to 7.

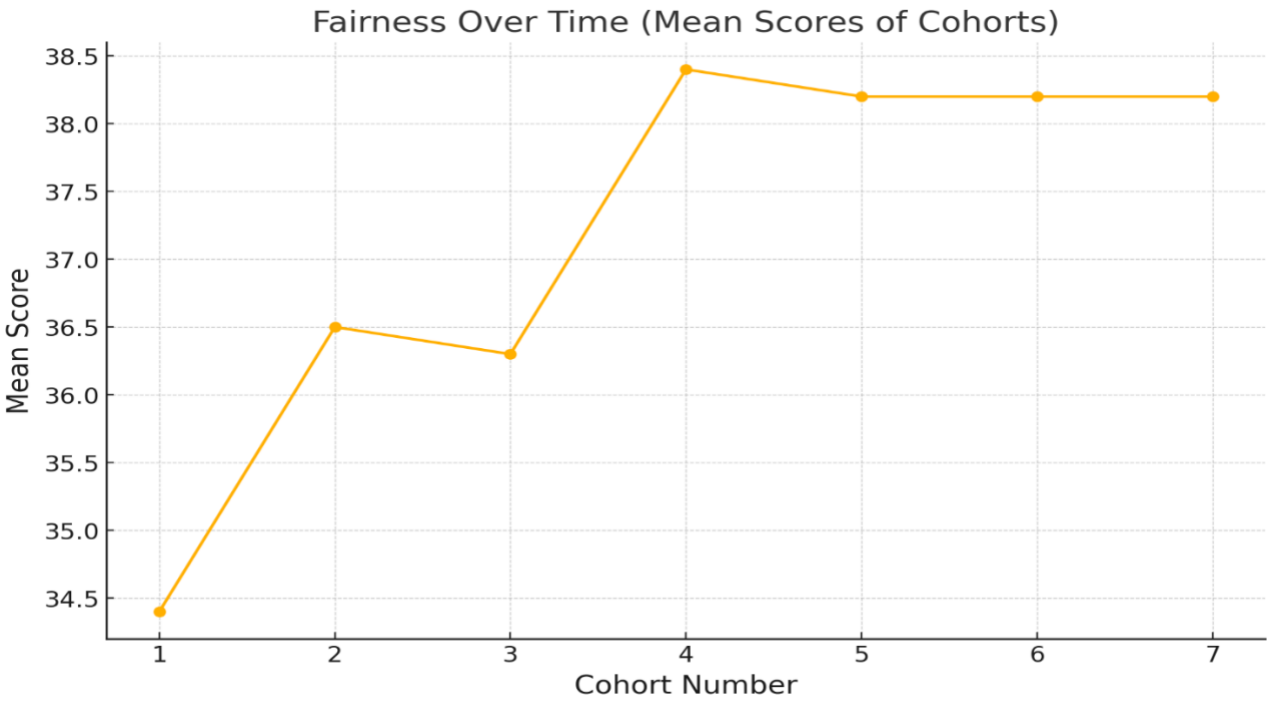


Figure 32. Mean value of Cohort Fairness Score across application Cohorts Over Time.

This result suggests that the system's scoring logic or model understanding improved over time, possibly due to enhanced keyword detection, refined heuristics, or feedback-driven calibration (Sculley et al., 2015). This trend implies that later Cohorts were likely scoring under more calibrated conditions, potentially giving them a fairer advantage in terms of evaluation consistency (Friedler et al., 2019).

The variance of fairness scores within each Cohort is shown in Figure 33. High bars represent higher variance and thus lower fairness, while lower bars indicate that applications within a cohort received more consistent scores (Corbett-Davies et al., 2017). Cohort 1, with the tallest bar, has the highest variance (over 20), indicating that applications were scored with greater

inconsistency possibly due to less stable model behaviour or insufficient tuning early in the evaluation process (Mitchell et al., 2019). In contrast, later Cohorts, like 4, 5, and 7, show significantly lower fairness scores (under 5), highlighting more uniformed treatment of applications. This pattern confirms that fairness improved as the model matured (Raji et al., 2020). It means that the evaluated applications in low-variance Cohorts were assessed under more equitable conditions, reinforcing confidence in the scoring framework (Barocas & Selbst, 2016).

Figure 33. The Variance of Fairness Scores within each Cohort.

4.3 Large Language Models Summarization

In the context of automated summarization, Large Language Models (LLMs) play a vital role in transforming lengthy and unstructured text into concise, coherent, and informative summaries (Liu

& Lapata, 2019). For this experiment, several state-of-the-art LLMs were tested to assess their performance in summarizing responses for a given task (Fabbri et al., 2021). The goal was to determine how well each model could capture the core meaning of input text while balancing brevity, clarity, accuracy, and coverage (Maynez et al., 2020). The evaluation exercise included a variety of foundational LLMs like Mixtral, Smol, Databrick, Falcon, Llama, Phi, and Qwen, each known for distinct architectural features and strengths (Raffel et al., 2020).

These models were benchmarked using four key criteria: *brevity*, which measures how concisely the model summarizes information, *clarity*, assessing the readability and ease of understanding**,** *faithfulness*, reflecting the model's accuracy in retaining factual information, and *coverage*, evaluating the completeness of key ideas included in the summary (Bhandari et al., 2020). Through this comparison, the strengths and limitations of each LLM were analysed to identify the most suitable options for domain-specific summarization tasks, such as those involved in evaluating applications, reports, or complex documents (Zhang et al., 2023).

Evaluation of various LLMs based on their summarization performance across four criteria is demonstrated in Table 7. Among the models, Falcon and both versions of Llama LLMs consistently stand out, scoring the highest in clarity, faithfulness, and coverage, with only slight variations in brevity. These models produce highly detailed and accurate summaries, making them well-suited for in-depth or comprehensive tasks. Phi and Qwen LLMs also perform well across all criteria, delivering clear, balanced summaries with minor drawbacks while Qwen may miss details or become repetitive.

In contrast, Mixtral and Smol LLMs represent more concise summarization styles, with Mixtral achieving the highest brevity score, though it tends to oversimplify the content. Smol strikes a better balance between clarity and faithfulness. Databrick, however, underperforms across all categories, particularly in coverage and clarity, due to its tendency to produce incomplete or inconsistent outputs. These evaluations indicate that, while some models excel in concise summarization, others are more effective for generating nuanced and faithful content. The selection of a model should thus align with the specific needs, whether they are being depth, speed, brevity, or faithfulness (Kryscinsji et al., 2020).

Table 7. Evaluation of Summarization Performance of Different LLMs.

Criteria	Mixtral	Smol	Databrick	Falcon	Llama	Llama-8B	Phi	Qwen
Brevity Score (1-5)	5	4	2	3	3	4	4	3
Clarity Score (1-5)	4	4	2	5	5	5	4	4
Faithfulness Score (1-5)	3	4	2	5	5	5	4	4
Coverage Score (1-5)	3	4	1	5	5	4	4	4
Comment	Excels in brevity But often oversimplifies.	Balanced Summaries with good clarity but faithfulness	Incomplete summaries, making it unsuitable overall.	Produces detailed and nuanced summaries	Highly detailed and insightful, suitable for comprehensive tasks but verbose for short needs.	Balanced and faithful summaries with clarity, but not ideal for extreme brevity.	Clear and concise summaries with balanced depth but occasionally Formulaic.	Balanced summaries but suffers from occasional missing outputs and repetitive structure.

A qualitative evaluation of various LLMs based on their strengths, weaknesses, and recommended use cases for summarization tasks is shown in Table 8. Each model is assessed to help determine when and how it performs best in generating summaries, which is especially valuable for selecting the most appropriate model in different contexts, mainly ranging from brief overviews to rich, in-depth analyses (Goyal et al., 2022).

Models, like Mixtral-8x7B and Phi, stand out for their brevity and clarity. Mixtral is best suited for high-level overviews due to its conciseness, although it tends to oversimplify and miss key details**.** SmolLM2-1.7B provides balanced summaries with moderate detail, but may be too wordy for tasks that demand extremely short outputs. In contrast, Databricks' dolly-v2-3b underperforms due to incomplete and inconsistent summaries, making it unreliable and not recommended for critical tasks.

Table 8. Qualitative Evaluation of Various LLMs in Summarization.

Model	Strengths	Weaknesses	Best Use Case
Mixtral-8x7B-Instruct-v0.1	Highly concise, good for brevity	Oversimplifies, misses critical details	High-level overviews
SmolLM2-1.7B-Instruct	Balanced summaries, moderate detail	Slightly verbose for extremely concise needs	Moderate detail tasks
Databricks' dolly-v2-3b	Concise but inconsistent, many missing summaries	Incomplete outputs, unreliable	Not recommended
tiiuae/falcon-7b-instruct	Detailed and nuanced	Slightly verbose for short summaries	Detailed summarization
tiiuae/falcon-7b-instruct (Max-token=250)	Highly detailed, insightful	Verbose for conciseness-required tasks	Tasks needing rich, comprehensive insights
Meta-Llama-3-8B-Instruct	Balanced, faithful, and clear	Not ideal for extreme brevity	General-purpose summarization
Phi (200-token output)	Clear and concise with balanced depth	Occasionally formulaic, repetitive	Moderate detail with consistency
Qwan (250-token output)	Balanced with good depth and clarity	Some missing outputs, structural repetition	Detailed summaries if all outputs are present

More robust performers include Meta-Llama-3-8B, Tiiuae Falcon-7B, and Qwen, which are praised for their clarity, detail, and balance. Meta-Llama-3-8B is ideal for general-purpose summarization, offering faithful and readable summaries, although it may not be suited for extremely concise outputs. Both versions of the Tiiuae Falcon-7B model (i.e., standard and Max-token) produce highly detailed, nuanced summaries, which are ideal for tasks that require deep comprehension, though they may be perfect for quick overviews (Paulus et al., 2018). Qwen and Phi also perform well in maintaining balanced clarity and depth, though Qwen occasionally misses outputs and may become repetitive (Cei et al., 2023). Overall, Table 7 offers a practical guide for selecting LLMs based on task-specific needs and output quality preferences.

The Bar chart in Figure 34 provides a visual comparison of different LLMs based on their summarization performance across the same four evaluation criteria: brevity, clarity, faithfulness, and coverage. Each model's score for each criterion is evaluated on a scale from 1 to 5. Models, like Meta-Llama-3-8B and Tiiuae Falcon-7B, stand out with perfect scores of 5 across three criteria (clarity, faithfulness and coverage), indicating that they generate clear, accurate and complete summaries. These models are particularly well-suited for tasks where high-quality

summarization is essential (Celikyilmaz et al., 2020). On the other hand, models, such as Databricks (v2-3b), scored the lowest, particularly in coverage, which suggests that they often produce incomplete or inaccurate summaries, making them less reliable for critical summarization tasks (Paulus et al., 2018).

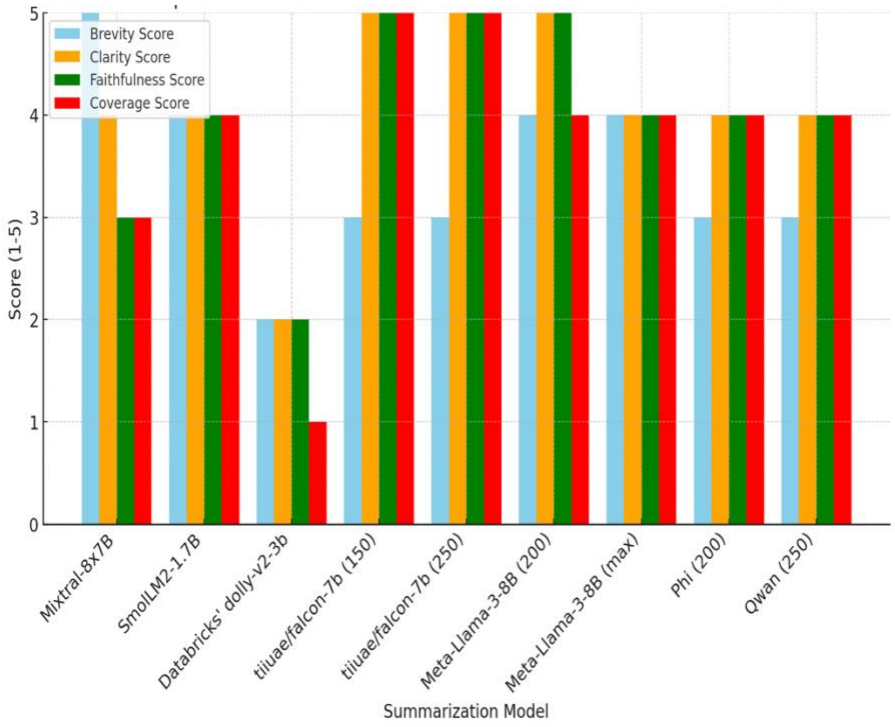


Figure 34. Comparison of LLMs Based on Evaluation Criteria.

Overall, Figure 34 helps visualize how different models trade-off between conciseness and detail, with the best-performing models striking a strong balance across all dimensions. These results are particularly useful for selecting the most appropriate model depending on whether the summarization task prioritizes compactness, depth, or factual accuracy. The dashboard to assess score disparities in AI-generated summaries is visualized in Figure 35. It presents a preview of text responses, calculates summary score statistics, and visualizes score deviations from the mean value to detect potential bias. It can be seen that the mean summary score is 2.56, while the score variance (bias) is 10.53, which is relatively high for a small dataset suggesting significant inconsistency in how the LLM scored (Raji et al., 2020). In this bar chart visualization green bars indicate positive deviations (scores are above the mean), while red bars indicate negative deviations (scores are below the mean). The size and distribution of the bars show that, while a

few LLMs scored well above average, many scored significantly lower, indicating inconsistency or possible scoring bias (Hoistein et al., 2019).

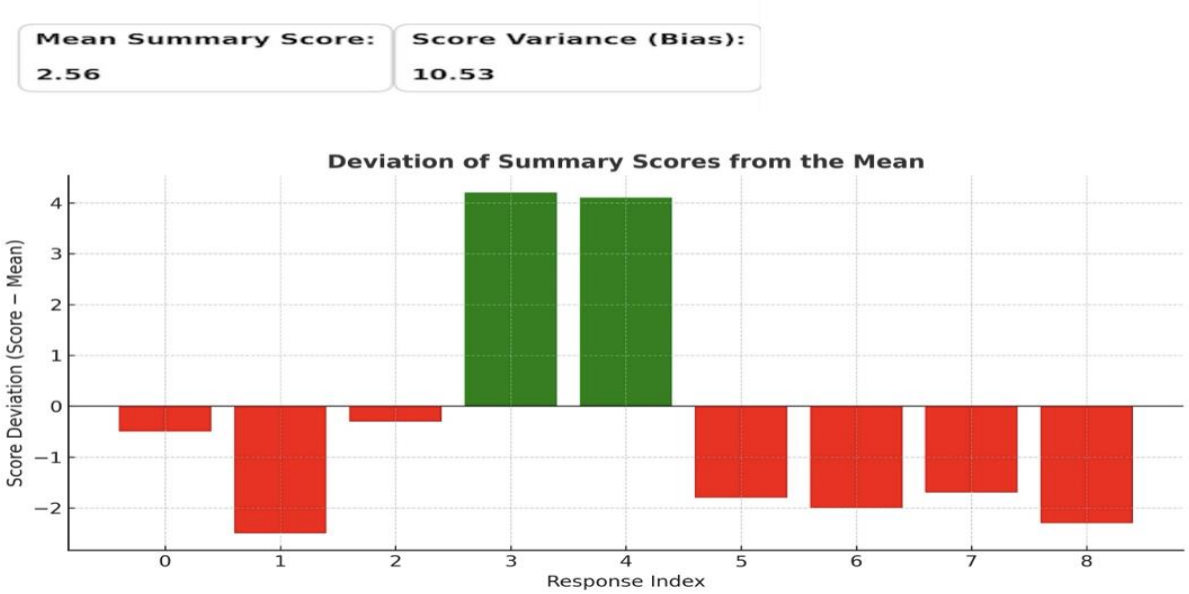


Figure 35. Deviation of LLM Generated Summary Scores From the Mean Value (Bias Detection).

Integrating this kind of bias analysis into the HAIPSE framework offers several benefits. First, it promotes transparency and fairness by identifying whether certain responses are consistently over- or under-valued by the scoring model. This allows administrators to take corrective action, such as reviewing outliers or refining model parameters. Second, it strengthens the credibility of the evaluation system, ensuring that automated scores are not only efficient but also equitable (Kearns & Roth, 2020). Finally, it supports the HAIPSE framework's commitment to responsible AI by embedding regular checks for bias and enabling continuous improvement in how applications are assessed (Corbett-Davies et al., 2017). By flagging anomalies early, the system becomes more reliable and better aligned with principles of fairness and inclusivity.

4.4 HAIPSE Implementation

The User Interface Module, NLP Module, Expert System Component, Cohort Fairness Score Module, and Human Feedback Module have been developed, and tested in Visual Studio Code

software (https://code.visualstudio.com), while Computer-Vision Module was trained, validated and tested on the Roboflow platform and LLM Summarization Module.

Figure 36 shows a screenshot of the code implementation to detect bias in LLM-generated summary. It processes the short summary essays generated by the LLM and turns them into easy-to-read numerical values. Initially, the program gets a small list of keywords (e.g., *ambition*, *community*, and *First Nation*). Whenever one of these words appears in an essay, the algorithm counts it. The final count becomes a simple "relevance" score for the essay: the more key words it contains, the higher the score. After scoring every essay, the algorithm totals the results of each keyword appearance in the summary. It signals how often each keyword shows up, what the average score is, and whether some essays are very different from the rest by having either very high or very low total word count.

Figure 36. A Screenshot of the Code to detect Bias in an LLM Generated Summary.

All of this information is displayed in a small web page that loads instantly on any laptop. Reviewers can glance at a bar chart of the word counts and a few headline numbers and immediately judge whether the summaries are balanced and fair.

The Cohort-Fairness Module is designed to track how consistently the scoring rules are applied across batches of applications. The script implementing internal processing logic of the module is shown in Figure 37. Initially, it launches a lightweight Dash web app and sets a fixed cohort size of 30 submissions. A small dictionary of key phrases (e,g,, *"full name," "educational goal," "First Nations"*) is stored for each application question, so that every file is evaluated against the same checklist. When a PDF document is uploaded, the script converts it to plain text, counts how many of the relevant phrases appear in each answer, and assigns a numeric score to every question.

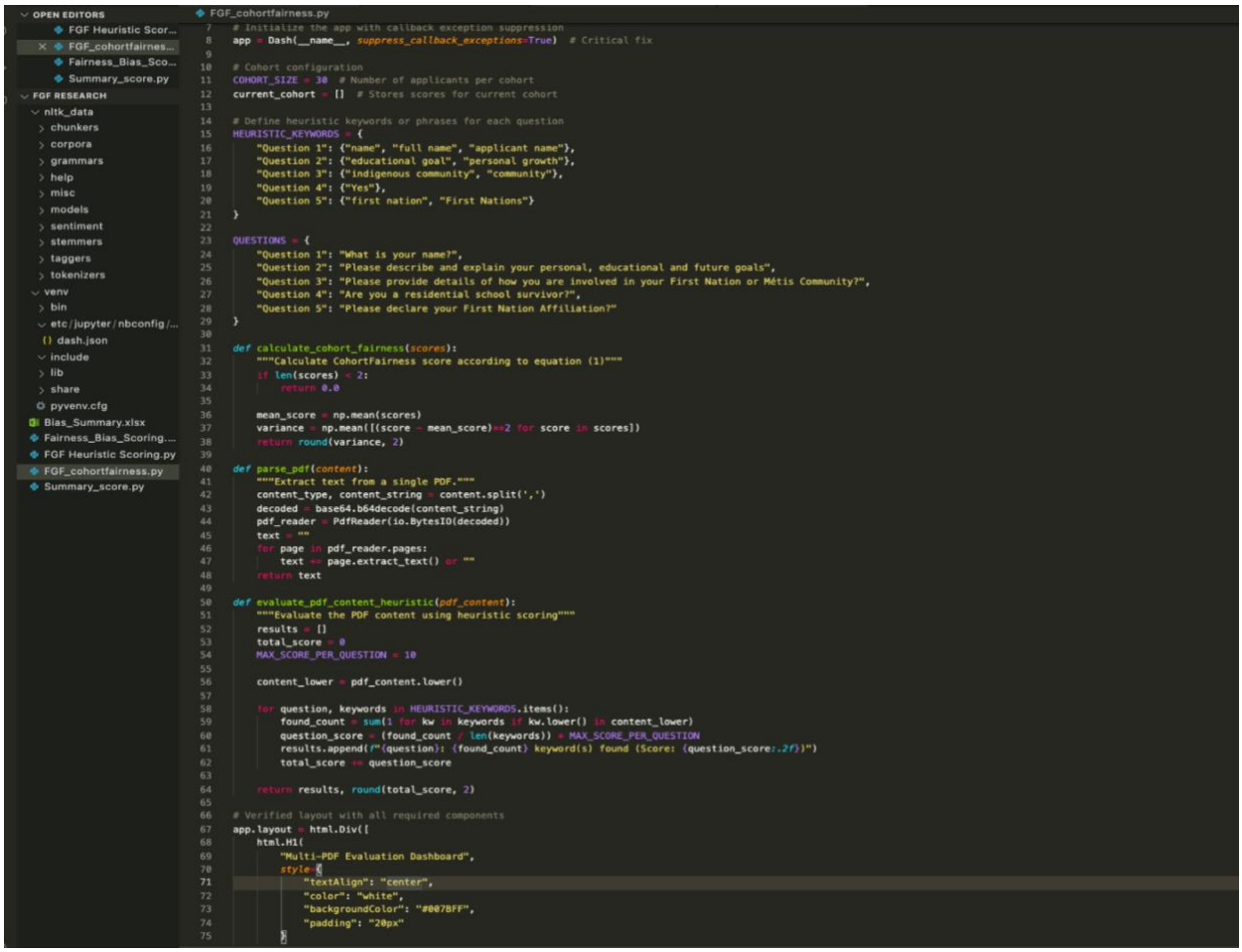
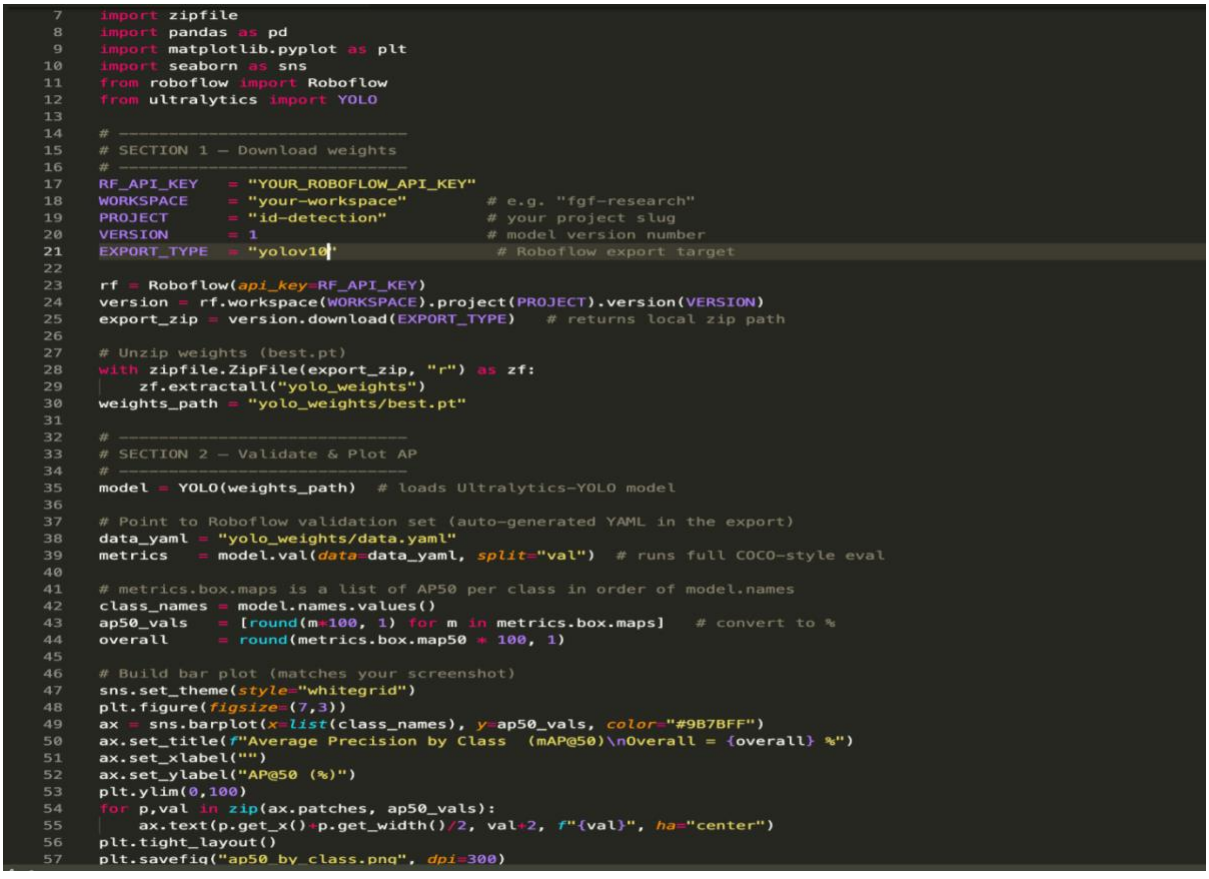


Figure 37. A Screenshot of the Code implementing the Cohort Fairness Score.

Once all 30 applications in a cohort have been processed, the program calculates the variance of those scores. A low variance means the scoring rubric is being applied evenly; a high variance signals possible bias or inconsistency. The Dash library interface (Plotly Dash, 2025) then

displays these results in real time, along with a simple dashboard that lets reviewers see keyword counts, individual scores, and the overall fairness metric for the current cohort. This approach keeps the analysis transparent and easy to interpret while requiring only modest computing resources.

The code implementation of the end-to-end evaluation pipeline for a Roboflow-trained YOLOV11 Model is shown in Figure 38. The execution starts with configuring the Roboflow SDK environment (https://pypi.org/project/roboflow/) according to the user's API key, workspace, project slug, and software version. The set of IDs is then loaded into Ultralytics' YOLO class, which executes a COCO (Common Objects in Context)-style validation routine to compute both per-class and overall mAP@0.50 metrics. Finally, the script employs Seaborn and Matplotlib libraries (Waskom, 2024; Hunter et al., 2025) to render a clean, publication-ready bar chart of each class's AP50 (annotated with exact percentage and styled to match the Roboflow dashboard).



```python
import zipfile
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from roboflow import Roboflow
from ultralytics import YOLO

# ----------------------------------
# SECTION 1 — Download weights
# ----------------------------------
RF_API_KEY    = "YOUR_ROBOFLOW_API_KEY"
WORKSPACE     = "your-workspace"          # e.g. "fgf-research"
PROJECT       = "id-detection"            # your project slug
VERSION       = 1                         # model version number
EXPORT_TYPE   = "yolov10"                 # Roboflow export target

rf = Roboflow(api_key=RF_API_KEY)
version = rf.workspace(WORKSPACE).project(PROJECT).version(VERSION)
export_zip = version.download(EXPORT_TYPE)   # returns local zip path

# Unzip weights (best.pt)
with zipfile.ZipFile(export_zip, "r") as zf:
    zf.extractall("yolo_weights")
weights_path = "yolo_weights/best.pt"

# ----------------------------------
# SECTION 2 — Validate & Plot AP
# ----------------------------------
model = YOLO(weights_path)  # loads Ultralytics-YOLO model

# Point to Roboflow validation set (auto-generated YAML in the export)
data_yaml = "yolo_weights/data.yaml"
metrics   = model.val(data=data_yaml, split="val")  # runs full COCO-style eval

# metrics.box.maps is a list of AP50 per class in order of model.names
class_names = model.names.values()
ap50_vals   = [round(m*100, 1) for m in metrics.box.maps]   # convert to %
overall     = round(metrics.box.map50 * 100, 1)

# Build bar plot (matches your screenshot)
sns.set_theme(style="whitegrid")
plt.figure(figsize=(7,3))
ax = sns.barplot(x=list(class_names), y=ap50_vals, color="#9B7BFF")
ax.set_title(f"Average Precision by Class  (mAP@50)\nOverall = {overall} %")
ax.set_xlabel("")
ax.set_ylabel("AP@50 (%)")
plt.ylim(0,100)
for p,val in zip(ax.patches, ap50_vals):
    ax.text(p.get_x()+p.get_width()/2, val+2, f"{val}", ha="center")
plt.tight_layout()
plt.savefig("ap50_by_class.png", dpi=300)
```

Figure 38. A Screenshot of the Code to Execute an End-to-End Evaluation Pipeline for a Roboflow-Trained YOLOV11 Model.

4.5 Integration of HAIPSE with the NIB Trust Portal

The integration of the HAIPSE with the NIB Trust Portal is aimed to minimize the human factor in the application processing, so that every time an applicant submits an application or updates their materials, those changes are immediately picked up without any manual intervention. To this end, the Apply Connect feature is enabled in SurveyMonkey cloud-based platform which allows to interconnect external components with other business applications, automate tasks and streamline workflows. As a result, a secure Client ID and Client Secret port are generated. The HAIPSE uses these credentials to request an access token, i.e., a temporary "key" that authorizes an external client to safely connect to SurveyMonkey API and confirms it has permission to retrieve application data. Figure 39 demonstrates Survey Monkey Apply Integrations dashboard within the NIB Trust Portal, showing the available connectors.

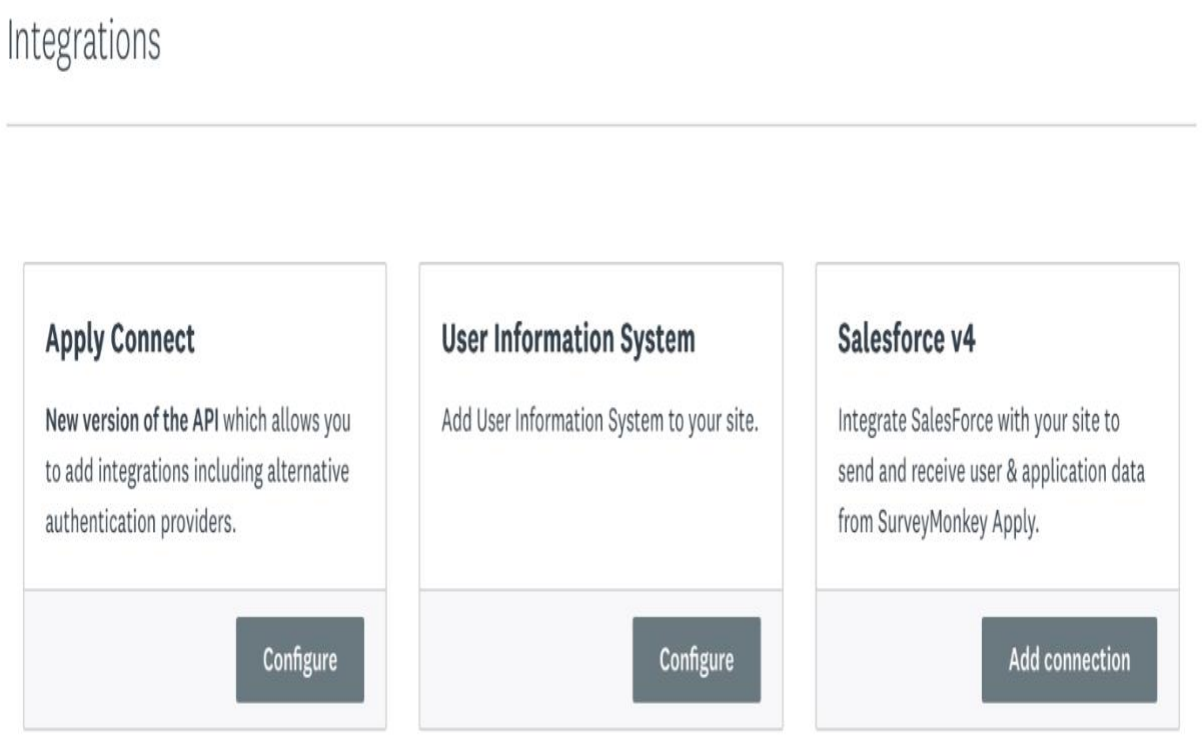


Figure 39. Survey Monkey Apply Integrations dashboard within the NIB Trust Portal, showing the available connectors.

On the NIB Trust Portal side, a user-defined HTTP callbacks are implemented that allow one application to notify another application about events in real-time instead of constantly polling

for updates, the so-named *webhook calls*. When the HAIPSE receives applications, it immediately extracts the relevant fields, downloads any attached PDF essays, and runs them through the Computer Vision module, NLP Heuristic scorer, Fairness calculator, and LLM summarizer. This way, every new or changed application is processed automatically, and the updated scores and summaries appear instantly on the portal, streamlining the review process and eliminating the need for manual data transfers. Figure 40 shows the principal integration of the HAIPSE framework with the NIB Trust Portal to enable efficient applications' evaluation.

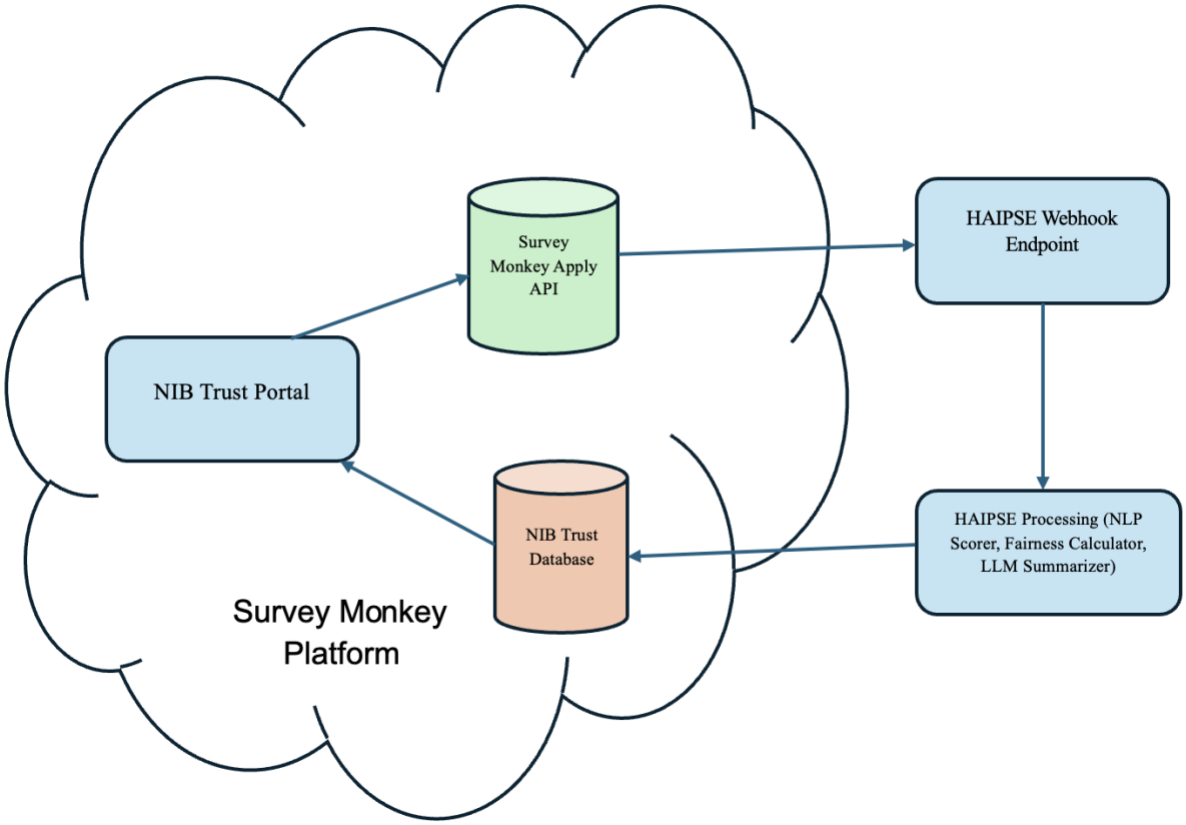


Figure 40. Integration of HAIPSE with the NIB Trust Portal via Survey Monkey API.

Chapter 5: Conclusion and Future Work

The objective of this thesis is to propose a methodology for the HAIPSE framework that merges NLP and expert rule-based systems integrated with generative AI to redefine fairness and bias in the application evaluation process (Suresh & Guttag, 2019). The HAIPSE demonstrates that the right mix of automation and human expertise can transform an often slow, subjective review process into one that is swift and consistent.

At the backend, a computer-vision module inspects every uploaded ID, catching blurred photos or tampered text (Holzinger, 2016). This feature frees staff from painstaking manual checks and ensures that only legitimate applications advance. Simultaneously, a natural-language pipeline powered by LLMs distils multi-page essays into concise summaries, highlighting key skills, experiences, and motivations without losing the actual context (Ribeiro et al., 2016). Reviewers no longer need to go through dense narratives, they just start with a clear snapshot of each application.

Behind this automated tool is a rule-based scoring engine that assigns points for experience and skills. Because the rules are transparent and adjustable, Program admins and officers can see exactly how a score is calculated and adjust weights when program priorities shift (Doshi-Velez & Kim, 2017). More important, humans remain in the loop, so if an applicant's unique background doesn't fit neatly into the scoring rubric, reviewers can override or adjust the score and feed that decision back into the system (Amershi et al., 2019).

Fairness in the HAIPSE is monitored at multiple levels. The platform tracks cohort-level variance, flagging batches in which scores scatter too widely shows an early sign of hidden bias (Wu et al., 2023). It also logs every override event, allowing analysts to spot patterns, such as systematic down-scoring of certain groups. When such patterns emerge, admins can retrain language models, refine keyword or adjust heuristic thresholds. Meanwhile, applicants benefit from faster turnaround times and clearer feedback, since every rejection includes a traceable rationale derived from the scoring engine and any reviewer notes.

In short, the HAIPSE shows that AI does not have to replace human judgment; instead, it can amplify it. By pairing fast, objective automation with informed human oversight, organizations can process large volumes of applications without sacrificing accuracy or fairness. The result is a review pipeline that is not only quicker and more scalable but also more transparent and trustworthy for applicants and evaluators.

Integrating expert rules with NLP transforms free-form application text into precise, machine-readable data and then scores it with clear, domain-specific logic. NLP components, like tokenization, domain-adapted entity recognition, and keyword-ontology mapping, automatically extract critical details, such as years of experience, degrees earned, leadership roles, and even biased or inclusive language (Jurafsky & Martin, 2021). This automation eliminates hours of manual fact-finding and ensures that every application is parsed in a uniformed way, regardless of writing style. The result is a rich, structured profile for each application that captures both quantitative qualifications and qualitative nuances.

Expert rules then convert these extracted features into transparent, adjustable point values. A rule might add ten points for three or more years of relevant experience, award partial credit via fuzzy logic for near-threshold cases (Zadeh, 1965). Because the rule set is explicit and auditable, program administrators can quickly refine thresholds as priorities shift. Together, automatic extraction plus rule-based scoring reduces subjectivity, speeds up processing, and embeds fairness checks directly in the workflow producing evaluations that are quicker, more consistent, and easier to justify to applicants and stakeholders alike (McNamera et al., 2016).

High-quality AI-generated summaries demonstrably help reviewers grasp application content more quickly and consistently, which in turn supports fairer decisions. In our resulting benchmark, the best-performing LLMs (Meta-Llama-3-8B and Tiiuae Falcon-7) which achieved perfect scores (5/5) for brevity, clarity, faithfulness, and coverage (Almazouei et al., 2023). These models condensed multi-page responses into concise yet complete overviews, allowing evaluators to absorb key facts in under two minutes without losing critical details (Goyal et al., 2022). Mid-tier models, such as Phi and Qwen also scored 4/5 across most criteria, offering reliable summaries that balance length and accuracy (Peyrard et al., 2017). Because every evaluator

receives the same structured snapshot, differences in personal reading speed or writing-style preference are minimized, reducing inter-reviewer variance and the potential for subjective bias (Narayan et al., 2018). Moreover, standardized summaries ensure that each applicant is presented through an equivalent lens, making it easier to apply scoring rubrics uniformly. While lower-performing models, like Databricks (v2-3b), can undermine fairness by omitting key information, selecting LLMs that score highly on clarity, faithfulness, and coverage markedly improves understanding in the application evaluation.

In the experiment results, the steady decline in Cohort Fairness Scores across successive batches show that the proactive fairness checks embedded in the HAIPSE are effectively reducing bias (Feldman et al., 2015). In the first Cohort, the fairness score was 24.56, indicating large score disparities that likely reflected inconsistent treatment or hidden bias. After fairness penalties for biased language were detected, rewards for inclusive terms, and automatic variance monitoring activated, later cohorts saw dramatic improvements: Cohort 3 dropped to 1.43, and Cohorts 4, 5, and 7 all recorded low scores. These low values mean application scores clustered closely around the mean, confirming that evaluation criteria were applied uniformly regardless of an applicant's profile or writing style (Ensign et al., 2018).

By flagging anomalies early and allowing administrators to retrain models or refine rules, HAIPSE not only speeds up the review process but also ensures that decisions remain fair (Rajkomar et al., 2018). Collaboration between AI and human evaluators through the HAIPSE framework enhances decision-making quality by merging the scalability of AI with the contextual expertise of humans. AI efficiently processes vast datasets, identifies patterns (e.g., scoring inconsistencies or demographic biases), and flags anomalies for review, reducing the risk of oversight.

Human evaluators then apply their own judgment to interpret complex cases, such as culturally specific responses or ambiguous language, which AI might misinterpret (Lai et al., 2019). This synergy ensures that decisions are both data-driven and contextually informed. For example, while AI model might highlight essays with unusual score discrepancies, humans can

recognize whether the variance stems from bias, creativity, or other subjective factors, enabling corrective actions like score adjustments.

In conclusion, the HAIPSE boosts user trust by fostering transparency and accountability (Abdul et al., 2018). The HAIPSE helps addressing concerns about "black box" opacity (Doshi-Velez & Kim, 2017). Users gain confidence knowing human oversight validates automated outputs and provides space for appeals, reinforcing system integrity (Holstein et al., 2019). Proactive bias checks and transparent reporting signal a commitment to fairness, critical in high-stake domains like hiring or admissions (Buolamwini & Gebru, 2018). Furthermore, human evaluators bridge the gap between technical AI outputs and user expectations by explaining scores in relatable terms, enhancing perceived legitimacy (Stumpf et al., 2016). Ultimately, this collaborative model builds trust by demonstrating that technology serves as a tool for human-driven fairness, not a replacement for judgment, creating a more credible and inclusive evaluation process (Weller, 2019). Thus, all research questions posed in Chapter 1 have now been fully addressed.

Future work will broaden the HAIPSE applicability and robustness by expanding its data coverage (e.g., supporting different ID formats and multilingual essays), deepening bias mitigation through intersectional audits and counterfactual testing, and transforming the current feedback loop into a real-time active-learning system that automatically retrains models on low-confidence predictions and reviewer overrides. There are ways to enhance transparency and trust with interactive explainability dashboards and extend the framework to multimodal inputs, such as audio or video interviews, while conducting longitudinal studies that link the HAIPSE scores to real-world outcomes (e.g., grant success, job performance) to validate and continuously refine its predictive accuracy.

References

Abdul, A., Vermeulen, J., Wang, D., Lim, B. Y., & Kankanhalli, M. (2018). Trends and trajectories for explainable, accountable and intelligible systems: An HCI research agenda. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (CHI 2018).

Abraham, A. (2005). Rule-based NLP expert systems. In *Handbook of Measuring System Design*, Vol. 2, pp. 913–919. John Wiley & Sons.

Agarwal, M. (2019). An overview of natural language processing. *International Journal for Research in Applied Science and Engineering Technology*, 7(5), pp. 2811–2813.

Aldosari, B., Aldosari, H., & Alanazi, A. (2025). Challenges of artificial intelligence in medicine. *Studies in Health Technology and Informatics*, 323, pp. 16–20.

Alhawiti, K. M. (2014). Natural language processing and its use in education. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 5(12), pp. 72–76.

Allam, A. M. N., & Haggag, M. H. (2012). The question answering systems: A survey. *International Journal of Research and Reviews in Information Sciences (IJRRIS)*, 2(3), pp. 211–221.

Almazrouei, E., Srivastava, H., Natarajan, S., et al. (2023). Falcon-7B-Instruct: Building and benchmarking a 7B-parameter generative language model.

Amershi, S., Weld, D., Vorvoreanu, M., Fourney, A., Nushi, B., Collisson, P., Suh, J., Iqbal, S., Bennett, P. N., Inkpen, K., Kamar, E., Horvitz, E., & Dulac-Arnold, G. (2019). *Guidelines for human–AI interaction*. In Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, pp. 1–13.

Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., & Mané, D. (2016). Concrete problems in AI safety. *arXiv Preprint*, arXiv:1606.06565 [cs.AI].

Bakar, N. H., Kasirun, Z. M., & Salleh, N. (2015). Feature extraction approaches from natural language requirements for reuse in software product lines: A systematic literature review. *Journal of Systems and Software*, 106, pp. 132–149.

Balcıoğlu, Y. S., Artar, M., & Erdil, O. (2023). Data-driven insights into HR recruitment performance: A machine learning approach based on real-world data from an Italian security company. In *Current Debates on Social Sciences,* pp. 282–294.

Bansal, M., Monteiro, V., & Ribeiro, M. (2021). Mitigating bias in rule-based language processing. *Journal of Artificial Intelligence Research, 72*, pp. 1125–1150.

Barocas, S., & Selbst, A. D. (2016). Big data's disparate impact. *California Law Review, 104*(3), pp. 671–732.

Bhandari, M., Madnani, J., & Baldwin, K. (2020). Evaluating the factual consistency of abstractive summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 5215–5225.

Binns, R. (2018). Algorithmic accountability and public reason. *Philosophy & Technology, 31*(4), 543–556. https://doi.org/10.1007/s13347-017-0263-5

Bird, S., Klein, E., & Loper, E. (2009). *Natural language processing with Python*. O'Reilly Media.

Bittner, Thomas & Donnelly, Maureen & Winter, Stephan. (2006). Ontology and Semantic Interoperability.

Bolukbasi, T., Chang, K.-W., Zou, J., Saligrama, V., & Kalai, A. (2016). Man is to computer programmer as woman is to homemaker? Debiasing word embeddings. In *Proceedings of the 30th Conference on Neural Information Processing Systems (NeurIPS 2016)*, pp. 4349–4357. Curran Associates, Inc.

Brants, T. (2003). Natural language processing in information retrieval. In *Proceedings of the 14th Meeting of Computational Linguistics in the Netherlands (CLIN 2003)*, pp. 37–48.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P. & Amodei, D. (2020). Language Models are Few-Shot Learners. *Advances in Neural Information Processing Systems, 33*, pp. 1877–1901.

Bubeck, S., Gunasekar, S., Javaheripi, M., Chen, W., Eldan, R., Gopi, S., Zhang, Y. (2023). *Phi-2: The surprising power of small language models*. Microsoft Research Blog. https://www.microsoft.com/en-us/research/blog/phi-2-the-surprising-power-of-small-language-models/

Buchanan, B. G., & Shortliffe, E. H. (1984). *Rule-based expert systems: The MYCIN experiments of the Stanford heuristic programming project*. Addison-Wesley.

Buolamwini, J., & Gebru, T. (2018). Gender shades: Intersectional accuracy disparities in commercial gender classification. *Proceedings of Machine Learning Research, 81*, pp. 1–15.

Buslaev, A., Parinov, A., Khvedchenya, E., Iglovikov, V. I., & Kalinin, A. (2020). Albumentations: Fast and flexible image augmentations. Information, 11(2), 125.

Cappelli, P. (2019). Your approach to hiring is all wrong. *Harvard Business Review*, 97(3), pp. 48–58.

Cavoukian, A. (2009). *Privacy by design: The 7 foundational principles*. Information and Privacy Commissioner of Ontario.

Celikyilmaz, A., Bosselut, A., He, X., & Choi, Y. (2020). Evaluation of text generation: A survey.

Chen, Q., Wu, Y., Shen, Y., et al. (2023). A systematic comparison of open-source large language models on summarization tasks.

Chouldechova, A., & Roth, A. (2020). A snapshot of the frontiers of fairness in machine learning. *Communications of the ACM, 63*(5), pp. 82–89.

Chowdhury, G. G. (2003). Natural language processing. *Annual Review of Information Science and Technology*, 37(1), pp. 51–89.

Conlon, S., Reithel, B., Aiken, M., & Shirani, A. (1994). A natural language processing-based group decision support system. *Group Decision and Negotiation*, 3(4), pp. 345–362.

Conover, J., Chang, K., Harris, L., et al. (2023). Dolly-v2-3B: Open weights for an instruction-following chat model (Technical Report No. DBX-RR-2023-04). Databricks.

Corbett-Davies, S., & Goel, S. (2018). The measure and mismeasure of fairness: A critical review of fair machine learning. *arXiv preprint*, arXiv:1808.00023.

Crawford, K., & Paglen, T. (2021). *Excavating AI: The politics of training sets*. Retrieved from Excavating.

Czerwinski, M., Horvitz, E., & Wilhite, S. (2009). *One-click authentication: Single sign-on in large distributed systems* (Technical Report MSR-TR-2009-14). Microsoft Research.

Dasu, T., & Johnson, T. (2003). *Exploratory data mining and data cleaning*. John Wiley & Sons.

De-Arteaga, A., Romanov, N., Wallach, H., Chayes, I., Borgesius, B., Dwork, C., & Procaccia, D. (2019). Bias in bios: A case study of semantic representation bias in a high-stakes setting. In *Proceedings of the 2019 Conference on Fairness, Accountability, and Transparency (FAccT 2019)*, pp. 120–128.

Degand, L., & Muller, P. (2020). Introduction to the special issue on dialogue and dialogue systems. *Traitement Automatique des Langues*, 61(3), pp. 7–15.

Dellermann, D., Calma, A., Lipusch, N., Weber, T., Weigel, S., & Ebel, P. (2019). The future of human–AI collaboration: A taxonomy of design knowledge for hybrid intelligence systems. *Proceedings of the 52nd Hawaii International Conference on System Sciences* (HICSS 2019), pp. 1–10.

Demner-Fushman, D., Chapman, W. W., & McDonald, C. J. (2009). What can natural language processing do for clinical decision support? *Journal of Biomedical Informatics*, 42(5), pp. 760–772.

Deng, L., & Yu, D. (2014). Deep learning: Methods and applications. *Foundations and Trends in Signal Processing, 7*(3–4), pp. 197–387.

Devaraju, S. (2022). NLP in AI-driven recruitment systems. *International Journal of Scientific Research in Computer Science Engineering and Information Technology*, 8(3), pp. 555–566.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). 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 (NAACL-HLT)* Vol. 1, pp. 4171–4186.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation, 10*(7), pp. 1895–1923.

Doan, S., Conway, M., Phuong, T. M., & Ohno-Machado, L. (2014). Natural language processing in biomedicine: A unified system architecture overview. In H. Chen, S. Fuller, C. Friedman, & W. Hersh (Eds.), *Medical Informatics: Knowledge Management and Data Mining in Biomedicine*, pp. 275–294. Springer.

Dong, Y., Zhang, H., Wei, F., et al. (2021). Towards unified key phrase generation for factual consistency and informativeness.

Doshi-Velez, F., & Kim, B. (2017). Towards a rigorous science of interpretable machine learning. arXiv preprint arXiv:1702.08608.

Du, K. et al. (2025). Natural language processing in finance: A survey. Information Fusion.

Du, K., Zhao, Y., Mao, R., Xing, F., & Cambria, E. (2025). Natural language processing in finance: A survey. *Information Fusion*.

Durieux, V., & Gevenois, P. A. (2010). Bibliometric indicators: Quality measurements of scientific publication. *Radiology*, 255(2), pp. 342–351.

Dwork, C., Hardt, M., Pitassi, T., Reingold, O., & Zemel, R. (2012). Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference (ITCS)*, pp. 214–226. ACM.

Ellegaard, O. (2018). The application of bibliometric analysis: Disciplinary and user aspects. *Scientometrics*, 116(2), pp. 181–202.

Ensign, D., Friedler, S. A., Neville, S., Scheidegger, C., & Venkatasubramanian, S. (2018). Runaway feedback loops in predictive policing. *Proceedings of Machine Learning Research, 81*, pp. 1–12.

Esposito De Falco, S., Renzi, A., Orlando, B., & Cucari, N. (2017). Open collaborative innovation and digital platforms. Production Planning & Control, 28(16), pp. 1344–1353.

Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. (2010). The PASCAL visual object classes (VOC) challenge. *International Journal of Computer Vision, 88*(2), pp. 303–338.

Fabbri, A. R., Kryściński, W., McCann, B., Xiong, C., Socher, R., & Radev, D. (2021). SummEval: Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics, 9*, pp. 391–409.

Feigenbaum, E. A. (1982). Knowledge engineering in the 1980s. In W. Reitman (Ed.), *Artificial intelligence applications for business*, pp. 37–55.

Feldman, M., Friedler, S. A., Moeller, J., Scheidegger, C., & Venkatasubramanian, S. (2015). Certifying and removing disparate impact. In *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2015)*, pp. 259–268.

Finin, T. W. (1986). GUMSHOE: An expert locator for consultation systems. In Proceedings of the 24th Annual Meeting of the Association for Computational Linguistics, pp. 80–86. Association for Computational Linguistics.

Friedman, C., Alves, P., Xu, H., Rahman, M., & Elhadad, N. (2013). Natural language processing in an industrial healthcare system: NLP for clinical narratives. *Journal of the American Medical Informatics Association*, 20(5), pp. 814–823.

Gallagher, J. (2023). *What is data augmentation? The ultimate guide.* Roboflow Blog. https://blog.roboflow.com/data-augmentation/

Gallant, S. I. (1993). *Neural network learning and expert systems*. Cambridge, MA: MIT Press.

Gallegos, I. O., Rossi, R. A., Barrow, J., Tanjim, M. M., Kim, S., Dernoncourt, F., Yu, T., Zhang, R., & Ahmed, N. K. (2024). *Bias and fairness in large language models: A survey*. Computational Linguistics, 50(3), pp. 1097–1160.

Garg, R., Kiwelekar, A. W., Netak, L. D., & Bhate, S. S. (2021). Potential use-cases of natural language processing for a logistics organization, Vol. 942, pp. 157–191. Springer.

Ge, D., Wang, S., Chen, Q., et al. (2021). YOLOX: Exceeding YOLO series in 2021. arXiv preprint arXiv:2107.08430.

Gehman, T., Gururangan, S., Sap, H., Choi, Y., & Zettlemoyer, L. (2020). Real toxicity prompts: Evaluating neural toxic degeneration in language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 3356–3369.

Geiger, R. S., Yu, K., Yang, Y., & McGregor, K. C. (2020). Garbage in, garbage out? Do context features and label biases distort fairness analysis in machine learning. In *Proceedings of the 2020 ACM Conference on Fairness, Accountability, and Transparency*, pp. 295–305.

Genest, P.-E., & Lapalme, G. (2012). A fully abstractive approach to guided summarization. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL 2012)*, pp. 354–358. Association for Computational Linguistics.

Giarelis, N., Mastrokostas, C., & Karacapilidis, N. (2023). Abstractive vs. extractive summarization: An experimental review. *Applied Sciences*, 13(13).

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.

Goyal, T., Li, J. J., & Durrett, G. (2022). News summarization and evaluation in the era of GPT-3. *Findings of ACL 2022*, pp. 4784–4804.

Greene, J. (2025). *Computer vision augmentations: An introduction.* Roboflow Blog. https://blog.roboflow.com/computer-vision-augmentations/

Grusky, M., Naaman, M., & Artzi, Y. (2018). Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *Proceedings of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2018)*, pp. 708–719.

Gupta, A., Dengre, V., Kheruwala, H. A., & Shah, M. (2020). Comprehensive review of text-mining applications in finance. *Financial Innovation, 6*(39).

Hahn, A., & Mani, I. (2000). The challenges of automatic summarization. *Computer*, 33(11), pp. 29-36.

Hajian, A., Bonchi, F., & Castillo, C. (2016). Algorithmic bias: From discrimination discovery to fairness-aware data mining. *ACM SIGKDD Explorations, 17*(2), pp. 18–22.

Hardt, M., Price, E., & Srebro, N. (2016). Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems 29 (NeurIPS 2016)*, pp. 3315–3323.

He, K., Zhang, X., Ren, S., & Sun, J. (2014). Spatial pyramid pooling in deep convolutional networks for visual recognition. In *Proceedings of the European Conference on Computer Vision (ECCV 2014)*, pp. 346–361.

Holstein, K., Wortman Vaughan, J., Daumé III, H., Dudík, M., & Wallach, H. (2019). Improving fairness in ML systems: What do industry practitioners need? In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (CHI 2019).

Holzinger, A. (2016). Interactive machine learning for health informatics: When do we need the human-in-the-loop? *Brain Informatics, 3*(2), pp. 119–131.

Honnibal, M., & Montani, I. (2017). *spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing*. (Software Release Technical Report). Explosion AI.

Hossain, E., Rahman, M., & Ahmed, A. (2023). EHR data transformation: An overview. *Journal of Biomedical Informatics*. (Advance online publication).

Howard, J., & Ruder, S. (2018). Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018)*, pp. 328–339.

Huang, B., & Wang, H. (2016). *Debiasing rule-based classifiers via constraint–driven post-processing*. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 885-894.

Humphreys, K., Gaizauskas, R., & Sanderson, M. (1999). *The University of Sheffield's TREC-8 Question Answering system*. Proceedings of the Eighth Text Retrieval Conference (TREC-8), pp. 409–416.

Hunter, J. D., Droettboom, M., & The Matplotlib Development Team. (2025, May 8). matplotlib (Version 3.10.3) [Computer software]. Python Package Index.

Hyscaler. (2024). *Llama 2 vs. Mistral 7B: A comparison of large language models.* Retrieved from https://hyscaler.com/insights/llama-2-vs-mistral-7b

Ibáñez, O., Cordón, Ó., Damas, S., & Magdalena, L. (2009). A review on the application of hybrid artificial intelligence systems to optimization problems in operations management.

Iroju, O. G., & Olaleke, J. O. (2015). A systematic review of natural language processing in healthcare. *International Journal of Information Technology and Computer Science*, 7(8), pp. 44–50.

Jach, T., & Zięski, T. (2015). Inference in expert systems using natural language processing. In *Communications in Computer and Information Science (CCIS)*, 538, pp. 288–298.

Jackson, P. (1999). *Introduction to expert systems* (3rd ed.). Addison-Wesley.

Jain, A. K., & Bolle, R. (2006). *Biometrics: Personal identification in networked society*. Springer.

Järvelä, S. (2023). Human–AI co-evolution and learning in hybrid-intelligence systems: Toward explainable collaboration. *British Journal of Educational Technology, 54*(4), pp. 1456 – 1472.

Järvelä, S., Zhao, G., Nguyen, A., & Chen, H. (2025). Hybrid intelligence: Human–AI coevolution and learning. *British Journal of Educational Technology, 56*(2), pp. 455–468. https://doi.org/10.1111/bjet.13560

Jiang, X., Liu, Y., Wang, Z., et al. (2023). Mixtral-8×7B-Instruct-v0.1: Fine-tuning a large language model for domain-specific summarization. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 4521–4532.

Jobin, A., Ienca, M., & Vayena, E. (2019). The global landscape of AI ethics guidelines. *Nature Machine Intelligence, 1*(9), pp. 389–399.

Jocher, G., Chaurasia, A., Qiu, J., & Stoken, E. (2020). *YOLOv5* [Computer software]. Ultralytics.

Joseph, S. R., Hlomani, H., Letsholo, K., Kaniwa, F., & Sedimo, K. (2016). Natural language processing: A review. *International Journal of Research in Engineering and Applied Sciences (IJREAS)*, 6(3), pp. 207–210.

Jurafsky, D., & Martin, J. H. (2021). Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition.

Kahneman, D. (2011). *Thinking, fast and slow*. New York: Farrar, Straus and Giroux.

Kairouz, P., Chen, M., McMahan, H. B., & Ramage, D. (2022). Advances and open problems in federated learning (v2). *arXiv preprint*, arXiv:1912.04977.

Kang, Y., Cai, Z., Tan, C.-W., Huang, Q., & Liu, H. (2020). Natural language processing (NLP) in management research: A literature review. *Journal of Management Analytics*, 7(2), pp. 139–172.

Kazmierczak, C. T. (1990). *Automatic synthesis of PROLOG rules for expert systems*. Proceedings of the Fourth International Conference on Industrial & Engineering Applications of Artificial Intelligence and Expert Systems (IEA/AIE ’90), pp. 514–521. Springer.

Kearns, M., & Roth, A. (2020). *The ethical algorithm: The science of socially aware algorithm design*. Oxford University Press.

Kenfack, C., Islam, M. R., & Watson, R. (2021). Human-in-the-loop ensemble learning for fair and equitable algorithmic decision making. *Journal of Artificial Intelligence Research, 71*, pp. 123–145.

Kent, K., & Souppaya, M. (2013). *Guide to computer security log management* (NIST Special Publication). National Institute of Standards and Technology.

Kleinberg, J., Lakkaraju, H., Leskovec, J., Ludwig, J., & Mullainathan, S. (2018). Human decisions and machine predictions. *The Quarterly Journal of Economics, 133*(1), pp. 237–293.

Kleinberg, J., Mullainathan, S., & Raghavan, M. (2017). Inherent trade-offs in the fair determination of risk scores. In *Proceedings of the 8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25 (NIPS 2012)*, pp. 1097–1105.

Kroll, J. A., Huey, J., Barocas, S., Felten, E., Reidenberg, M., Robinson, D., & Yu, H. (2017). Accountable algorithms. *University of Pennsylvania Law Review, 165*(3), pp. 633–705.

Kryściński, W., McCann, B., Xiong, C., & Socher, R. (2020). Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 9332–9346.

Kshetri, N. (2014). Big data's impact on privacy, security and consumer welfare. *Telecommunications Policy, 38*(11), pp. 1134–1145.

Kuhn, M., & Johnson, K. (2013). *Applied predictive modeling*. Springer.

Kulkarni, N., et al. (2012). NLP for software development. Journal of Systems and Software, 85(9).

Lai, V., Liu, H., Chen, C., & Lakkaraju, H. (2019). Human-in-the-loop learning: Designing trust-aware explanations. In *NeurIPS 2019 Human-Centered AI Workshop*.

Li, J., Burnham, J. F., Lemley, T. D., & Britton, R. M. (2010). Citation analysis: Comparison of Web of Science and Scopus impact on citation counts in the field of medical informatics. *Journal of the Medical Library Association, 98*(2), pp. 187–195.

Li, Y., Zhang, L., & Zhang, Y. (2023). *Fairness of ChatGPT*. arXiv preprint arXiv:2305.18569.

Li, Z., Liu, S., Yuan, L., & Sun, J. (2020). Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. In *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*.

Liddy, E. D. (1998). Natural language processing for information retrieval and knowledge discovery. In P. A. Cochrane & E. H. Johnson (Eds.), *Visualizing Subject Access for 21st Century Information Resources*, pp. 63–79.

Lifewire. (2024). *Llama 2 vs. Llama 3: What's new in Meta's AI models?* Retrieved from https://www.lifewire.com/llama-2-vs-llama-3-8714445

Lin, C. Y. (2004). ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL Workshop*, pp. 74–81.

Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017)*, pp. 936–944.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. In *Proceedings of the European Conference on Computer Vision (ECCV 2014)*, pp. 740–755.

Liu, B., & Mazumder, S. (2021). Lifelong and continual learning dialogue systems: Learning during conversation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(17), pp. 15058–15063

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2018). Path aggregation network for instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2018)*, pp. 8759–8768.

Liu, Y., & Lapata, M. (2019). Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3730–3740.

Madras, D., Creager, E., Louizos, C., & Zemel, R. S. (2018). Learning adversarially fair and transferable representations. In *Proceedings of the 35th International Conference on Machine Learning (ICML 2018)*, pp. 3381–3389.

Mah, P. M., Skalna, I., Muzam, J., & Song, L. K. (2022). Analysis of natural language processing in the FinTech models of mid-21st century. *Journal of Information Technology and Digital World*, 4(3), pp. 183–211.

Mäkelä, E. (2008). Survey of semantic search research. In *Proceedings of the Seminar on Knowledge Management on the Semantic Web (University of Helsinki, 2007)*. University of Helsinki.

Manaris, B. (1998). Natural language processing: A human–computer interaction perspective. *Advances in Computers*, 47, pp. 1–66.

Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval*. Cambridge University Press.

Mase, M., Owen, A. B., & Seiler, B. B. (2021). Cohort Shapley value for algorithmic fairness. *arXiv preprint*, arXiv:2105.07168.

Maynez, J., Narayan, S., Bohnet, B., & McDonald, R. (2020). On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020)*, pp. 501–514.

McNamara, D. S., Crossley, S. A., Roscoe, R. D., Allen, L. K., & Dai, J. (2016). A hierarchical classification approach to automated writing evaluation. *Computers and Composition, 39*, pp. 35–50.

McTear, M. F. (2020). Conversational AI: Dialogue Systems, Conversational Agents, and Chatbots, Synthesis Lectures on Human Language Technologies, Vol. 13, No. 3, pp. 1–251.

Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., & Galstyan, A. (2021). A survey on bias and fairness in machine learning. *ACM Computing Surveys*, 54(6), pp. 1–35.

Mell, P., & Grance, T. (2011). *The NIST definition of cloud computing* (NIST Special Publication 800-145). National Institute of Standards and Technology.

Meta AI. (2024). *Meta-LLaMA-3-8B-Instruct: Open and efficient transformer models for instruction-following*(Technical Report No. 2024-01). Meta AI.

Mihalcea, R. (2003). Turning WordNet into an information retrieval resource: Systematic polysemy and conversion to hierarchical codes. *International Journal of Pattern Recognition and Artificial Intelligence*, 17(5), pp. 689–704.

Miranda, C., & Kessaci, Y. (2020). Hybrid supervised reinforced model for dialogue systems. *arXiv Preprint*, arXiv:2011.12707 [cs.CL], 1–11.

Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Hutchinson, B., Spitzer, E., Vasserman, L., & Raji, I. D. (2019). Model cards for model reporting. In *Proceedings of the 2019 Conference on Fairness, Accountability, and Transparency (FAccT 2019)*, pp. 220–229.

Mittelstadt, B. D., Allo, P., Taddeo, M., Wachter, S., & Floridi, L. (2016). The ethics of algorithms: Mapping the debate. *Big Data & Society, 3*(2), pp. 1–21.

Munot, N., & Govilkar, S. (2015). Abstractive vs. extractive summarization. International Journal of Computer Applications, 116(22), pp. 1–5.

Narayan, S., Cohen, S. B., & Lapata, M. (2018). Ranking sentences for extractive summarization with reinforcement learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 1747–1759.

Nimbekar, T., et al. (2019). Real-time recruitment insights. International Journal of Human Resource Management, 30(15), pp. 2234–2252.

Osyk, B. L., & Vijayaraman, B. S. (1995). Hybrid expert systems: Integrating neural networks with rule-based reasoning. *Expert Systems with Applications, 9*(4), pp. 511–520.

Padilla, R., Passos, W. L., Dias, T. L. B., Netto, S. L., & da Silva, E. A. B. (2021). A comparative analysis of object detection metrics with a companion open-source toolkit. *Electronics, 10*(3).

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). BLEU: A method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311-318.

Parisi, G. I., Kemker, R., Part, J. L., Kanan, C., & Wermter, S. (2019). Continual lifelong learning: A review. *Neural Networks, 113*, pp. 54–71.

Paulus, R., Xiong, C., & Socher, R. (2018). A deep reinforced model for abstractive summarization. In *Proceedings of the International Conference on Learning Representations (ICLR 2018)*.

Peyrard, M., Herrmann, T., & Eckle-Kraske, S. (2017). Learning lexical evaluation metrics for sentence-level summarization quality. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1347–1357.

Pileggi, Salvatore. (2023). Ontology in Hybrid Intelligence: a concise literature review. 10.48550/arXiv.2303.17262.

Plotly Technologies Inc. (2025, June 30). Dash (Version 3.1.1).

Provost, F., & Fawcett, T. (2013). *Data science for business: What you need to know about data mining and data-analytic thinking*. O'Reilly Media.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research, 21*(140), pp. 1–67.

Raji, I. D., & Schmidt, L. (2022). Challenges in the auditing of AI systems. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency* (FAccT 2022), pp. 1619–1630.

Raji, I. D., Smart, A., White, R. N., et al. (2020). Closing the AI accountability gap: Defining an end-to-end framework for internal algorithmic auditing. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pp. 33–44.

Rajkomar, A., Oren, E., Chen, K., Dai, A. M., Hajaj, N., Hardt, M., Dean, J. (2018). Scalable and accurate deep learning with electronic health records. *npj Digital Medicine, 1*(1), Article 18.

Rajman, M., & Besançon, R. (1997). Text mining: Natural language techniques and text mining applications. In *Proceedings of the IFIP Conference on Database Semantics (DS-7)*, pp. 215–226. Chapman & Hall.

Rao, S. (2024, October). YOLOv11 architecture explained: Next-level object detection with enhanced speed and accuracy. *Medium*.

Redmon, J., & Farhadi, A. (2018). YOLOv3: An incremental improvement. *arXiv preprint*, arXiv:1804.02767.

Rekha, V. (2023). Natural language processing in clinical practice. *Health Informatics Journal*, 29(1), pp. 1–15.

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems 28 (NIPS 2015)*, pp. 91–99.

Reshamwala, A., Mishra, D., & Pawar, P. (2013). Review on natural language processing. *Engineering Science and Technology: An International Journal (ESTIJ)*, 3(1), pp. 113–116.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?" Explaining the predictions of any classifier. In KDD '16, pp. 1135-1144.

Ricketts, I., et al. (2023). Bidirectional transformers for text classification. Proceedings of ACL, pp. 456–467.

Rindflesch, T. C., Fiszman, M., & Libbus, B. (2005). Semantic interpretation for the biomedical research literature, pp. 399–422. Springer.

Roboflow Inc. (2025, July 9). Roboflow (Version 1.2.1). Python Package Index.

Ruder, S. (2019). *Neural transfer learning for natural language processing* (Doctoral dissertation, National University of Ireland, Galway).

Saggion, H., & Lapalme, G. (2000). Concept-based summarization. *Natural Language Engineering*, 6(2), pp. 211–223.

Sally Jo Cunningham, Littin, J., & Witten, I. H. (2002). Applications of machine learning in information retrieval. *Annual Review of Information Science and Technology*, 34, pp. 341–384.

Sammut, C., & Webb, G. I. (Eds.). (2011). *Encyclopedia of machine learning*. Springer.

Sánchez-Bocanegra, C. L. et al. (2024). Clinical decision support and natural language processing in medicine: Systematic literature review. *Journal of Medical Internet Research, 26*(9).

Sandhu, R. S., Coyne, E. J., Feinstein, H. L., & Youman, C. E. (1996). Role-based access control models. *IEEE Computer, 29*(2), pp. 38–47.

Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J.-F., & Dennison, D. (2015). Hidden technical debt in machine learning systems. In *Advances in Neural Information Processing Systems 28 (NeurIPS 2015)*, pp. 2503–2511.

Selbst, A. D., Boyd, D., Friedler, S. A., Venkatasubramanian, S., & Vertesi, J. (2019). Fairness and abstraction in sociotechnical systems. *In Proceedings of the 2019 Conference on Fairness, Accountability, and Transparency*, pp. 59–68. Association for Computing Machinery.

Shazeer, N. (2019). Fast transformer decoding: One write-head is all you need. arXiv preprint arXiv:1911.02150.

Shazeer, N. (2020). GLU variants improve transformer. arXiv preprint arXiv:2002.05202.

Shen, Y., Zhang, M., Chen, T., et al. (2024). SmolLM2-1.7B-Instruct: A lightweight model for fast and concise summarization. In *Proceedings of the 2024 Conference on Neural Information Processing Systems (NeurIPS)*, pp. 7812–7826.

Sherif, N. H., Ali, R. R., & Jawarneh, S. A. (2023). Integrating NLP in the business decision support system to promote customer loyalty. In *Proceedings of the 2023 International Conference on Emerging Research in Computational Sciences (ICERCS)*, pp. 1–6.

Shrestha, R., Kalita, D., & Mahmood, A. N. (2021). Deep learning-based identity-document authentication using MRZ and portrait verification. *Neural Computing and Applications, 33*, pp. 10457–10470.

Sinha, A. K., Akhtar, M. A. K., & Kumar, A. (2021). Automated resume screening using natural language processing and machine learning: A systematic review. *Machine Learning and Information Processing* (Advances in Intelligent Systems and Computing, Vol. 1311, pp. 207–214.

Smeaton, A. F. (1999). WordNet in IR. Information Retrieval, pp. 135–152.

Smith, J., & Martin, K. (2019). Rethinking Applicant Tracking: Semantic Analysis in the Modern Era. *Journal of HR Technology, 14*(2), pp. 112–119.

Smith, R. (2007). An overview of the Tesseract OCR engine. In *Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, Vol. 2, pp. 629–633.

Srihari, R., & Li, W. (2000). A question answering system supported by information extraction. In *Proceedings of the 6th Applied Natural Language Processing Conference (ANLP 2000)*, pp. 166–172. Association for Computational Linguistics.

Strohmeier, S., & Piazza, F. (2015). Artificial Intelligence techniques in human resource management—A conceptual exploration. *In Proceedings of the 4th International Conference on Data Mining.*

Studer, R., Benjamins, V. R., & Fensel, D. (1998). Knowledge engineering: Principles and methods. *Data & Knowledge Engineering, 25*(1–2), pp. 161–197.

Stumpf, S., Rajaram, V., Li, L., Wong, W., Burnett, M., & Grover, M. (2016). Interacting meaningfully with machine learning systems: Three experiments. *ACM Transactions on Interactive Intelligent Systems, 7*(4), Article 16.

Su, J., Lu, Y., Pan, J., Wu, W., Wang, Z., Liu, Y., Li, Z., & Wen, L. (2021). *RoFormer: Enhanced Transformer with Rotary Position Embedding* (arXiv 2104.09864).

Subhashini, R., & Kumar, V. J. (2011). Noun-verb vector models. Journal of Intelligent Systems, 20(3), pp. 287–302.

Subhashini, R., & Kumar, V. J. S. (2010). Shallow NLP techniques for noun phrase extraction. In *Proceedings of the 2010 International Conference on Trend in Information Sciences & Computing (TISC2010)*, pp. 73–77.

Suresh, H., & Guttag, J. V. (2019). A framework for understanding sources of harm in machine learning, Association for Computing Machinery, pp. 113-122.

Swire, C., & Ahmad, K. (2011). Privacy and security: A framework for non-disclosure agreements in research collaborations. *Journal of Law and Policy, 21*(2), pp. 345–378.

Tezgider, M., Yildiz, B., & Aydin, G. (2022). Text classification using improved bidirectional transformer. *Concurrency and Computation: Practice and Experience*, 34(9).

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., & Jernite, Y. (2023). LLaMA: Open and Efficient Foundation Language Models. *arXiv preprint arXiv:2302.13971*.

Upadhyay, S., Sharma, N., & Patel, R. (2020). Human-in-the-loop hybrid approach for equitable applicant screening in high-stakes domains. In *Proceedings of the IEEE International Conference on Responsible AI (IRAI 2020)*, pp. 45–52.

Van Esch, P., Black, J. S., & Ferolie, J. (2019). Marketing AI recruitment: The next phase in job application and selection. *Computers in Human Behavior,* 90, p. 215.

Veale, M., Van Kleek, R., & Binns, L. (2018). Fairness and accountability design needs for algorithmic support in high-stakes public sector decision-making. In *Proceedings of the 2018 ACM Conference on Fairness, Accountability, and Transparency*, pp. 239–248.

Verma, S., & Rubin, J. (2018). Fairness definitions explained. In *Proceedings of the 2018 IEEE/ACM International Workshop on Software Fairness (FairWare 2018)*, pp. 1–7.

Vogt, A., & Von dem Bussche, P. (2017). Encryption techniques for database security: A survey. *International Journal of Information Security and Privacy, 11*(4), pp. 1–20.

Voorhees, E. M. (2000). The TREC-8 question answering track report. In *Proceedings of the Eighth Text Retrieval Conference (TREC-8)*. Gaithersburg, MD: NIST.

Wachter, S., Mittelstadt, B., & Floridi, L. (2017). Why a right to explanation of automated decision-making does not exist in the General Data Protection Regulation. *International Data Privacy Law, 7*(2), pp. 76–99.

Wang, C. Y., Sun, Q., Liu, X., Feng, Z., Y., & Guo, B. (2020). CSPNet: A new backbone that can enhance learning capability of CNN. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW 2020)*.

Wang, Y., & Zhang, Z. (2020). An ontology-based approach to keyword extraction for domain-specific text classification. *IEEE Access*.

Waskom, M. (2024, January 25). seaborn (Version 0.13.2) [Computer software]. Python Package Index.

Waterman, D. A. (1986). *A guide to expert systems*. Addison-Wesley.

Weiss, K., Khoshgoftaar, T. M., & Wang, D. (2016). A survey of transfer learning. *Journal of Big Data*, 3(1), p. 9.

Weiss, S. M., Indurkhya, N., Zhang, T., & Damerau, F. J. (2015). *Text mining: Predictive methods for analyzing unstructured information* (2nd ed.). Springer.

Weller, A. (2019). Transparency: Motivations and challenges. In W. Samek, G. Montavon, A. Vedaldi, L. K. Hansen, & K.-R. Müller (Eds.), *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, pp. 23–40, Springer.

Woo, S., Park, J., Lee, J. Y., & Kweon, I. S. (2018). CBAM: Convolutional block attention module. In *Proceedings of the European Conference on Computer Vision (ECCV 2018)*, pp. 3–19.

Wu, B., Zhang, L., Li, X., & Kumar, S. (2023). Evaluating human-AI collaboration in essay summarization: Reducing cognitive load with LLM-generated overviews. In *Proceedings of the 2023 Conference on Human Factors in Computing Systems (CHI),* pp. 1–12.

Yousef, M., McDougall, L., & King, J. (2020). Adaptive document validation in biometric systems. *IEEE Transactions on Information Forensics and Security, 15*(3), pp. 504–516.

Zadeh, L. A. (1965). Fuzzy sets. *Information and Control, 8*(3), pp. 338–353.

Zemel, R., Wu, Y., Swersky, K., Pitassi, T., & Dwork, C. (2013). Learning fair representations. *Proceedings of the 30th International Conference on Machine Learning* (ICML 2013), pp. 325–333.

Zhang, B., & Sennrich, R. (2019). *Root Mean Square Layer Normalization*.

Zhang, L., & Zhao, X. (2019). Domain-specific keyword extraction using ontology mapping. *International Journal of Semantic Web and Information Systems*, 15(2), pp. 45–60.

Zhang, S., Chi, C., Yao, Y., Lei, Z., & Li, S. Z. (2021). Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV 2021)*, pp. 975–984.

Zhang, T., Song, K., Tan, W., et al. (2023). Benchmarking large language models for abstractive summarization.

Zhao, B., Chen, C., Wang, Q.-W., He, A., & Xia, S.-T. (2023). Combating unknown bias with effective bias-conflicting scoring and gradient alignment. *Proceedings of the AAAI Conference on Artificial Intelligence, 37*(3), pp. 3561–3569.

Zhu, T. (2023). NLP applications in education. Computers & Education, 184.

Zupic, I., & Čater, T. (2015). Bibliometric methods in management and organization. *Organizational Research Methods*, 18(3), pp. 429–472.