



**Australian Government**



**GovHack**  
empower. enable. connect.  
**Hackerspace**

---

# Transforming Bureaucratic Jargon into Plain English: A Holistic Approach to AI Integration in Government Documents Workflows

---

Project Team: WEKA

GUAN, YONGZHEN; JOSIAH WATSON  
South Australia  
SEPTEMBER 7, 2024

## Table of Contents

Introduction.....	1
1. Developing or Applying AI Tools for Clear and Accurate Government Content .....	2
1.1 Platform Architecture and Technology Selection .....	2
1.2 Data Collection and Annotation Process .....	2
1.3 Model Fine-Tuning and Optimisation Strategy .....	3
1.4 Real-time Monitoring and Continuous Improvement.....	3
2. Ensuring AI Tools Follow Australian English Conventions .....	3
2.1 Expanding Localised Language Datasets .....	3
2.2 Automated Spelling and Grammar Correction .....	4
2.3 Context-Aware Language Switching .....	4
3. Leveraging Retrieval-Augmented Generation (RAG) for Accurate Content .....	4
3.1 Implementing the RAG Architecture .....	5
3.2 Source Quality Control and Optimisation.....	5
4. Enhancing Content Accessibility .....	5
4.1 Accessible Content Generation Framework .....	5
4.2 Automated Subtitles and Transcripts .....	6
5. Modular AI Architecture for Task Specialisation .....	6
5.1 Layered, Modular AI Design .....	6
5.2 Collaborative Workflow Management .....	6
6. Integrating AI Tools into Existing Government Workflows .....	6
6.1 API Design and Integration .....	7
6.2 User-friendly Interface and Feedback System .....	7
6.3 Continuous Improvement through Feedback .....	7
Conclusion .....	7
Appendix.....	8
Table1 Technologies of Integrating AI in Government Document Workflows .....	8

## Introduction

The integration of Artificial Intelligence (AI) into government document workflows presents a valuable opportunity to streamline the creation and editing of clear, accurate, and user-friendly

content. In line with the challenge of using AI to transform bureaucratic jargon into plain English, this report provides a comprehensive framework to address this need. By ensuring that government documents follow the Australian Government Style Manual and meet the specific requirements of Australian citizens, the approach focuses on a structured, technology-driven solution. This report outlines the step-by-step process for selecting the appropriate technology stack, applying AI tools, and seamlessly integrating them into document workflows, ensuring adherence to Australian English standards, content accuracy, and improved accessibility.

## 1 Developing or Applying AI Tools for Clear and Accurate Government Content

### 1.1 Platform Architecture and Technology Selection

The first step is selecting the right AI platform based on the **data sensitivity** and **deployment needs**.

- **Cloud-based deployment:** For general tasks requiring real-time content generation and editing, **Azure OpenAI Service** is an ideal platform. It offers robust **Natural Language Processing (NLP)** capabilities and API integration. To secure sensitive data, a **Virtual Network** can be used to create an isolated, secure environment, ensuring all data processing happens within a closed network.
- **On-premise deployment:** If network isolation or data security is a concern, **local deployment of open-source models** like **LLaMA** can be implemented. This setup requires high-performance local servers equipped with **NVIDIA A100 GPUs** to support the training and inference workloads. Additionally, **load balancing** and **distributed processing** are necessary to manage multiple concurrent tasks efficiently.

### 1.2 Data Collection and Annotation Process

To train AI tools effectively, the next crucial step is to collect and annotate a high-quality dataset that mirrors the language and style required in Australian government documents.

- **Data Collection:** Automated data collection systems should be implemented using tools like **Scrapy** or **BeautifulSoup** to regularly extract text from **government websites, policy databases, and parliamentary records**. This ensures the data is both up-to-date and comprehensive.
- **Data Cleaning and Annotation:** Use **Natural Language Processing (NLP)** tools like **NLTK** and **SpaCy** to clean and format the data, removing redundant or irrelevant information. Establish an annotation team or platform (e.g., **Amazon Mechanical Turk**)

to manually label datasets according to the Style Manual. Key attributes such as **tone**, **grammar**, **punctuation**, and **clarity** should be tagged for machine learning purposes.

### 1.3 Model Fine-Tuning and Optimisation Strategy

AI models must be fine-tuned for specific government-related tasks using the collected data.

- **Multi-task Learning Architecture:** To handle various tasks such as **spell checking**, **sentence simplification**, and **tone adjustment**, a **multi-task learning framework** should be implemented. By leveraging models like **T5** or **BART** through **Azure ML Studio**, AI can be trained to handle multiple document tasks simultaneously, improving overall efficiency and consistency.
- **Deep Learning Optimisation:** Using **Transformer optimisation techniques** such as **sparse attention** reduces computational overhead, speeding up content generation without sacrificing accuracy. In environments with limited computational resources, **knowledge distillation** can compress model size to suit lightweight devices like mobile or low-power servers.

### 1.4 Real-time Monitoring and Continuous Improvement

To maintain AI performance and ensure adherence to the Australian Government Style Manual, continuous monitoring and optimisation are essential.

- **Performance Monitoring:** Implement systems like **Prometheus** and **Grafana** to track the AI's **latency**, **response times**, and **success rates**. Any deviations from the set performance benchmarks should trigger immediate adjustments in model parameters or data processing strategies.
- **Continuous Integration/Continuous Deployment (CI/CD):** Set up a **CI/CD pipeline** to automatically test and deploy model updates. Should the model exhibit performance degradation or output errors, a rollback mechanism should restore a previous, stable version.

## 2 Ensuring AI Tools Follow Australian English Conventions

### 2.1 Expanding Localised Language Datasets

To ensure the AI uses **Australian English** consistently, it is essential to gather and maintain datasets that reflect **Australian spelling, grammar, and style**.

- **Dynamic Corpus Expansion:** Regularly source language data from **government archives, Australian media, and the National Library** to keep the dataset current. The AI should learn new terms and usage trends from various sectors, including agriculture, healthcare, and finance, to remain accurate and relevant.
- **Sector-Specific Datasets:** For government departments such as **mining, agriculture, or finance**, collecting domain-specific documents will enhance the AI's ability to accurately generate and correct technical content in these areas.

## 2.2 Automated Spelling and Grammar Correction

Ensuring consistent spelling and grammar in **Australian English** can be achieved by developing dynamic AI models that adjust based on context.

- **Dynamic Spell Checking:** Implement a **dynamic spell-checking module** using algorithms like **Hunspell**, which adapts to context and corrects spelling based on sentence structure. For example, in government announcements, the AI should prioritise “realise” over “realize.”
- **Post-Processing for Language Correction:** After generating text, an additional **post-processing step** should apply **rule-based systems** to automatically verify spelling, punctuation, and adherence to the Australian Government Style Manual.

## 2.3 Context-Aware Language Switching

In certain documents, particularly those with international content, the AI needs to adjust language use based on context.

- **Contextual Embedding:** Using **contextual embedding techniques**, the AI can distinguish when to use Australian English (e.g., "labour" vs. "labor") and when to retain globally recognised terms in specific agreements or reports.
- **Terminology Exceptions:** Set up a **whitelist mechanism** that allows specific technical terms (e.g., "meter" for international measurements) to bypass automatic Australian spelling rules.

## 3 Leveraging Retrieval-Augmented Generation (RAG) for Accurate Content

### 3.1 Implementing the RAG Architecture

To ensure the content generated by the AI is accurate and sourced from **reliable government databases**, **Retrieval-Augmented Generation (RAG)** models should be used.

- **Bidirectional Retrieval Process:** A RAG model retrieves background information from **government databases** before generating content. By using tools like **BERT embeddings**, the retrieved information is integrated into the AI's generation pipeline, ensuring that the content reflects authoritative sources.
- **Generate-Retrieve-Verify Loop:** Incorporate a loop where AI-generated content is verified against retrieved data, and any discrepancies trigger a regeneration of the text to match the source information.

### 3.2 Source Quality Control and Optimisation

The accuracy of the AI depends on the reliability of its sources.

- **Source Credibility Scoring:** Develop a **source quality control system** that scores data sources based on **credibility**, **freshness**, and **completeness**. Prefer authoritative databases like the **Australian Bureau of Statistics** or **legislative records**.
- **Efficient Caching and Pre-retrieval:** To speed up the retrieval process, implement a **Redis caching system** to store frequently accessed policy documents and legal records.

## 4 Enhancing Content Accessibility

### 4.1 Accessible Content Generation Framework

AI-generated government content must be accessible to **all citizens**, including those with disabilities and diverse cultural backgrounds.

- **Automated Accessibility Checkers:** Integrate accessibility checkers like **WAVE** or **Axe** to ensure generated content complies with **WCAG 2.1** standards. The AI should ensure that all images have appropriate alt text, headings are properly structured, and text is screen-reader compatible.
- **Cross-device Compatibility:** Ensure AI-generated content adapts to different formats like **HTML**, **PDF**, and **Word**, allowing seamless access on various devices (e.g., computers, mobile phones, or e-readers).

## 4.2 Automated Subtitles and Transcripts

Providing subtitles and transcripts for multimedia content increases accessibility.

- **Speech-to-Text Models:** Use **Azure Speech-to-Text** to automatically generate accurate transcripts of government audio or video content. Open-source models like **Vosk** or **DeepSpeech** can complement these tools to ensure accuracy and language coverage.
- **Multi-language Subtitle Generation:** Train AI to support multi-language subtitle generation using **Seq2Seq translation models**, ensuring that non-English speakers can access content.

## 5 Modular AI Architecture for Task Specialisation

### 5.1 Layered, Modular AI Design

Creating specialised AI modules allows for efficient task execution.

- **Task Layering:** Design a layered architecture where **base tasks** like spelling correction are handled in the first layer, while **advanced tasks** like tone adjustment and accessibility checks occur in later layers. This approach ensures that simpler tasks are completed before more complex modifications are applied.
- **Dependency Management:** Implement a **dependency management system** to ensure modules interact effectively. For example, tone adjustment should occur after grammar correction, preventing conflicts in task execution.

### 5.2 Collaborative Workflow Management

To improve AI collaboration, build a workflow system using tools like **Apache Airflow**.

- **Task Scheduling and Load Balancing:** Implement task scheduling and **load-balancing algorithms** to distribute workloads among AI modules. This ensures the system remains responsive, even during peak usage times.

## 6 Integrating AI Tools into Existing Government Workflows

## 6.1 API Design and Integration

Seamless integration of AI into existing government workflows is essential for improving document quality and efficiency.

- **REST API Development:** Develop a **multi-function REST API** that allows government staff to access AI tools for **text generation, editing, and style correction**. API endpoints should offer flexible parameters to customise document generation, such as tone and style preferences.
- **API Security Enhancements:** Ensure the API adheres to **security standards** by implementing **OAuth 2.0** for authentication and **TLS encryption** to secure data transmission.

## 6.2 User-friendly Interface and Feedback System

A user-friendly interface will enable staff to interact with the AI effectively.

- **Document Editor Plugins:** Develop plugins for commonly used editors like **Microsoft Word** and **Google Docs**. These plugins should offer real-time feedback on AI-generated suggestions and allow users to accept, reject, or adjust changes.
- **Visual Feedback and Learning Tools:** Provide a visual feedback system that highlights AI-driven changes, explaining why certain modifications were suggested. This feature helps users learn from the AI and improve their future document creation.

## 6.3 Continuous Improvement through Feedback

AI models must evolve based on user feedback.

- **Feedback Loops:** Establish **feedback collection mechanisms**, where government staff can rate and comment on AI suggestions. Use this feedback to fine-tune AI models through **reinforcement learning**.
- **Version Control and Grey Release Strategy:** To minimise disruptions, implement a **grey release strategy** where new versions of the AI model are gradually introduced. This allows for the collection of performance data before full deployment, with the option to revert to previous versions if issues arise.

## Conclusion

By adopting this structured, holistic approach, governments can effectively use AI to transform bureaucratic jargon into plain English. This approach ensures that documents adhere to Australian English standards while maintaining accuracy and clarity, making them more



accessible to citizens. AI tools enable seamless simplification of complex language, ensuring that government communication is more transparent and user-friendly. By integrating AI into existing systems with intuitive interfaces and continuous feedback, this solution remains flexible, secure, and capable of modernizing government communication for the long term.

## Appendix

*Table1 Technologies of Integrating AI in Government Document Workflows*

Step	Details
Platform Architecture and Technology Selection	<ul style="list-style-type: none"> <li>- Cloud-based deployment using Azure OpenAI Service for general tasks, VNet for secure processing.</li> <li>- On-premise deployment with open-source models like LLaMA, using NVIDIA A100 GPUs for local servers.</li> </ul>
Data Collection and Annotation Process	<ul style="list-style-type: none"> <li>- Automated data collection from government sites using Scrapy or BeautifulSoup.</li> <li>- Data cleaning with NLP tools like NLTK and SpaCy.</li> <li>- Manual annotation using platforms like Amazon Mechanical Turk.</li> </ul>
Model Fine-Tuning and Optimisation	<ul style="list-style-type: none"> <li>- Multi-task learning architecture using models like T5 or BART.</li> <li>- Deep learning optimisation techniques, including sparse attention and knowledge distillation.</li> </ul>
Real-time Monitoring and Continuous Improvement	<ul style="list-style-type: none"> <li>- Performance tracking using Prometheus and Grafana.</li> <li>- CI/CD pipeline setup for automatic testing and deployment.</li> </ul>
Ensuring AI Tools Follow Australian English	<ul style="list-style-type: none"> <li>- Expanding localised datasets from government archives and media.</li> <li>- Context-aware spell-checking with algorithms like Hunspell.</li> </ul>
Retrieval-Augmented Generation (RAG)	<ul style="list-style-type: none"> <li>- Use of RAG architecture with a bidirectional retrieval process.</li> <li>- Generate-Retrieve-Verify loop to ensure content accuracy.</li> </ul>

Enhancing Content Accessibility	<ul style="list-style-type: none"> <li>- Use automated checkers like WAVE or Axe for accessibility.</li> <li>- Speech-to-text models for transcript and subtitle generation.</li> </ul>
Modular AI Architecture	<ul style="list-style-type: none"> <li>- Task layering for specialized AI modules (spelling correction, tone adjustment).</li> <li>- Dependency management system to ensure smooth task execution.</li> </ul>
Integrating AI into Government Workflows	<ul style="list-style-type: none"> <li>- Development of REST APIs for seamless AI integration.</li> <li>- User-friendly plugins for document editors and real-time feedback.</li> </ul>