Corpus Based Approach to Identifying and Documenting Legal Terminology in Malayalam.

By

Joel P John | H00MACL202300087

A Dissertation

Submitted to the Department of Computational Linguistics

School of Language Sciences

The English and Foreign Languages University

Hyderabad - 500007, India

In Partial Fulfilment of the Requirement

For the Degree of Master of Arts

April 2025

Supervisor

Dr. Atreyee Sharma

CERTIFICATE

This is to certify that the work contained in this project entitled "**Corpus Based Approach to Identifying and Documenting Legal Terminology in Malayalam**" submitted by **Joel P John** (**H00MACL202300087**) to the English and Foreign Languages University, Hyderabad towards the partial requirement of Master of Arts in Computational Linguistics has been carried out by him under my supervision and that it has not been submitted elsewhere for the award of any degree.

Dr. Atreyee Sharma

Dissertation Supervisor

DECLARATION

I hereby confirm that the dissertation titled "**Corpus Based Approach to Identifying and Documenting Legal Terminology in Malayalam**" has not previously formed the basis for the award of any degree, diploma, associateship, fellowship, or any other similar title of recognition from any other university or institution.

Name of the student: Joel P John

Registration No.: H00MACL202300087

Date: 01.05.2025

ACKNOWLEDGEMENT

First and foremost, I offer my deepest gratitude to God Almighty, whose grace, guidance, and unwavering presence have sustained me throughout this academic journey. It is by His strength and wisdom that I have been able to complete this dissertation.

I would like to sincerely thank the Vice Chancellor for fostering an academic environment that encourages research, innovation, and interdisciplinary exploration.

A special word of thanks is due to Prof. Hariprasad, whose inspiring teaching and commitment to the discipline have profoundly shaped my understanding and dedication to this field. I remain grateful for his encouragement and scholarly example.

My heartfelt appreciation goes to my supervisor, Dr. Atreyee, for her patient guidance, critical insights, and continuous support throughout this research. Her expertise in linguistics and thoughtful feedback have greatly enriched the quality of this work.

I also wish to thank my classmates and academic peers, whose discussions, questions, and camaraderie have provided both intellectual stimulation and emotional support during this process.

A special mention must be made of Mr. Shone, an advocate who offered valuable legal insight and assistance during the refinement of the corpus. His practical knowledge of legal discourse was instrumental in bridging linguistic analysis with legal reality.

Lastly, I express my gratitude to everyone named and unnamed who contributed to this dissertation in ways large and small. This work stands as a reflection of the collective support and generosity I have been blessed to receive.

Abstract

Legal language in Malayalam can be complex and difficult to understand, creating obstacles for legal professionals, students, translators, and everyday citizens. Despite the importance of legal texts in daily life contracts, court decisions, and government policies there is a lack of systematic documentation and technological support for Malayalam legal terminology. This absence of resources makes it harder to simplify legal language and develop useful tools like legal summarization systems and translation services.

This research aims to address that gap by developing a structured, data-driven approach to identifying and documenting Malayalam legal jargon. The process involves analyzing a large collection of legal texts, cleaning and organizing the data, and identifying frequently used legal terms especially multi-word phrases. These legal terms are then compared against a general Malayalam corpus to assess their distinctiveness in legal writing.

After ranking and filtering the most distinctive legal terms, the study compiles a glossary of 100 highly specific legal words, complete with usage examples, statistical information, and a refined reference corpus suitable for corpus-based training. This glossary is stored in formats like JSON and CSV, making it accessible for future research and technological applications.

By bridging the gap between specialized legal terminology and everyday Malayalam usage, this study lays the foundation for better legal education, improved accessibility, and advancements in computational linguistics. The methodology developed here can also be adapted for other low-resource languages and specialized domains.

Table of Contents

CHAPTER 1

Legal Jargon Documentation

1.1 Background

In recent years, making legal language more accessible has become a pressing issue especially in multilingual countries like India. While English has long been the language of the judiciary, much of legal work, from contracts to court documents, happens in regional languages like Malayalam. Despite this, Malayalam legal texts lack proper digital infrastructure, making them hard to access and process. Many legal documents exist only in scanned images or poorly formatted PDFs, often sourced from aging government archives. Optical Character Recognition (OCR) tools struggle with transcribing them accurately due to the complexity of Malayalam script and the poor quality of these scans. Adding to the challenge, there is no centralized or standardized collection of Malayalam legal texts, making systematic linguistic analysis nearly impossible. This gap creates difficulties for legal professionals, researchers, and developers who want to build digital tools that make legal resources more accessible.

At the same time, Natural Language Processing (NLP) and Machine Learning (ML) have made great strides in automatically summarizing legal texts, helping lawyers, judges, and citizens by condensing lengthy judgments, statutes, and contracts into clear, concise summaries. However, most of these advancements have focused on high-resource languages like English, French, and Chinese. The lack of large, well-annotated datasets for Indian languages including Malayalam has severely limited progress in this field. Legal writing is particularly challenging for AI models to understand because it relies on archaic vocabulary, formal structures, and highly specific terminology. To bridge this gap, developing domain-specific resources is essential for improving NLP applications in Indian legal contexts.

Corpus linguistics, an approach that studies language through large collections of real-world texts, has proven to be a powerful tool for both linguistic and AI-driven research. When combined with computational linguistics, corpus-based methods help build language technologies such as POS taggers and neural machine translation systems.

As deep learning models grow more data-hungry, there is an urgent need for clean, well-structured corpora that can serve as high-quality training datasets.

Legal texts, with their structured language and reliance on precedent, are ideal for corpus-based analysis and AI model development. Training AI-driven summarization models on a well-curated legal corpus allows them to grasp the unique patterns of legal writing, improving accuracy and relevance in generated summaries.

This research takes a crucial step in that direction by documenting and identifying legal jargon to create a refined Malayalam legal corpus. While legal texts are becoming more available online through court portals and government initiatives, they remain difficult for non-experts to understand due to dense and unfamiliar legal terminology. By isolating and cataloging legal jargon through a data-driven approach using frequency-based and specificity-based analysis this study aims to improve both human and machine comprehension of legal texts. The resulting glossary of legal terms, complete with contextual examples, makes legal language more accessible to the general public. In a country like India, where the judicial system faces a massive backlog of cases, even small improvements in document retrieval, comprehension, and summarization can make a significant difference in processing legal information more efficiently. Additionally, this glossary can be integrated into legal websites and digital tools, enabling features like hover-tooltips and instant definitions to help users navigate complex legal content effortlessly.

Finally, the legal corpus developed in this project cleaned, tokenized, and tailored to the Malayalam legal domain can be refined and expanded to train machine learning models for legal summarization. By ensuring the corpus captures the linguistic nuances of legal Malayalam and prioritizes key jargon, future AI models can be designed to be more context-aware and precise. This project contributes not only to corpus linguistics but also to the larger mission of democratizing legal information in India. As a result, a structured legal glossary of 100 highly specific terms complete with frequency data and example usage was compiled and exported in JSON and CSV formats. This resource lays the groundwork for better NLP tools in the Malayalam legal domain, creating both immediate and long-term benefits for legal research, public accessibility, and AI-driven legal automation.

1.2 Objectives

The primary objectives of this research are:

- Identify and extract legal jargon unique to Malayalam by analyzing its usage patterns across legal texts.
- Develop a well-structured, domain-specific corpus tailored for training machine learning and NLP models in legal Malayalam, ensuring high-quality linguistic data for computational applications.
- Establish a scalable and replicable methodology for creating specialized linguistic corpora in under-resourced Indian languages, making it easier to expand research and technology to other domains.
- Support legal professionals and researchers by helping streamline document processing and analysis, addressing bottlenecks in the legal system through automated tools and improved accessibility.
- Facilitate the integration of the compiled legal jargon glossary into digital platforms, enabling features like hover-tooltips and instant definitions on legal websites to enhance user comprehension and navigation.

1.3 Research Questions

The key research questions explored in this dissertation are:

1. How to identify and distinguish Malayalam legal language from ordinary language using corpora?
2. What difficulties correspond to the availability and the retrieval of processed data of Malayalam legal texts for computation?
3. What is the significance of corpus linguistics for the development of artificial intelligence-powered legal tools in languages with scant resources?
4. How can documenting and explaining legal terms improve access and the understanding of legal documents for users who do not have relevant expertise?

1.4 Scope and Significance

This study lies at the intersection of corpus linguistics, legal informatics, and computational linguistics, with a special emphasis on the Malayalam language a major Dravidian language spoken primarily in Kerala. The research focuses on constructing a domain-specific corpus by collecting, cleaning, and structuring legal documents from publicly available sources such as legislation, court orders, and official gazettes. Many of these documents exist only in scanned formats with inconsistent structures, OCR errors, and missing metadata, making them difficult to process. To address this, the study applies frequency analysis, n-gram modeling, and comparative techniques to isolate legal jargon terms that appear significantly more often in legal texts than in everyday Malayalam. A structured glossary of high-specificity legal terms, complete with contextual examples and usage frequencies, will serve as a foundational resource for linguistic researchers and NLP practitioners. Furthermore, the refined corpus and extracted jargon aim to support domain-adapted NLP tools, including legal text summarizers and legal information retrieval systems. The project also explores interactive features for legal websites, such as hover-to-explain tooltip systems, improving accessibility for users unfamiliar with complex legal terminology.

The significance of this research extends beyond linguistic analysis it contributes to legal accessibility, technological innovation, and digital justice, particularly in underrepresented Indian languages. While English-language legal NLP has seen steady advancements, regional languages like Malayalam lack comprehensive annotated corpora and domain-specific datasets. By compiling a curated corpus and structured glossary, this study provides crucial resources to support future NLP developments.

Identifying legal jargon is also essential for tasks like document classification, information retrieval, and automated summarization, making this study a foundational step toward robust legal NLP pipelines in Malayalam. Additionally, improving access to legal texts benefits the public by demystifying complex legal language, helping citizens understand their rights and legal processes. The study aligns with broader AI training paradigms, where domain-specific corpora enhance the accuracy of language models in specialized fields. Beyond research, practical applications include improving judicial efficiency by enabling NLP-driven document processing, categorization, and summarization, addressing India’s vast case backlog. The glossary and corpus can also be integrated into legal platforms to support

interactive tools like contextual highlighting and voice-enabled legal assistants, promoting legal tech innovation in the Malayalam language.

CHAPTER 2

LITERATURE REVIEW

Recent advancements in Natural Language Processing (NLP) for the legal domain have highlighted both its transformative potential and its limitations, particularly when applied to multilingual and underrepresented languages like Malayalam. While AI-driven legal tools have made significant strides in high-resource languages such as English and French, legal NLP in Indian regional languages remains underdeveloped due to the scarcity of annotated corpora and domain-specific resources. This study initially set out to build a Malayalam legal text summarizer, but in exploring existing research, it became clear that a more fundamental issue needed addressing the lack of structured linguistic data for Malayalam legal terminology. Without an adequately curated corpus of legal texts, downstream tasks such as automated summarization, classification, and retrieval become unreliable. As a result, this research shifted its focus toward identifying and documenting Malayalam legal jargon through a corpus-based approach, laying the groundwork for future advancements in legal NLP.

A review of existing literature underscores the importance of domain-specific resources in legal language processing. Studies have shown that general-purpose transformer models, such as BERT and GPT, struggle with the syntactic complexity and domain-specific vocabulary of legal texts. To address this, models like LegalBERT and CaseLawBERT were developed, trained specifically on legal corpora. However, these models tend to falter when applied across different legal jurisdictions and multilingual environments, as highlighted by Jain et al. (2021). This issue is particularly relevant in India, where legal texts often incorporate regional linguistic variations, code-mixing, and transliteration. Efforts like InLegalBERT have attempted to fine-tune NLP models for the Indian legal context, but they remain largely limited to English-language legal texts, leaving a gap in resources for languages such as Malayalam.

Legal text summarization, another growing area within legal NLP, further emphasizes the need for structured data in Indian regional languages. Jain et al. (2021) distinguish between extractive and abstractive summarization methods while extractive techniques retain the original wording of a document, abstractive approaches aim to simplify the content into a more readable format. While abstractive summarization holds promise, it faces challenges in

legal contexts due to the risk of introducing inaccuracies, which can have serious consequences in judicial proceedings. The effectiveness of both summarization methods is heavily dependent on high-quality, annotated corpora, something that remains largely absent for Malayalam. Without a well-defined corpus, AI-driven summarization tools cannot accurately process or condense legal texts, reinforcing the urgency of compiling and documenting Malayalam legal jargon.

Beyond legal NLP, research in low-resource language processing sheds light on the broader challenges that regional Indian languages face in AI development. Studies on Indic language processing reveal that multilingual pre-trained models, such as XML-R, often struggle with low-resource languages unless they undergo targeted fine-tuning. Das et al. (2022), in their work on abusive content detection in Indic languages, found that domain-specific adaptation is crucial for ensuring model accuracy. These findings parallel the issues in legal NLP, where models designed for high-resource languages fail to perform effectively in specialized legal registers, particularly when complex terminology and formal syntactic structures are involved. This highlights the necessity of building representative and annotated datasets for regional languages like Malayalam, not just to support legal NLP but also to improve fairness and linguistic inclusivity in AI-driven applications.

Together, these studies paint a clear picture: existing legal NLP models and methodologies cannot be simply transplanted into new linguistic and jurisdictional environments. Indian regional languages, particularly Malayalam, lack the foundational linguistic resources necessary to support sophisticated AI-driven applications. Without a structured legal corpus, any effort to develop advanced NLP tools such as legal text summarization, document classification, or information retrieval would be inherently limited. This realization guided the shift in focus for this research, moving beyond summarization toward a more foundational goal: the identification and documentation of Malayalam legal jargon.

The transition in this study's objectives reflects a practical understanding of what is needed to advance legal NLP in Malayalam. Instead of attempting to build a summarization model on an incomplete dataset, the research now prioritizes corpus construction and jargon identification as essential first steps. By isolating and cataloging legal terminology through frequency analysis and specificity-based comparisons, the study aims to create a structured, domain-specific corpus that can serve as a valuable resource for both linguistic researchers

and AI practitioners. The resulting legal glossary will provide contextual examples of usage, enhancing accessibility for legal professionals, researchers, and citizens alike.

Ultimately, this research contributes to a broader vision one that seeks to democratize legal information in India and address the longstanding linguistic gaps in legal technology. In a country where legal texts often remain inaccessible to the general public, simplifying legal language and improving digital access to legal resources can have meaningful impacts. By bridging the divide between specialized legal registers and everyday Malayalam usage, this study lays the groundwork for practical tools that enhance comprehension and accessibility. In doing so, it aligns with global efforts to create more linguistically inclusive AI systems while addressing local needs in the Indian legal landscape.

CHAPTER 3

METHODOLOGY

3.1 Corpus Collection

The primary source was the official website of the Legislative Department of India, which offers legal documents in various regional languages, including Malayalam. To ensure a diverse and representative dataset, texts were selected from multiple legal domains, such as land rights, disability law, and welfare policies. Instead of focusing solely on one type of legal document, a broad selection was made to capture a more comprehensive range of legal language and terminology in Malayalam.

However, an unexpected challenge arose during this phase the legal texts were available only as scanned PDFs, making them difficult to process. Unlike structured text files, these documents lacked selectable text or machine-readable formats, which meant that direct extraction was not an option. To automate the retrieval process, PyTesseract, an Optical Character Recognition (OCR) tool, was used to convert the scanned images into text. Unfortunately, the results were unreliable due to the complexities of Malayalam script, poor scan quality, and irregular formatting common in official documents. The extracted text contained significant errors, including misrecognized characters and fragmented words, rendering it unsuitable for linguistic analysis.

Given these limitations, the legal corpus had to be transcribed manually. Each document was carefully reviewed and typed out to preserve the accuracy of legal terminology, sentence structures, and contextual meaning. This meticulous process allowed for real-time corrections and ensured the integrity of the dataset, making it suitable for subsequent annotation and analysis. Although this approach was time-intensive, it resulted in a high-quality, error-free corpus laying a strong foundation for identifying and documenting legal jargon in Malayalam.

3.2 Refining Corpus

Once the initial collection of legal texts was completed, the corpus underwent a meticulous refinement process to ensure its relevance and linguistic accuracy. During consultations with

a registrar, it became clear that many of the documents some dating as far back as 1991 contained outdated terminology, phrasing, and orthographic conventions that were no longer in active use in contemporary judicial and administrative settings. Legal language evolves over time, and many of the expressions found in these older texts did not reflect the current norms of Malayalam legal discourse. This posed a challenge, as an unrefined corpus could introduce inconsistencies in computational processing and hinder NLP applications aimed at modern legal texts.

To address this issue, a detailed manual refinement was undertaken. Each document was carefully reviewed to identify archaic spellings, obsolete vocabulary, and inconsistent syntactic patterns. These elements were updated to align with current orthographic standards while preserving the original legal meaning and context. Beyond linguistic accuracy, this step was essential for ensuring that the corpus could be effectively used in computational tasks like Named Entity Recognition (NER), Part-of-Speech (POS) tagging, and legal term extraction. Additionally, this process helped eliminate errors introduced during the manual transcription phase, reducing redundancies and improving overall consistency. The finalized corpus is now a clean, standardized dataset, well-suited for legal NLP research and the systematic documentation of Malayalam legal jargon.

3.3 Preprocessing the Corpus

Once the corpus was refined and standardized, the next crucial step was preprocessing the text to make it suitable for computational analysis. This process was carried out using Python, utilizing standard text-processing libraries to systematically clean and structure the corpus. The primary objective of preprocessing was to transform the raw legal text into a machine-readable format, preparing it for key tasks such as legal jargon identification, frequency analysis, and annotation.

The preprocessing process consisted of the following key steps:

3.3.1 Punctuation Removal

Legal texts often contain intricate punctuation patterns, including semicolons, brackets, and extended clauses. While these punctuation marks play a vital role in legal interpretation, they introduce unnecessary noise during token-based analysis, making computational processing

more challenging. To standardize the text, all punctuation marks were removed, ensuring a cleaner tokenization process and eliminating barriers to effective linguistic analysis.

3.3.2 Whitespace Tokenization

With the refined text free from excessive punctuation, the next step was to tokenize it based on whitespace. This involved splitting the text into discrete word units, using spaces as delimiters. Unlike English, which relies on capitalization cues to mark sentence boundaries, Malayalam does not have capitalization conventions, making whitespace tokenization a simple yet efficient approach for segmenting the text.

3.3.3 List Tokenization and File Generation

Once tokenization was completed, each extracted word was stored as an individual item in a list format. These tokens were then written into a structured text file, ensuring that each word appeared on a separate line. This approach not only facilitated word-level frequency analysis but also provided a structured representation for annotation tasks, allowing for both manual review and automated tagging.

The preprocessing phase ensured that the corpus was transformed into a clean, standardized, and granular format, making it highly suitable for both rule-based and statistical approaches to legal language analysis. By streamlining the text, this step laid the foundation for building a structured lexicon of legal jargon and creating annotations for essential NLP tasks, including entity recognition, phrase chunking, and semantic classification. This well-prepared dataset paves the way for more advanced computational techniques, allowing researchers and developers to explore new possibilities in Malayalam legal NLP.

3.4 Term Frequencies

3.4.1 Term Frequencies and N-gram Analysis:

Once the preprocessing phase was completed, the next crucial step involved computing term frequencies to identify key legal terms particularly those that appear frequently and exhibit high specificity within the Malayalam legal corpus. This step marked the beginning of legal jargon identification, providing the quantitative foundation for further filtering and documentation.

Legal texts rely heavily on domain-specific terminology, shaping the way laws, regulations, and rulings are understood and communicated. By calculating term frequencies, the goal was to pinpoint statistically prominent words that act as indicators of legal jargon. This method helps uncover both commonly used and less frequent terms that are significant in legal discourse but may not be prevalent in general Malayalam usage. Furthermore, legal jargon often consists of multi-word expressions such as *“ഭൂമിയുടെ ഉടമസ്ഥാവകാശം”* (“ownership of land”) rather than standalone words. This makes it critical to incorporate bigram and trigram analysis to capture the structural complexity of legal language.

3.4.2 Token Frequency Calculation

Once the corpus was preprocessed into a list of tokens, a frequency counter was applied using Python’s collections.Counter module. The frequency counter generated an output like:

{'സർക്കാരിന്റെ': 138, 'വൈകല്യങ്ങളുള്ള': 134, 'ആക്ടിന്റെ': 108}

This process allowed for the identification of terms with high absolute frequency, which were then examined as potential candidates for legal jargon. However, frequency alone does not necessarily indicate legal specificity. Some high-frequency words such as *″അനുസരിച്ച്”* (“according to”) serve grammatical functions rather than carrying domain-specific significance.

3.4.3. Bigram and Trigram Extraction:

To capture multi-word legal expressions, the analysis was expanded to include bigrams (two-word sequences) and trigrams (three-word sequences) using the nltk.bigrams() and nltk.trigrams() functions.

The results included:

Bigrams:

[('കേന്ദ്ര സർക്കാർ': 141), ('ഈ വകുപ്പിലെ': 10),

('വ്യാവസായിക പുനർനിർമ്മാണ': 10)]

Trigrams:

[('ഈ ആക്ട് പ്രകാരം': 36), ('കേന്ദ്ര സർക്കാർ ഔദ്യോഗിക': 11),

('രജിസ്റ്റർ ചെയ്ത സംഘടനകളുടെ':9)]

Frequencies of these n-grams were counted and stored. Multi-word expressions often provide deeper insights into the structured nature of legal writing, capturing phrases that convey precise legal meanings.

Bigrams are useful for detecting established legal collocations terms that frequently appear together in official legal writing.

Trigrams help identify formal clauses or technical expressions commonly used in legal statutes and judicial rulings.

These structured linguistic units are essential not only for identifying jargon but also for documenting legal expressions in their natural form. By mapping them systematically, this research contributes to the creation of legal glossaries, enhances NLP model training, and facilitates the development of legal translation tools. Understanding term frequencies and multi-word patterns ensures that legal documents can be effectively processed, making them more accessible both for human readers and AI-driven applications.

3.5 Term Specificity

3.5.1 Term Specificity and Comparison with General Malayalam Corpus

After calculating raw term frequencies within the legal corpus, the next crucial step was to assess term specificity determining how unique or characteristic a term is to legal discourse compared to general Malayalam usage. This step was necessary to ensure that the extracted terms were not merely frequent, but genuinely domain-specific, qualifying them as legal jargon rather than everyday language.

3.5.2 Selection of General Malayalam Corpus

To conduct a meaningful comparison, a general Malayalam corpus was needed as a reference point. The selected corpus was sourced from the GitHub repository titled "Malayalam Corpus" maintained by Swathanthra Malayalam Computing. This open-access linguistic resource comprises a diverse range of Malayalam texts, primarily drawn from Wikipedia articles and general-purpose documents.

A random subset of the corpus was selected to ensure broad representation across various topics, including culture, science, politics, and daily life. This allowed for a well-rounded linguistic baseline against which legal term frequencies could be evaluated, ensuring that identified legal jargon was distinctly characteristic of legal texts rather than common Malayalam usage.

3.5.3 Calculating Term Specificity

The key objective of this step was to measure how much more frequently a term appears in legal texts compared to general Malayalam texts. Term specificity was calculated using a ratio-based approach, leveraging normalized frequencies:

Specificity Score=(freqlegal(t)Nlegal)(freqgeneral(t)+1Ngeneral)\text{Specificity Score} = \frac{\left(\frac{\text{freq}_\text{legal}(t)}{N_\text{legal}}\right)}{\left(\frac{\text{freq}_\text{general}(t) + 1}{N_\text{general}}\right)}

Where:

- freqlegal(t) = The frequency of term t in the legal corpus.
- Nlegal = The total number of tokens in the legal corpus.
- freqgeneral(t) = The frequency of term t in the general Malayalam corpus.
- Ngeneral = The total number of tokens in the general corpus.
- The "+1" in the denominator ensures that terms absent in the general corpus do not create division-by-zero errors.

A high specificity score indicated that a term was significantly more common in legal discourse than in general Malayalam usage, making it a strong candidate for inclusion in the legal jargon glossary.

3.5.4 Filtering and Ranking of Jargon Terms

Once specificity scores were computed for all terms in the legal corpus, they were ranked in descending order based on specificity. However, frequency alone was not the sole criterion for selection. To ensure practical relevance and statistical reliability, a minimum occurrence threshold was applied. Only terms that appeared at least five times in the legal corpus were retained, preventing the inclusion of rare or potentially anomalous entries.

The final filtered and ranked list of terms represented the top candidates for legal jargon, balancing high domain specificity with reasonable frequency. These terms then underwent further manual inspection, contextual analysis, and structured documentation in the Malayalam Legal Jargon Glossary, contributing to a foundational linguistic resource for legal NLP applications and legal text accessibility.

3.6 Manual Review

Once the top candidate terms were identified using frequency and specificity metrics, the next critical phase involved manual review and expert validation. This step ensured that the computationally extracted terms truly represented domain-specific legal jargon rather than coincidental or misleading patterns. To achieve this, the validation process was divided into two distinct rounds preliminary filtering by the researcher and a detailed expert review conducted by a practicing advocate.

The first stage of manual validation was carried out by the researcher, focusing on refining the initial list of jargon terms. This involved scanning through the top-ranked words and multi-word expressions (unigrams, bigrams, and trigrams) using specificity scores and minimum frequency thresholds as guiding criteria. During this process, obviously non-legal terms, commonly occurring stopwords, grammatical fragments, and generic collocations were systematically filtered out.

For example, words such as **"അതിനാൽ"** ("therefore") and **"സമയത്ത്"** ("at the time") frequently appeared in legal documents but lacked meaningful legal relevance. Their inflated

specificity scores were primarily due to corpus imbalance rather than any true significance in legal discourse. Removing such terms allowed the study to focus on genuinely domain-specific words, ensuring a cleaner and more representative jargon set.

To validate the accuracy and contextual appropriateness of the legal jargon, a second round of manual filtering was conducted in collaboration with a practicing advocate. This phase was essential for refining the list, ensuring its precision, and confirming that each term was integral to legal discourse.

The expert review addressed several key areas:

1. **Disambiguation of Terms** – Some words, while appearing frequently in legal texts, carry different meanings in everyday language. The advocate helped clarify ambiguous terms within the legal framework. For instance, **"അഭിപ്രായം"** ("opinion") has general usage but holds a specific meaning in judgments and tribunal recommendations.
2. **Validation of Legal Salience** – The expert assessed whether the identified terms were not only present in legal documents but also essential in legal reasoning, procedural language, and statutory framing.
3. **Identification of Omissions and Variants** – The review also helped identify overlooked variations or compound forms of key legal terms. Due to preprocessing constraints, some inflected words or embedded legal constructs were initially missed. The advocate's input ensured that critical terminology was included in the final list.

This two-tier validation process was more than just a corrective measure it was a fundamental methodological step in ensuring the integrity of the corpus-based approach. NLP models, particularly when applied to low-resource languages like Malayalam, often struggle with semantic nuance and contextual accuracy. By incorporating human expertise into the validation process, the study introduced a human-in-the-loop mechanism, bridging the gap between algorithmic detection and real-world legal relevance.

3.7 Context for Jargon Terms

Following the identification and expert validation of legal jargon terms, the next critical step was extracting contextual information for each selected term. This process served two essential purposes: first, ensuring that each term was accurately situated within its legal discourse environment, and second, documenting example contexts that could support linguistic research, legal education, and computational applications. Whether for the development of glossaries, NLP datasets, or legal literacy tools, capturing the surrounding text helps clarify the precise function of legal jargon in real-world usage.

Legal language is deeply context-dependent the meaning and function of a term are shaped by its collocational patterns, syntactic positioning, and the legal principles governing its use.

Without context, even statistically significant words can be ambiguous or misleading. For example, the term "അവകാശം" ("right/claim") may refer to a legal entitlement, a property right, or even a moral claim, depending on its surrounding text.

By extracting surrounding passages, this step enhances:

- Disambiguation of terms with multiple meanings – Helping differentiate legal technicalities from general language use.
- Recognition of typical syntactic structures or legal clause patterns – Capturing legal writing conventions for better analysis.
- Understanding of jargon in actual legal judgments, statutes, or tribunal orders – Strengthening applied legal NLP models and education tools.

3.8 Formatted Jargon Glossary

The culmination of the methodological workflow was the creation of a structured and accessible glossary of validated Malayalam legal jargon terms. This output serves as a primary resource for linguistic analysis, NLP applications, and educational or legal reference. The glossary not only compiles key legal terms but also presents them in a richly annotated, searchable, and standardized format.

Output Structure and Design Each entry in the glossary consists of the following elements:

- Legal Term (Unigram, Bigram, or Trigram): The core term identified as legal jargon.
- Specificity Score: A numerical indicator of the term's exclusivity to the legal domain relative to general Malayalam usage.
- Token Frequency: The number of times the term appeared in the legal corpus, serving as a marker of its usage prominence.
- Example Context(s): One or more real excerpts from legal documents that demonstrate the term’s practical application and usage.

This data was formatted in both JSON and CSV formats to maximize its utility across different audiences and platforms: JSON format supports integration into NLP pipelines or software tools for annotation, classification, or search functionalities. CSV format provides easy access for manual inspection, educational use, or integration into spreadsheets for comparative linguistic research. Implementation and Documentation Python scripts were employed to export the reviewed and structured data into these formats. A sample CSV row includes: Additionally, metadata such as date of extraction, reviewer details, and corpus source was included in the accompanying documentation to ensure transparency and reproducibility.

The formatted glossary is more than a linguistic artifact—it is a functional bridge between legal language and computational analysis. By documenting Malayalam legal jargon in this structured way, the project provides a foundational resource for: Developing legal text summarizers and classifiers. Building terminology databases for regional law digitization efforts. Supporting language technology development in underrepresented linguistic domains. Moreover, the glossary serves a critical social purpose: enhancing the accessibility and comprehensibility of legal language for non-experts and native Malayalam speakers navigating complex legal documentation.

CHAPTER 4

RESULT

4.1 Statistical Distinctiveness and Overrepresentation of High-Specificity Malayalam Legal Terms

Term specificity identifies a distinct set of Malayalam terms demonstrating high statistical specificity (S) within the legal corpus. This specificity stems from their significant *overrepresentation* in legal texts relative to general language usage, indicating their crucial role in constructing legal meaning. The following provides an analysis of the contextual and functional factors contributing to the high specificity scores for the most prominent terms:

1. വകുപ്പ് *(Vakuppu - Department)* – S=246.29
   In legal and administrative contexts, വകുപ്പ് is the standard term for designating government departments overseeing specific domains (e.g., സാമൂഹ്യനീതിവകുപ്പ് - Social Justice Department, ആരോഗ്യവകുപ്പ് - Health Department). Its frequent and formal invocation within official documents contrasts sharply with its minimal use in everyday conversation or general writing.

   This term carries a high degree of formality and is typically embedded within institutional or bureaucratic discourse, often collocating with hierarchical terminology.

2. സർക്കാർ *(Sarkkār - Government)* – S=242.30
   As the primary legislative and executive actor, "government" is semantically central to legal language. It appears ubiquitously in phrases denoting government actions, objections, or procedures (e.g., സർക്കാർ ആക്ഷേപം - government objection, സർക്കാർ നടപടികൾ - government procedures). Laws are formulated, enacted, and enforced by the government, necessitating its frequent mention.

   While present in general Malayalam, its density and frequency are significantly lower in non-legal genres like fiction or informal communication.

3. കേന്ദ്ര *(Kēndra - Central)* – S=236.97
   Primarily refers to the *Central Government* (കേന്ദ്ര സർക്കാർ) or associated central authorities. Within India's federal structure, legal texts frequently delineate

jurisdictional boundaries between central (കേന്ദ്ര) and state (സംസ്ഥാന) powers, leading to the term's prevalence.

Often appears alongside terms signifying legal instruments or control, such as നിയമം (law), നിയന്ത്രണം (regulation), or ആക്ട് (Act).

4. സംസ്ഥാന *(Samsthāna - State)* – S=229.65
   Functions as the counterpart to കേന്ദ്ര *(Central)*, appearing frequently in contexts related to state-level governance (e.g., സംസ്ഥാന സർക്കാരിന്റെ ഉത്തരവ് - State Government's order, സംസ്ഥാന നിയമസഭ - State Legislative Assembly). Legal discourse often addresses state-specific laws, provisions, and agencies.

   Crucial for marking the federal division of legal powers and responsibilities within India.

5. രജിസ്റ്റർ *(Rejiṣṭar - Register)* – S=173.73
   This is a technical term essential for legal documentation concerning formal records, such as property titles (ഭൂമിയുടെ രജിസ്റ്റർ - land register), vital statistics (births/deaths), or registered entities (സംഘങ്ങളുടെ രജിസ്റ്റർ - register of societies). Its specialized function makes it rare in casual Malayalam but vital for legal record-keeping.

   Indicates formal enrollment, documentation, or official listing within a legal or administrative framework.

6. ആക്ട് *(Ākṭ - Act)* – S=172.40
   A direct transliteration referring to specific statutes or pieces of legislation (e.g., *Disabilities Act*, *Land Reform Act*). It is often retained in its English form within Malayalam legal texts due to its codified status and to avoid ambiguity, ensuring precise reference to specific laws.

   Its status as an untranslated loanword, often capitalized and used in formal citations, marks it as distinctively legal or administrative terminology.

7. ഭൂവിവരണ *(Bhūvivaraṇa - Land Assignment/Settlement)* – S=133.80
   A specialized compound term used almost exclusively within the domain of land law, particularly concerning land assignment, settlement, or related records (e.g.,

ഭൂവിവരണ ആക്ട് - Land Assignment Act). Its highly specific application makes it virtually absent from general discourse.

The compound structure itself (*bhūmi* - land + *vivaraṇa* - description/assignment) signals a specialized administrative or legal action related to land.

8. ഉപവകുപ്പ് *(Upavakuppu - Sub-department/Subdivision)* – S=107.83

   This term denotes finer levels within administrative or departmental hierarchies (e.g., ഭൗമശാസ്ത്ര ഉപവകുപ്പ് - Geology Subdivision). Its usage is confined to formal, bureaucratic, or organizational contexts, making it rare outside official documentation.

   Facilitates detailed classification and organizational structure within legal and administrative frameworks.

4.2 Syntactic Structures

Beyond just specific words, the way Malayalam builds sentences and modifies terms plays a key role in legal texts. 7 notable grammatical features were observed in the high-specificity terms, showing how the language adapts for legal precision and efficiency:

- **The Power of Compound Nouns**
  Think of words like ഭൂവിവരണ (land classification/description) or കേന്ദ്രസർക്കാർ (Central Government) as efficient linguistic packages. Legal Malayalam often joins smaller nouns together to create these *compound nouns*. Why? It's a neat way to label complex legal or administrative concepts very specifically without needing lengthy descriptive phrases. Syntactically, they act as a single unit, making sentences cleaner and more direct – a valuable trait when clarity is paramount.

- **The "-ന്റെ" (-nṟe) Possessive Marker**
  Who does this rule belong to? Whose decision is this? Legal texts constantly need to clarify ownership or authority. The small but mighty -ന്റെ suffix is key here. Adding it to nouns like സർക്കാർ (government) to get സർക്കാരിന്റെ (of the government/government's) clearly flags possession, responsibility, or the source of

authority. This *genitive construction* is a grammatical workhorse in legal drafting for assigning jurisdiction or ownership.

- **Describing Categories Neatly: Adjectives from Verbs (-ഉള്ള -uḷḷa)**
  How do you efficiently describe groups like "persons with disabilities" (വൈകല്യങ്ങളുള്ള വ്യക്തികൾ)? Malayalam often uses adjective-like modifiers ending in -ഉള്ള (*-uḷḷa*), derived from verbs. It's like adding a descriptive tag ("having X" or "with X") directly onto the noun. This *adjective-participial construction* keeps sentences compact while clearly defining the categories of people or things that laws often need to address.

- **Marking Completed Actions: The Role of "ചെയ്ത" (ceyta - done)**
  Legal language frequently needs to refer to actions already completed. The past participle ചെയ്ത (done/performed) is crucial for this. It can describe something based on a past action (like ചെയ്ത പ്രവർത്തനം – the action that was done) or link actions in sequence. Its power lies in precisely and concisely indicating completed actions, essential when establishing facts or consequences in legal contexts.

- **Grammar Stacking: Efficient Inflections**
  Malayalam can pack a lot of grammatical information into a single word. Take വ്യവസ്ഥകൾക്ക് (vyavasthakaḷkkŭ - to the regulations). This single word tells you it's plural (-കൾ -kaḷ) *and* that it's the recipient or target of an action (dative case, -ക്ക് -kkŭ). This ability to *stack multiple inflections* onto one word makes the language very efficient, conveying complex relationships without extra words – perfect for the precise and often dense nature of legal writing.

- **Saying "Where" Concisely: Location Markers (-ൽ -l)**
  Where exactly *in* the document or *within* which department does a rule apply? The locative suffix -ൽ (-l), meaning 'in' or 'at', often attaches directly to nouns (e.g.,

ഉപവകുപ്പിൽ - upavakuppil, 'in the subsection'). Instead of needing separate prepositions like in English, Malayalam frequently integrates this positional or jurisdictional information right into the noun itself, efficiently pinpointing location within texts or structures.

- **Expressing "Must Do": Verbs of Obligation**
  Laws are full of obligations – things that *must* or *should* be done. Legal Malayalam clearly signals these duties using specific *compound verb phrases*, often involving forms like -േണ്ട (-ēṇṭa) combined with auxiliaries. For instance, ചെയ്യേണ്ടതാണ് (ceyyēṇṭatāṇŭ) translates roughly to "it must be done" or "it is to be done." This structure, expressing *deontic modality* (the language of obligation/permission), is fundamental for unambiguously stating requirements and mandates in statutes.

4.3 Corpus and Orthographic Observations

4.3.1 Orthographic Differences

Feature	Original Text	Refined Text	Analysis
Quotation marks	Mismatched closing quote in phrases like “പ്രാധിക്യത തർജ്ജമകൾ"	Preserved without correction	Still visually distracting; future versions could correct it.
Spacing	Inconsistent spacing (e.g., "1ാം നമ്പരായി ഒരു പുറത്ത്")	Corrected to single spacing: "1ാം നമ്പരായി ഒരു പുറത്ത്"	Enhances visual clarity and reduces distraction during reading.
Zero-width joiners	Used in compound forms (e.g., “പ്രസിദ്ധീകരിക്കുകയും”)	Maintained	Retains original rendering of conjunct characters.

4.3.2 Linguistic Differences

Feature	Original Text	Refined Text	Analysis
Syntax modernization	Complex sentence structures	Mild simplification in some clauses	Improves rhythm and flow subtly, especially for modern readers.
Clarity in enumeration	Dense paragraph format	Cleaner spacing, better indentation	Makes legal clauses easier to identify and comprehend.
Passive voice	Commonplace (e.g., “പ്രസിദ്ധീകരിക്കുകയും”)	Retained	Consistent with legal conventions; no change needed.

The refined version improves the structural presentation of the document while also altering its legal language. Few lexical simplification is attempted; but terms like *“ആക്റ്റ്”* and *“തർജ്ജമകൾ”* are preserved as-is. Hence, it is not a “plain language” rewrite.Legal terminology and document authenticity are fully maintained.

CHAPTER 5

CHALLENGES AND LIMITATIONS

5.1 Limited Access to High-Quality Datasets

One of the major challenges encountered in this research was the scarcity of publicly available, well-structured Malayalam legal datasets. Unlike English, which benefits from abundant digitized legal resources, Malayalam suffers from a significant resource gap in the domain of Natural Language Processing (NLP), especially in specialized areas such as law. Many legal texts remain inaccessible due to paywalls, exist only in scanned formats (such as PDFs), or are fragmented across various sources without a standardized structure. These limitations restricted linguistic diversity within the corpus, making it difficult to achieve broad generalizability in the findings.

5.2. Antiquated and Archaic Language in Source Texts

A substantial portion of legal documents used in this study particularly legislative acts and government notifications were drafted several decades ago, often prior to modern linguistic standardization. These documents frequently contain archaic grammatical structures, outdated terminology, and phrasing styles that are no longer in active use. This posed a complex challenge in linguistic refinement, as modern equivalents for certain legal expressions were either nonexistent or difficult to introduce without altering legal accuracy. While readability improvements were implemented, extensive modernization was avoided to prevent misinterpretation of legal intent.

5.3. Requirement of Legal Domain Knowledge

Legal text refinement requires not only linguistic expertise but also functional legal literacy. Many Malayalam legal terms are highly domain-specific and deeply contextual, making interpretation difficult without specialized legal knowledge. Although efforts were made to validate accuracy through iterative checks, the absence of formal legal training introduced a risk of misclassification, particularly when refining syntactic structures or deciding which jargon terms should be included or excluded.

5.4. Orthographic Complexity and Digital Formatting Challenges

Malayalam's script presents unique orthographic complexities, which are further magnified in digital processing. During corpus refinement, inconsistencies in diacritic placement, conjunct

characters, and zero-width joiners (ZWJs) frequently emerged. These elements do not always render uniformly across different platforms, leading to issues in NLP-based tokenization. Even minor typographic variations can falsely split words or skew frequency counts, making text standardization particularly challenging.

5.5. Limitations of Tokenization and NLP Tools for Malayalam

Existing NLP tools for Malayalam, while useful for basic text processing, struggle with agglutinative and morphologically rich languages like Malayalam. Many available tools rely on rule-based systems, which do not generalize well to legal discourse, leading to constraints in sentence segmentation, lemmatization, and named entity recognition (NER). Open-source linguistic tools for Malayalam remain limited in scope, restricting the ability to implement advanced NLP techniques, such as syntactic parsing or POS-tagging, which could have enhanced jargon identification and corpus refinement.

5.6. Lack of Parallel Corpora for Validation or Translation

The absence of a parallel legal corpus (Malayalam–English or Malayalam–Plain Language) made cross-linguistic validation difficult. Such corpora are valuable for assessing semantic equivalence, ensuring that refined terms maintain their legal intent across languages. Without these references, accuracy assessment relied solely on manual linguistic judgment, rather than quantifiable metrics or translation-based validation.

5.7. Subjectivity in Defining and Filtering Legal Jargon

Legal terminology exists on a continuum between general language and domain-specific jargon, making categorization inherently subjective. While this study employed computationally rigorous specificity scoring, results were influenced by threshold settings

and corpus composition. Additionally, manual filtering of top terms introduced a degree of subjectivity, particularly when deciding whether a term should be classified as "legal jargon" or "general formal usage."

CHAPTER 6

CONCLUSION

This dissertation set out to tackle a critical gap at the intersection of computational linguistics, corpus linguistics, and legal informatics, with a specific focus on low-resource

Indian languages particularly Malayalam. By refining and analyzing legal texts, this research contributes both theoretical insights and practical advancements to the emerging field of legal Natural Language Processing (NLP) in Indian-language contexts.

At its core, this study pursued two key objectives:

1. To construct a refined, annotated corpus of Malayalam legal documents through systematic linguistic and orthographic editing.
2. To develop a glossary of domain-specific legal jargon, providing a foundational resource for future legal text summarization, translation, and layperson comprehension.

This was accomplished by curating publicly available legal texts, identifying and correcting orthographic inconsistencies, encoding issues like diacritic mismatches, and zero-width joiners, and applying both linguistic expertise and frequency-based metrics to extract relevant terminology. The resulting corpus successfully preserves legal authenticity while enhancing structural and typographic clarity an essential balance in legal communication, where accuracy is paramount.

6.1 Major Contributions of the Study

One of the most significant contributions of this research is the documentation of the orthographic, lexical, and stylistic characteristics of Malayalam legal texts. These observations provide both a descriptive and prescriptive model for digital editing of similar low-resource corpora. Furthermore, by cataloging and statistically validating legal jargon using specificity scores, this work establishes the building blocks for domain-specific NLP applications, including:

- Hovertip-enabled legal websites that simplify complex legal terminology for general users.
- Assistive language tools for judicial services, improving accessibility for legal professionals.
- Semantic simplification mechanisms for citizen-facing legal portals, making legal documents easier to understand.

6.2 Corpus Linguistics and Computational Processing

Situated within the broader context of corpus linguistics and computational text processing, this dissertation reinforces the importance of corpus-based approaches in training accurate, ethical AI models especially in sensitive domains such as law.

The rise of digitized legal records in India presents both a unique opportunity and a serious challenge. While vast amounts of legal data are now accessible, much of it remains incomprehensible to the general public due to dense legal jargon and inconsistent formatting. This research argues that refining and structuring legal data is not merely an editorial task but rather a critical prerequisite for downstream NLP tasks, including:

- Legal text summarization
- Machine translation
- Question-answering systems for legal research

6.3 Democratization of Legal Information Through Language

A core argument of this dissertation is the democratization of legal information through computational linguistics. In countries like India where judicial backlogs are extensive, and legal literacy varies widely enhancing legal accessibility can directly contribute to greater procedural transparency and public empowerment.

Although the glossary compiled in this study is limited in scope, it introduces a scalable model for integrating computational methods with legal annotation. Practical applications, such as hover-tooltips on legal websites, could significantly improve comprehension and usability for non-experts trying to navigate complex legal documentation.

Despite its contributions, this research acknowledges several limitations, including:

- Restricted availability of legal data
- Absence of parallel corpora for validation
- Challenges in modernizing archaic legal language
- Orthographic complexities that complicate digital standardization

These challenges highlight the need for sustained interdisciplinary collaboration. Moving forward, computational linguists, legal professionals, software developers, and policymakers must work together to expand and refine tools that make legal texts more accessible.

6.4 Future Directions and Impact

This dissertation lays the groundwork for the computational processing of Malayalam legal texts, bridging gaps between raw digitized data and structured legal corpora. It offers new possibilities for research and application in:

- Indian-language NLP
- Legal technology and AI-driven document processing
- Civic accessibility through better legal language tools

Future research could build upon this study by scaling the corpus, creating annotated datasets for supervised learning, and leveraging computational insights to develop fully functional legal summarization tools helping both legal professionals and the public engage with legal texts more effectively.

By addressing these key challenges and expanding interdisciplinary efforts, the work initiated here has the potential to transform legal accessibility, NLP advancements, and AI-driven applications in regional Indian languages.

REFERENCES

- Das, M., Banerjee, S., & Mukherjee, A. (2022). Data bootstrapping approaches to improve low resource abusive language detection for Indic languages. arXiv. https://arxiv.org/abs/2204.12543

- Jain, D., Borah, M. D., & Biswas, A. (2021). Summarization of legal documents: Where are we now and the way forward. Computer Science Review, 40, 100388. https://doi.org/10.1016/j.cosrev.2021.100388
- Paul, S., Mandal, A., Goyal, P., & Ghosh, S. (2023). Pre-trained language models for the legal domain: A case study on Indian law. Unpublished manuscript. Indian Institute of Technology Kharagpur. https://doi.org/10.1145/nnnnnnn.nnnnnnn
- Katz, D. M., Bommarito, M. J. II, & Blackman, J. (2016). A general approach for predicting the behavior of the Supreme Court of the United States. PLOS ONE, 12(4), e0174698. https://doi.org/10.1371/journal.pone.0174698
- Trivedi, P., Jain, D., Gite, S., Kotecha, K., Bhatt, A., & Naik, N. (2024). Indian Legal Corpus (ILC): A dataset summarizing Indian legal proceedings using natural language. *Engineered Science, 27*, Article 1022. https://doi.org/10.30919/es1022
- Bhatia, V. K., Langton, N. M., & Lung, J. (2004). Legal discourse: Opportunities and threats for corpus linguistics. In U. Connor & T. A. Upton (Eds.), *Discourse in the professions: Perspectives from corpus linguistics* (pp. 203–231). Amsterdam: John Benjamins.
- Legislative Department. (n.d.). *Malayalam legislative documents*. Government of India. Retrieved April 30, 2025, from https://legislative.gov.in/malayalam/
- Swathanthra Malayalam Computing. (n.d.). *SMC corpus* [GitHub repository]. GitHub. Retrieved April 30, 2025, from https://github.com/smc/corpus

APPENDICES

Excerpts from the original corpus :-

“ 1988 മാർച്ച് 1-ാം തീയതി നിലവിലിരുന്ന പ്രകാരമുള്ള 1980-ലെ കരിഞ്ചന്ത തടയലും അവശ്യസാധനങ്ങളുടെ വിതരണം നിലനിറുത്തലും ആക്റ്റിന്റെ (1980-ലെ 7-ാം ആക്റ്റ്) മലയാളത്തിലെ ആധികാരിക പരിഭാഷ ഈ പതിപ്പിൽ അടങ്ങിയിരിക്കുന്നു. 1988 ആഗസ്റ്റ് 2-ാം തീയതിയിലെ അസാധാരണ കേന്ദ്രഗസറ്റ് XI-ാം ഭാഗം 1-ാം വകുപ്പ്, 1-ാം വാല്യം 1-ാം നമ്പരായി 26 മുതൽ 31 വരെയുള്ള പുറങ്ങളിൽ ഇത് പ്രസിദ്ധീകരിച്ചിരുന്നു.

മലയാളത്തിലെ ഈ ആധികാരിക പരിഭാഷ 1973-ലെ ആധികാരികപാഠം (കേന്ദ്രനിയമങ്ങൾ) അക്റ്റ് 2-ാം വകുപ്പ് (ക) ഖണ്ഡം അനുസരിച്ച് പ്രസിഡൻറ" അധികാരപ്പെടുത്തിയ പ്രകാരം പ്രസിദ്ധീകരിക്കുകയും അങ്ങനെ പ്രൈസിദ്ധിക്കരിച്ചതിന്മേൽ അത് ആ ആക്ടിന്റെ മലയാളത്തിലെ ആധികാരിക പരിഭാഷയായിത്തീരുകയും ചെയ്തു.

കരിഞ്ചന്ത തടയുകയും സമൂഹത്തിൽ ആവശ്യമായ സാധനങ്ങളുടെ വിതരണം നിലനിറുത്തുകയും ചെയ്യുന്ന ആവശ്യത്തിലേക്ക് ചില സംഗതികളിൽ തടങ്കൽ നൽകുന്നതിനും അതുമായി ബന്ധപ്പെട്ട വിഷയങ്ങൾക്കും വേണ്ടിയുള്ള ഒരു ആക്റ്റ് ”.

Excerpts from refined corpus:-

“ 1988 മാർച്ച് 1-ാം തീയതി നിലവിലുള്ള പ്രകാരമുള്ള 1980-ലെ കരിഞ്ചന്ത തടയലും അവശ്യസാധനങ്ങളുടെ വിതരണം നിലനിറുത്തലും ആക്ടിന്റെ (1980-ലെ 7-ാം ആക്ട്) മലയാളത്തിലെ ആധികാരിക പരിഭാഷ ഈ പതിപ്പിൽ അടങ്ങിയിരിക്കുന്നു. 1988 ആഗസ്റ്റ് 2-ാം തീയതിയിലെ അസാധാരണ കേന്ദ്ര ഗസറ്റ് XI-ാം ഭാഗം 1-ാം വകുപ്പ്, 1-ാം വാല്യം 1-ാം നമ്പരായി 26 മുതൽ 31 വരെയുള്ള പുറങ്ങളിൽ ഇത് പ്രസിദ്ധീകരിച്ചിരുന്നു.മലയാളത്തിലെ ഈ ആധികാരിക പരിഭാഷ 1973-ലെ ആധികാരികപാഠം (കേന്ദ്രനിയമങ്ങൾ) ആക്ട് 2-ാം വകുപ്പ് (ക) ഖണ്ഡം അനുസരിച്ച് പ്രസിഡൻറ് അധികാരപ്പെടുത്തിയ പ്രകാരം പ്രസിദ്ധീകരിക്കുകയും അങ്ങനെ പ്രസിദ്ധീകരിച്ചതിനാൽ അത് ആ ആക്ടിന്റെ മലയാളത്തിലെ ആധികാരിക പരിഭാഷയായിത്തീരുകയും ചെയ്തു.കരിഞ്ചന്ത തടയലും സമൂഹത്തിൽ ആവശ്യമായ സാധനങ്ങളുടെ വിതരണം നിലനിറുത്തലും സംബന്ധിച്ച ആവശ്യകതകൾ, ചില സന്ദർഭങ്ങളിൽ തടസ്സം വിതയ്ക്കുകയും അതുമായി ബന്ധപ്പെട്ട വിഷയങ്ങൾക്കുള്ള പരിഹാരങ്ങൾക്കായി ഒരു ആക്ട്.

ഭാരത റിപ്പബ്ലിക്കിന്റെ മുപ്പത്തിഒന്നാം സംവത്സരത്തിൽ പാർലമെന്റ് താഴെ പറയുന്ന പ്രകാരം നിയമം നിർമ്മിച്ചിട്ടുണ്ട്:

(1) ഈ ആക്ടിന് 1980-ലെ കരിഞ്ചന്ത തടയലും അവശ്യസാധനങ്ങളുടെ വിതരണം നിലനിറുത്തലും ആക്ട് എന്ന് പേരുപറയാം.

ചുരുക്കപ്പേരും വ്യാപ്തിയും പ്രാരംഭവും

(2) ജമ്മു-കാശ്മീർ സംസ്ഥാനം ഒഴികെ 2000 മുഴുവൻ ഇതിന് വ്യാപ്തി ഉണ്ടായിരിക്കും.

(3) ഇത് 1979 ഒക്ടോബർ 5-ാം തീയതി പ്രാബല്യത്തിൽ വന്നതായി കരുതപ്പെടുന്നു ".

Codes used :-

In [ ]:
```python
import re

def tokenize_malayalam_text(input_file, output_file):
    # Read the corpus
    with open(input_file, 'r', encoding='utf-8') as f:
        text = f.read()

    # Standardize case for English characters (Malayalam doesn't have case)
    text = text.lower()

    # Remove punctuation (keep only Malayalam letters, English letters, numbers, and spaces)
    text = re.sub(r'[^\u0D00-\u0D7F\sa-zA-Z0-9]', '', text)

    # Tokenize by whitespace
    tokens = text.split()

    # Write tokens to output file, one token per line
    with open(output_file, 'w', encoding='utf-8') as f:
        for token in tokens:
            f.write(token + '\n')

# Example usage:
tokenize_malayalam_text('Final1.1.txt', 'Final1.2.txt')
```

In [ ]:
```python
import re
from collections import Counter

# Function to clean and tokenize text
def preprocess_malayalam(text):
    # Remove unwanted characters, normalize spaces
    text = re.sub(r'[०-९]+', '', text)  # Remove Devanagari digits
    text = re.sub(r'\s+', ' ', text).strip()  # Normalize spaces
    tokens = text.split()  # Simple whitespace tokenization
    return tokens

# Process the specific file
legal_tokens = []

input_file = 'Final1.2.txt'

with open(input_file, 'r', encoding='utf-8') as file:
    text = file.read()
    tokens = preprocess_malayalam(text)
    legal_tokens.extend(tokens)

# Count frequencies
legal_freq = Counter(legal_tokens)
```

3.

```
import re
from collections import Counter


# Function to clean and tokenize text
def preprocess_malayalam(text):
    # Remove unwanted characters, normalize spaces
    text = re.sub(r'[०-९]+', '', text)  # Remove Devanagari digits
    text = re.sub(r'\s+', ' ', text).strip()  # Normalize spaces
    tokens = text.split()  # Simple whitespace tokenization
    return tokens


# Process the specific file
legal_tokens = []


input_file = 'Final1.2.txt'


with open(input_file, 'r', encoding='utf-8') as file:
    text = file.read()
    tokens = preprocess_malayalam(text)
    legal_tokens.extend(tokens)
```

```
# Count frequencies
legal_freq = Counter(legal_tokens)

# Save the frequencies into a txt file
with open('token_frequencies.txt', 'w', encoding='utf-8') as f:
    for token, freq in legal_freq.items():
        f.write(f"{token}\t{freq}\n")
```

4.

```
from collections import Counter
from nltk import bigrams, trigrams

# Read tokens from file (already tokenized)
input_file = 'Final1.2.txt'
legal_tokens = []

with open(input_file, 'r', encoding='utf-8') as file:
    for line in file:
        token = line.strip()
```

```
        if token:  # skip empty lines
            legal_tokens.append(token)


# --- WORD FREQUENCY ---
word_freq = Counter(legal_tokens)


# Save word frequencies
with open('token_frequencies.txt', 'w', encoding='utf-8') as f:
    for token, freq in word_freq.most_common():
        f.write(f"{token}\t{freq}\n")


# --- BIGRAMS ---
bi_grams = list(bigrams(legal_tokens))
bigram_freq = Counter(bi_grams)


# Save bigram frequencies
with open('bigram_frequencies.txt', 'w', encoding='utf-8') as f:
    for pair, freq in bigram_freq.most_common():
        bigram_text = ' '.join(pair)
        f.write(f"{bigram_text}\t{freq}\n")
```

```
# --- TRIGRAMS ---

tri_grams = list(trigrams(legal_tokens))

trigram_freq = Counter(tri_grams)


# Save trigram frequencies

with open('trigram_frequencies.txt', 'w', encoding='utf-8') as f:

    for triplet, freq in trigram_freq.most_common():

        trigram_text = ' '.join(triplet)

        f.write(f"{trigram_text}\t{freq}\n")
```

5.

```
from collections import Counter

import re

def preprocess_malayalam(text):

    # Example: simple preprocessing

    text = re.sub(r'[^\w\s]', '', text)  # Remove punctuation

    text = text.lower()  # Convert to lowercase

    tokens = text.split()  # Split into tokens

    return tokens
```

```
# Read and preprocess general corpus
with open('General.txt', 'r', encoding='utf-8') as file:
    text = file.read()
    general_tokens = preprocess_malayalam(text)

# Read preprocessed legal corpus
with open('Final1.2.txt', 'r', encoding='utf-8') as file:
    legal_tokens = file.read().split()  # Assuming tokens are space-separated

# Create frequency counters
general_freq = Counter(general_tokens)
legal_freq = Counter(legal_tokens)

# Calculate term specificity
jargon_scores = {}
for term, legal_count in legal_freq.items():
    general_count = general_freq.get(term, 0) + 1  # Add 1 to avoid division by zero
    specificity = (legal_count / len(legal_tokens)) / (general_count / len(general_tokens))
    jargon_scores[term] = specificity

# Write term specificity scores to a file
```

```
with open('term_specificity.txt', 'w', encoding='utf-8') as f:
    for term, score in sorted(jargon_scores.items(), key=lambda x: x[1], reverse=True):
        f.write(f"{term}\t{score:.6f}\n")


print("Term specificity scores saved to 'term_specificity.txt'")
```

6.

```
# Step 4: Identify Legal Jargon (file-based version)


# 1. Load term specificity scores from 'term_specificity.txt'
jargon_scores = {}
with open('term_specificity.txt', 'r', encoding='utf-8') as f:
    for line in f:
        parts = line.strip().split('\t')
        if len(parts) == 2:
            term, score = parts
            jargon_scores[term] = float(score)


# 2. Load legal token frequencies from 'token_frequencies.txt'
```

```
legal_freq = {}
with open('token_frequencies.txt', 'r', encoding='utf-8') as f:
    for line in f:
        parts = line.strip().split('\t')
        if len(parts) == 2:
            term, freq = parts
            legal_freq[term] = int(freq)

# 3. Sort terms by specificity score (highest first)
potential_jargon = sorted(jargon_scores.items(), key=lambda x: x[1], reverse=True)

# 4. Filter terms that appear frequently enough
min_frequency = 5  # Adjust if needed
filtered_jargon = [(term, score) for term, score in potential_jargon
                   if legal_freq.get(term, 0) >= min_frequency]

# 5. Take top 500 terms
top_jargon = filtered_jargon[:500]

# 6. Save top jargon terms to a file
with open('top_jargon.txt', 'w', encoding='utf-8') as f:
```

```
    for term, score in top_jargon:
        f.write(f"{term}\t{score:.6f}\n")

print("Top jargon terms saved to 'top_jargon.txt'")
```

7.

```
import re
import unicodedata

# Normalize and clean Malayalam token
def clean_token(token):
    token = unicodedata.normalize('NFC', token)
    token = token.replace('\u200c', '').replace('\u200d', '')
    return token

# Preprocess text: normalize and tokenize safely
def preprocess_malayalam(text):
    text = unicodedata.normalize('NFC', text)
    text = text.replace('\u200c', '').replace('\u200d', '')
    text = re.sub(r'[“”‘’——!"#$%&\'()*+,\-./:;<=>?@[\\]^_`{|}~]', '', text)
```

```
    tokens = text.strip().split()
    return [clean_token(t) for t in tokens]


# Find up to 5 context windows for a term
def find_contexts(term, corpus_file, window=5):
    contexts = []
    norm_term = clean_token(term)

    with open(corpus_file, 'r', encoding='utf-8') as f:
        text = f.read()
        tokens = preprocess_malayalam(text)

    positions = [i for i, t in enumerate(tokens) if t == norm_term]
    for i, pos in enumerate(positions[:5]):
        start = max(0, pos - window)
        end = min(len(tokens), pos + window + 1)
        context = ' '.join(tokens[start:end])
        contexts.append(f"Context {i+1}: {context}")
    return contexts


# Read term-score map
```

```
def read_score_file(filename):
    score_dict = {}
    with open(filename, 'r', encoding='utf-8') as f:
        for line in f:
            parts = line.strip().split()
            if len(parts) == 2:
                term, score = parts
                score_dict[term] = score
    return score_dict


# Read term-frequency map
def read_frequency_file(filename):
    freq_dict = {}
    with open(filename, 'r', encoding='utf-8') as f:
        for line in f:
            parts = line.strip().split()
            if len(parts) == 2:
                term, freq = parts
                freq_dict[term] = freq
    return freq_dict
```

```
# Main script
jargon_file = 'top_jargon2.txt'
score_file = 'term_specificity.txt'
freq_file = 'token_frequencies.txt'
corpus_file = 'Final1.1.txt'
output_file = 'jargon_contexts_output_final1.txt'

# Load dictionaries
score_dict = read_score_file(score_file)
freq_dict = read_frequency_file(freq_file)

# Process each term
with open(jargon_file, 'r', encoding='utf-8') as jfile, open(output_file, 'w', encoding='utf-8') as out:
    for line in jfile:
        term = line.strip()
        if not term:
            continue
        score = score_dict.get(term, 'N/A')
        freq = freq_dict.get(term, '0')

        out.write(f"Term: {term}\n")
```

```
        out.write(f"Score: {score}\n")
        out.write(f"Frequency: {freq}\n")
        out.write("Context:\n")

        contexts = find_contexts(term, corpus_file)
        if contexts:
            for ctx in contexts:
                out.write(f"  {ctx}\n")
        else:
            out.write("  No contexts found.\n")

        out.write("\n")
```

8.

```
pythonimport json
import csv

# Save as JSON
with open('malayalam_legal_jargon.json', 'w', encoding='utf-8') as f:
    json.dump(jargon_with_context, f, ensure_ascii=False, indent=2)
```

```
# Save as CSV
with open('malayalam_legal_jargon.csv', 'w', encoding='utf-8', newline='') as f:
    writer = csv.writer(f)
    writer.writerow(['Term', 'Specificity Score', 'Frequency', 'Example Context'])
    for term, data in jargon_with_context.items():
        writer.writerow([term, data['score'], data['frequency'],
                         data['contexts'][0] if data['contexts'] else ''])
```

Tokenized Dataset:-

1986

മേയ്

1ാം

തീയതി

നിലവിലുള്ള

പ്രകാരമുള്ള

1926ലെ

പ്രോമിസറിനോട്ട്

സ്റാമ്പ്

ആക്ടിന്റെ

1926ലെ

11ാം

ആക്ട്

മലയാളത്തിലെ

ആധികാരിക

പരിഭാഷ

ഈ

പതിപ്പിൽ

അടങ്ങിയിരിക്കുന്നു

Tokenized and Frequency:-

ഒരു 804

അല്ലെങ്കിൽ 709

ഈ 637

ഏതെങ്കിലും 619

പ്രകാരം 433

1 397

2 393

വകുപ്പ് 370

സർക്കാർ 364

കേന്ദ്ര 356

സംസ്ഥാന 345

ആ 343

ക 312

ഖ 291

മറ്റ് 275

അതിന്റെ 270

അങ്ങനെയുള്ള 262

രജിസ്ട്രർ 261

ആക്ട് 259

Bigram:-

കേന്ദ്ര സർക്കാർ 141

സംസ്ഥാന സർക്കാർ 111

രജിസ്റ്റർ ചെയ്ത 106

ഉപവകുപ്പ് പ്രകാരം 98

ഈ ആക്ട് 96

ഈ ആക്ടിന്റെ 87

ബഹുവിധ പരിവഹണ 85

വകുപ്പ് പ്രകാരം 70

വിജ്ഞാപനം വഴി 66

1 ഈ 64

ഔദ്യോഗിക ഗസറ്റിൽ 63

തീയതി മുതൽ 59

2 1ാം 55

1ാം ഉപവകുപ്പ് 54

വകുപ്പ് 1ാം 53

അല്ലെങ്കിൽ ഖ 53

ഒരു ഭൂവിവരണ 53

ഭൂവിവരണ സൂചനയുടെ 53

ഈ ആക്ട് 51

Trigram:-

ഔദ്യോഗിക ഗസറ്റിൽ വിജ്ഞാപനം 47

ഗസറ്റിൽ വിജ്ഞാപനം വഴി 37

വകുപ്പ് 1ാം ഉപവകുപ്പ് 37

1ാം ഉപവകുപ്പ് പ്രകാരം 36

ഈ ആക്ട് പ്രകാരം 36

വകുപ്പ് 2ാം ഉപവകുപ്പ് 30

സമുചിത സർക്കാരുകളും തദ്ദേശ 24

ഈ ആക്ട് പ്രകാരമുള്ള 22

ബഹുവിധ പരിവഹണ ഓപ്പറേറ്റർ 22

കേന്ദ്ര സർക്കാർ നിർണ്ണയിക്കുന്ന 20

രജിസ്റ്റർ ചെയ്ത ഒരു 20

പ്രകാരം നാമനിർദ്ദേശം ചെയ്ത 19

കേന്ദ്രസർക്കാർ ഔദ്യോഗിക ഗസറ്റിൽ 18

ഒരു ഭൂവിവരണ സൂചന 18

ഈ ആക്ടിന്റെ കീഴിൽ 18

1 ഈ ആക്ട് 17

വകുപ്പ് 3ാം ഉപവകുപ്പ് 16

Term Specificity:-

ഒരു 535.181036

അല്ലെങ്കിൽ 471.944471

ഏതെങ്കിലും 412.036146

പ്രകാരം 288.225608

വകുപ്പ് 246.289780

സർക്കാർ 242.295892

കേന്ദ്ര 236.970707

സംസ്ഥാന 229.648579

മറ്റ് 183.053215

അതിന്റെ 179.724975

അങ്ങനെയുള്ള 174.399790

രജിസ്റ്റർ 173.734142

ആക്ട് 172.402846

എന്നാൽ 161.752477

1ാം 138.454795

ഭൂവിവരണ 133.795259

ചെയ്ത 129.135723

ഉപവകുപ്പ് 107.834985

Top Jargons after filtering:-

ഉപവകുപ്പ് 107.834985

അപേക്ഷ 93.190728

സർക്കാരിന്റെ 91.859432

വൈകല്യങ്ങളുള്ള 89.196839

ബോർഡ് 86.534247

ഉത്തരവ് 71.224342

രജിസ്ട്രേഷൻ 69.227398

വിജ്ഞാപനം 65.233509

സർക്കാരിന് 64.567861

അപ്പീൽ 46.595364

ഡിസൈൻ 46.595364

തടങ്കൽ 45.929716

അധികാരം 43.267124

സാധനങ്ങളുടെ 42.601475

വ്യവസ്ഥകൾക്ക് 42.601475

സാമ്പത്തിക 37.941939

വിവരങ്ങൾ 37.276291

കാലാവധി 37.276291

ഏകോപന 37.276291

Glossary with Contexts:-

Term: വകുപ്പ്

Score: 246.289780

Frequency: 370

Context:

1. 1988 ആഗസ്റ്റ് 2-ാം തീയതിയിലെ അസാധാരണ കേന്ദ്ര ഗസറ്റ് XI-ാം ഭാഗം 1-ാം [വകുപ്പ്], 1-ാം വാല്യം 1-ാം നമ്പരായി 26 മുതൽ 31 വരെയുള്ള പുറങ്ങളിൽ ഇത് പ്രസിദ്ധീകരിച്ചിരുന്നു.

2. ഈ [വകുപ്പിന്റെ] ആവശ്യങ്ങൾക്കു, സംസ്ഥാനസർക്കാർ പ്രത്യേകമായി അധികാര നൽകിയിട്ടുള്ള ആ സർക്കാരിന്റെ...

3. .(3) (2)-ാം [ഉപവകുപ്പിൽ] പറഞ്ഞിട്ടുള്ള ഒരു ഉദ്യോഗസ്ഥൻ ഈ വകുപ്പു (പകാരം ഏതെങ്കിലും ഉത്തരവ് പുറപ്പെടുവിക്കുമ്പോൾ, ഉത്തരവ് പുറപ്പെടുവിക്കാനുള്ള കാരണങ്ങളും അയാളുടെ അഭിപ്രായത്തിൽ സംഗതിയോടു ബന്ധപ്പെട്ട മറു വിവരങ്ങളും സഹിതം ആ വിവരം ഉടനെ തന്നെ അയാൾ വിധേയനായിരിക്കുന്ന സംസ്ഥാന സർക്കാരിന് റിപ്പോർട്ട് ചെയ്യേണ്ടതും, ...

4. (4) ഈ [വകുപ്പുപ്രകാരം] സംസ്ഥാന സർക്കാർ ഏതെങ്കിലും ഉത്തരവ് പുറപ്പെടുവിക്കുകയോ അംഗീകരിക്കുകയോ ചെയ്യുമ്പോഴും (1)-ാം ഉപവകുപ്പ് പ്രകാരം പ്രത്യേകമായി അധികാരപ്പെടുത്തിയിട്ടുള്ള, സംസ്ഥാന സർക്കാരിന്റെ സെക്രട്ടറിയുടെ പദവിയിൽ താഴെയല്ലാത്ത, ആ സർക്കാരിന്റെ ഒരു ഉദ്യോഗസ്ഥൻ ഈ വകുപ്പുപ്രകാരം ഏതെങ്കിലും ഉത്തരവ് പുറപ്പെടുവിക്കുമ്പോഴും...

5. 12-ാം [വകുപ്പിന്റെ] അനുസരണം, സ്ഥിരീകരിച്ച ഏതെങ്കിലും തടങ്കൽ ഉത്തരവ്, തടങ്കലിന്റെ പരിമിതിയുടെ അനുസരണം, തടങ്കലിൽ വയ്ക്കുന്ന ഒരാളുടെ പരമാവധി കാലയളവ്, തടങ്കൽ തീയതി മുതൽ ആറുമാസം ആയിരിക്കും.എന്നാൽ, ഈ വകുപ്പിൽ അർത്ഥവത്തായ പ്രവർത്തനം ഏതും,

തടങ്കൽ ഉത്തരവ് ഏതെങ്കിലും സമയത്ത് പിൻവലിക്കുകയോ ഭേദഗതി ചെയ്യുകയോ ചെയ്യുന്ന സമുചിത സർക്കാരിന്റെ അധികാരത്തെ ബാധിക്കാത്തതാണ്.