

AI in Society and Public Services

Session 5: Artificial Intelligence in Public Services

Mário Antunes

January 30, 2026

Universidade de Aveiro

Table of Contents i

AI in Society and Public Services

Part I: Leveraging Large Language Models

Part II: Using the LLM models

Part III: Use Cases

Part IV: RAGs

Q&A

AI in Society and Public Services

Session 5: Artificial Intelligence in Public Services

Duration: 3 Hours

Instructor: [Mário Antunes](#)

Session details ii

Scan the QR code below to access all slides, code examples, and resources for this workshop.



Figure 1: Repository QR Code

Link: <https://github.com/mario-antunes/aiml-society>

Part I: Leveraging Large Language Models

Refresher: The AI Hierarchy

Setting the stage: Where do LLMs fit?

- **Artificial Intelligence (AI):** The broad discipline of creating intelligent machines.
- **Machine Learning (ML):** Systems that learn from data rather than being explicitly programmed.
- **Deep Learning (DL):** ML using multi-layered neural networks (mimicking the human brain).
- **Generative AI:** DL models capable of generating *new* content (text, images, audio).

The Engine: The Transformer Architecture

- Introduced by Google in 2017 (“Attention Is All You Need”).
- **Key Innovation:** The “Self-Attention” mechanism.
- **Function:** It allows the model to weigh the importance of different words in a sentence regardless of their distance from each other.
- *Example:* In “The animal didn’t cross the street because **it** was too tired,” the model understands **it** refers to *animal*, not *street*.

How LLMs “Think” (Next Token Prediction)

- LLMs are fundamentally **probabilistic engines**.
- They do not “know” facts; they predict the most likely next piece of text (token).
- **The Equation:**
Probability of word , given the sequence of previous words.

Tokenization: The Language of Machines

- Models don't read words; they read numbers.
- **Tokenization:** Breaking text into smaller units (tokens).
- 1 Token 0.75 words.
- Common words like "the" are one token. Complex words like "bureaucracy" might be split: bur eau cracy.
***Relevance:** This impacts cost (API usage) and context window limits.*

The Training Pipeline

To get a usable model for public service, we go through three stages:

1. **Pre-training:** Reading the internet (learning grammar, facts, reasoning). Result: *Base Model*.
2. **Supervised Fine-Tuning (SFT):** Learning Q&A formats. Result: *Instruct Model*.
3. **RLHF (Reinforcement Learning from Human Feedback):** Learning alignment, safety, and preference. Result: *Chat Model*.

Part II: Using the LLM models

Why Local Models? (Public Service Context)

Before we start our practical demo, why run this locally?

- **Data Sovereignty:** No data leaves your secure infrastructure.
- **GDPR/Privacy Compliance:** Essential when handling citizen PII (Personally Identifiable Information).
- **Cost Control:** No per-token API fees to external vendors.
- **Reliability:** Works offline or in air-gapped networks.

- **Tool:** Ollama (Open-source tool to run LLMs locally).
- **Model:** `qwen3:0.6b` (A hypothetical highly efficient small language model).

Note: Small models are faster and require less hardware, perfect for edge devices in government offices.

First Interaction (Sanity Check)

Let's test the model's basic reasoning.

Prompt:

Explain the concept of 'Public Value' in one sentence.

Expected Output:

Public Value describes the benefit derived by the public from the services provided by the government, focusing on equity, efficiency, and citizen satisfaction.

Discussion: Notice the speed. Small models generate text very quickly.

Part III: Use Cases

Use Case 1: Summarizing Policy Documents

- **Challenge:** Civil servants often have to read lengthy regulations.
- **Goal:** Extract key points quickly.

Prompt:

Summarize the following text into 3 bullet points for a briefing note. These points should identify who, what and when.

TEXT:

"The new Digital Services Act requires all municipal websites to adhere to WCAG 2.1 accessibility standards. This includes mandatory alt-text for images, keyboard navigability, and screen-reader compatibility. Failure to comply by Q3 2025 will result in audits."

Model Output:

- Municipal websites must comply with WCAG 2.1 accessibility standards.
- Key requirements include alt-text, keyboard navigation, and screen-reader support.
- Compliance deadline is Q3 2025, with audits for non-compliance.

Verdict: *The model accurately identified the Who, What, and When.*

Use Case 2: Citizen Communication (Drafting)

- **Challenge:** Replying to repetitive inquiries while maintaining a polite, formal tone.
- **Goal:** Draft a template response.

Prompt:

Write a polite email to a resident explaining that their permit application (ID: #4059) is currently 'Under Review' and that the expected processing time is 10 business days. Tone: Formal but empathetic.

Use Case 2: Analysis

Model Output:

Subject: Update regarding your Permit Application #4059

Dear Resident,

Thank you for your patience. We are writing to inform you that your permit application (ID: #4059) is currently Under Review. We understand the importance of this request and are working diligently to process it. Please expect a decision within the next 10 business days.

Sincerely,

The Planning Department

Critique: This draft is 90% ready. The human officer only needs to review and send.

Use Case 3: Sentiment Analysis on Feedback

- **Challenge:** Analyzing thousands of open-text survey responses from citizens.
- **Goal:** Categorize sentiment automatically.

Prompt:

Classify the sentiment of the following feedback as Positive, Negative, or Neutral:

"I waited in line for 2 hours and the system crashed when I finally got to the counter. The staff tried their best, but the technology is broken."

Model Output:

Negative.

- **Advanced Tip:** You can ask the model to extract *why*:

What is the root cause of the complaint?

Root cause: Long wait times and system/technology failure.

The Hallucination Problem

- **Definition:** When an LLM generates factually incorrect information with high confidence.
- **Why?** Remember, it predicts the *next likely token*, not the truth.
- **Risk in Government:** Citing non-existent laws or inventing incorrect procedures.

Let's try to trick the model.

Prompt:

Who was the King of France in 2005?

- **Potential Hallucination:** "The King of France in 2005 was [Name]." (Incorrect, France is a Republic).
- **Ideal Response:** "France was a Republic in 2005, so it did not have a King." > *Note:* Smaller models (0.6b) are more prone to hallucinations than larger ones (70b). Always verify.

- Models are trained on the internet. The internet contains bias.
 - **Risk:**
 - Hiring algorithms favoring certain demographics.
 - Predictive policing models targeting specific neighborhoods.
- Mitigation:*** We must use “System Prompts” to enforce neutrality and fairness.

You can set the “behavior” of the model before the conversation starts.

System Prompt Example:

You are a helpful assistant for the City Council. You must answer questions accurately based only on provided facts. If you do not know the answer, state that you do not know. Do not invent information.

Advanced Technique: Few-Shot Prompting

Instead of asking the model to guess what you want, provide examples.

Prompt:

Convert these addresses to standard format.

Input: 123 main st, apt 4, springfield

Output: 123 Main St, Apt 4, Springfield

Input: 45 west elm, chicago, il

Output: 45 W Elm St, Chicago, IL

Input: 999 broadway, floor 2, ny

Output:

- **Result:** 999 Broadway, Floor 2, NY (The model learns the pattern instantly).

Advanced Technique: Chain of Thought (CoT)

Encourage the model to “show its work.” This reduces errors in logic.

Prompt:

A department has a budget of \$10,000.
They spent 20% on software and \$3,000 on hardware.
How much is left?
Let's think step by step.

Output:

1. Total Budget: \$10,000
2. Software: 20% of 10,000 = \$2,000
3. Hardware: \$3,000
4. Total Spent: \$2,000 + \$3,000 = \$5,000
5. Remaining: \$10,000 - \$5,000 = \$5,000

Part IV: RAGs

Retrieval Augmented Generation (RAG)

- **The Problem:** LLMs are frozen in time (training data cut-off) and don't know your private documents.
 - **The Solution:** RAG.
1. **Retrieve:** Search your internal database for relevant documents.
 2. **Augment:** Paste those documents into the prompt.
 3. **Generate:** Ask the LLM to answer using *only* that text.

RAG in Practice (Conceptual)

Standard Prompt: "What is the vacation policy?" *Model:*
(Gives generic advice).

RAG Prompt:

Context: "According to the 2024 Employee Handbook
Section 4.2, employees accrue 1.5 days of leave
per month."

Question: "What is the vacation policy?"
Answer using the context above.

Model: "Employees accrue 1.5 days of leave per month per
the 2024 Handbook."

- **Public Cloud (OpenAI/Microsoft):** Data leaves your premises.
- *Pros:* Best performance.
- *Cons:* Privacy risks, data sovereignty issues.

- **Private Cloud / Local (Ollama):** Data stays with you.
- *Pros:* Secure, compliant.
- *Cons:* Requires hardware maintenance, models are generally smaller/dumber.

When deploying AI in public service, adhere to the **FAST** principles:

1. **Fairness:** No bias in output.
2. **Accountability:** A human must always be responsible for the final decision (Human-in-the-loop).
3. **Safety:** The system must not harm citizens or infrastructure.
4. **Transparency:** Citizens have a right to know if they are interacting with an AI.

Implementation Strategy

1. **Identify Low-Risk Use Cases:** Start with drafting internal memos, not automated decision-making on benefits.
2. **Pilot Locally:** Use tools like Ollama to test without cost or privacy risk.
3. **Human Verification:** Implement a mandatory review step for all AI-generated content.
4. **Scale:** Move to larger, hosted models only when governance is in place.

Summary of Techniques

Technique	Best For	Complexity
Zero-Shot	Simple queries, creative writing	Low
Few-Shot	Formatting data, specific styles	Medium
Chain of Thought	Math, logic, complex reasoning	Medium
RAG	Q&A on internal/private documents	High

- **Multimodal Models:** AI that can “see” images (e.g., assessing potholes from photos).
- **Agents:** AI that can perform actions (e.g., “Schedule a meeting” or “Update the database”) not just write text.
- **Smaller, Specialized Models:** Moving away from “One model to rule them all” to specific models for Legal, Health, and Finance.

Conclusion

- LLMs are tools, not magic.
- In Public Services, **Trust** is the currency. We cannot afford to lose it through careless AI adoption.
- Start small, run locally (Ollama), verify everything, and focus on augmenting human civil servants, not replacing them.

Q&A
