Dedalo: On-Premise Hybrid Neuro-Symbolic Recognition of Administrative Procedures in Italian Small Municipalities

1st Antonio Capodieci
Dipartimento di Ingegneria dell'Innovazione
Università del Salento
73100 Lecce, Italy
antonio.capodieci@unisalento.it

2nd Mirko Lefonso
Ricerca e Sviluppo
Datapolis S.r.l.
Maglie, Lecce, Italy
mirko.lefonso@datapolis.it

3rd Luca Lucioli
Ricerca e Sviluppo
Datapolis S.r.l.
Maglie, Lecce, Italy
luca.lucioli@datapolis.it

4th Luca Mainetti
Dipartimento di Ingegneria dell'Innovazione
Università del Salento
73100 Lecce, Italy
luca.mainetti@unisalento.it

5th Fernando Stifani
Ricerca e Sviluppo
Datapolis S.r.l.
Maglie, Lecce, Italy
fernando.stifani@datapolis.it

6th Giorgio Fontana
Dipartimento di Ingegneria dell'Innovazione
Università del Salento
73100 Lecce, Italy
giorgio.fontana@unisalento.it

***Abstract*—Italian public administration registers every incoming document under Presidential Decree (DPR) 445/2000 and attaches it to an administrative procedure that determines its deadlines, competent office, and archival classification. We propose a Symbolic[Neuro] multi-agent system for this task in small municipalities, deployable on-premise within a 110 W envelope on commodity hardware. Two learner-agents run in shadow on a pilot tenant: a procedure-recognition agent (hybrid semantic + lexical scoring against a normative catalog) and an office-routing agent (multi-signal conditional cascade with large language model (LLM) fallback below a confidence threshold); a procedure-lifecycle monitoring agent is in alpha. Each agent maintains a per-tenant knowledge base curated by human-in-the-loop feedback; recurrent corrections crystallise into explicit rules only after a formal act of micro-organisation by the records manager (Legislative Decree, D.Lgs, 165/2001). Per-agent production cutover is gated by quantitative agreement-rate thresholds, not elapsed time. The combination addresses accountability, sovereignty, and frugality in Italian local-government AI.**

***Index Terms*—e-Government, administrative procedure recognition, neuro-symbolic AI, large language models, on-premise deployment, human-in-the-loop**

I. Introduction

Among Italy's roughly 7,900 municipalities, over 5,500 have fewer than 5,000 inhabitants; they bear the same legal obligations as large cities with one or two clerks and no IT staff. The full triage of each incoming certified email (PEC, *Posta Elettronica Certificata*) — reading, subject normalisation, sender resolution, procedure identification, office assignment, deadline computation — is performed manually.

Naive translations of cloud AI tools fail on multiple constraints simultaneously: commercial APIs introduce costs incompatible with municipal budgets and conflict with sovereignty requirements (AgID, Italy's national digital agency; EU AI Act [1]); black-box classifiers offer no audit trail for the records manager who signs the decisions; generic document-type taxonomies fail to align with the normative procedure catalogs already in use for archival classification. We target on-premise deployment as the prudent choice for the current regulatory moment. The framework governing AI in public administration is still consolidating as the EU AI Act phases in and AgID's guidance takes shape, and keeping administrative data within the issuing authority's own perimeter is the conservative option while these rules settle. Moreover, a centralised, powerful cloud-based solution would answer a constraint this task does not have: recognition over a bounded per-tenant catalog runs on a 4B-parameter quantised model within a ˜110 W envelope (Table II), so the binding constraints — data egress, processor obligations, per-call API spend — are legal and economic rather than computational, and replicated deployment on municipality-owned hardware is practical.

The decisive factor is structural. Unlike a private organisation, which can segment what it exposes to a cloud service, a protocol office processes an unselected input stream: under DPR 445/2000 every incoming PEC must be registered, and that stream routinely carries special-category and judicial data (GDPR arts. 9–10: welfare, health, judicial notifications). Centralising the triage therefore means wholesale egress of precisely the most protected categories, under the personal accountability of the records manager who signs the registrations; on-premise processing keeps the AI support entirely within that perimeter of accountability. A centralised cloud service, amortising model-serving cost across municipalities and centralising maintenance, remains a natural evolution once this transitional period closes and the applicable obligations are stable.

Recent work has begun integrating LLMs into Italian public-

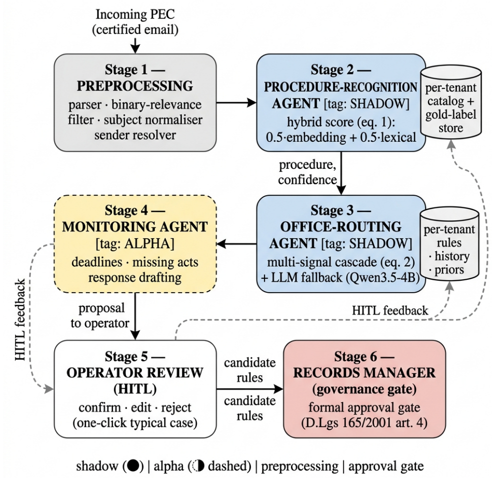

Fig. 1. Architectural view of Dedalo as instantiations of the abstract learner-agent type. All three concrete agents inherit the same four sealed properties; the maturity indicators report per-property implementation status. The data-flow ordering is Recognition → Routing → Monitoring, as detailed in §II-B/C.

administration workflows [2]. Architectural patterns for machine learning (ML)-enabled and sustainable AI within the CINI (National Interuniversity Consortium for Informatics) Smart Cities laboratory [3] and edge/on-premise computing as a smart-city paradigm [4] provide a backdrop. We contribute: (i) Dedalo, a Symbolic[Neuro] multi-agent system deployable on commodity hardware; (ii) a crystallisation governance mechanism linking pattern emergence to formal acts of micro-organisation; (iii) a per-agent shadow-validation methodology with explicit quantitative cutover criteria.

II. System Architecture

A. Multi-agent framing

Dedalo is structured as a multi-agent system (Fig. 1) in the neuro-symbolic family [5], [6]: in Kautz's taxonomy it falls in the Symbolic[Neuro] regime (Type 2), where deterministic symbolic scorers orchestrate the routing decision and a neural sub-component (the locally hosted LLM) is invoked only as a conditional sub-routine.

By *learner-agent* we denote a module satisfying four properties — (i) a per-tenant knowledge base, (ii) a human-in-the-loop (HITL) feedback loop, (iii) shadow-validation infrastructure, (iv) crystallisation governance — without claiming autonomy beyond these properties. Two learner-agents are currently in shadow-mode operation: a procedure-recognition agent and an office-routing agent. A procedure-lifecycle monitoring agent is in alpha; further agents (deadline tracking, missing-document detection, response drafting) are planned. Both realised agents receive input from a deterministic pre-processing pipeline (parser, attachment binary-relevance filter, subject normaliser, sender resolver) scheduled for similar uplift (Section IV).

B. Procedure-recognition agent

The agent scores all active procedure candidates against the input through a hybrid of semantic and lexical similarity:

$$S_{rec}(p) = \frac{1}{2} \cdot \cos(\mathbf{e}_q, \mathbf{e}_p) + \frac{1}{2} \cdot J(\tau_q, \tau_p) \qquad (1)$$

where $\mathbf{e}_q$, $\mathbf{e}_p$ are paraphrase-multilingual-mpnet-base-v2 embeddings of the normalised subject and the procedure label, τ_q, τ_p their morphology-normalised token sets, and J a token-overlap score. The top-1 candidate is retained above a confidence threshold of 0.45, with a rescue band 0.35–0.45 for unusually distinctive lexical matches; below 0.35 the case is routed to manual review. Weights and thresholds were tuned on the pilot tenant and are configurable per tenant. The catalog is a per-tenant projection of a normative baseline: procedures can be added, deactivated, or remapped administratively without retraining.

C. Office-routing agent

The agent consumes (procedure_code, confidence) from §II-B and routes the case to the competent unit. Organisational structure is modelled as a five-tier tree (Area, Sector, Service, Office, Operating Unit), respecting the regulatory autonomy granted under the Consolidated Local Authorities Act (TUEL) art. 5. Routing is a multi-signal conditional cascade: deterministic scorers (rule-based matchers, sender-office history, sender priors) score all nodes simultaneously, with scores propagating upward:

$$S(n) = S_0(n) + \frac{1}{2} \sum_{c \in ch(n)} S(c) + \frac{1}{4} \sum_{g \in gch(n)} S(g) \qquad (2)$$

where $S_0(n)$ is the direct score at node n, and $ch(n)$, $gch(n)$ are its direct children and grandchildren in the org-unit tree. The coefficients (1/2, 1/4) implement a standard decay-by-level schedule, configurable per tenant and scheduled for empirical optimisation once the routing-agent gold set is populated. A locally hosted LLM (Qwen3.5-4B-Q4_K_M via llama.cpp) is invoked only when the top deterministic score falls below a configured threshold. Final fusion is a weighted sum with weights adapted to tenant maturity.

D. Per-agent learning and crystallisation governance

Each agent's per-tenant artifacts — catalog, rules, history, priors, gold-label store (GLS) — are updated by HITL feedback transactionally with protocol registration. Gold labels enter via retrospective annotation by the records manager or via operator confirmations passing a per-band acceptance criterion. Beyond continuous updates, the design specifies that an agent propose new explicit rules from recurring corrections, gated by formal approval from the records manager — mirroring the acts of micro-organisation already required under D.Lgs 165/2001 art. 4. The proposal data structures (DS) and provenance logging are in place; the approval workflow user interface (UI) is under development. Every decision is logged with full provenance (input span, model version, scorer contributions, confidence) for audit. Table I summarises the resulting maturity profile across the three agents.

TABLE I
COMPARISON OF THE THREE LEARNER-AGENTS

Property	Recognition	Routing	Monitoring
Status	shadow (prod-ready)	shadow (val. pending)	alpha
Input	normalised PEC	(proc, conf)	(office, conf)
Output	(proc, conf)	(office, conf)	lifecycle events
Per-tenant KB	catalog · GLS	rules · hist · priors	(planned)
HITL feedback	yes	yes	planned
Crystallisation	DS in place; UI in dev	DS in place; UI in dev	planned
Cutover	$A_b \geq \theta$ (eq. 3)	$A_b \geq \theta$ (eq. 3)	TBD

The four properties are uniform across agents; only realisation status differs.

TABLE II
REFERENCE DEPLOYMENT AND ENGINEERING ENVELOPE

Aspect	Value
Hardware (reference)	AMD Ryzen 9 7945HX + NVIDIA RTX 4060 (8 GB), 32 GB DDR5
Local LLM	Qwen3.5-4B-Q4_K_M via llama.cpp; ˜3.5 GB VRAM
Embedding model	paraphrase-multilingual-mpnet-base-v2 (CPU)
Test workload	50 PECs sustained at 1/min (steady-state)
AMD package power	25 W idle / 25 W mean / 64 W peak (PPT via hwmon)
GPU utilisation	0% idle / 10% mean / 100% peak (nvidia-smi)
System power[a]	˜60 W idle / ˜110 W under load
Rec. latency	n=50; p50 17 s; p95 120 s, max 233 s

[a]Composed estimate: AMD PPT (Package Power Tracking) direct, GPU power as utilisation × power.limit (120 W), chassis overhead ˜25 W; wall-power measurement planned as future work.

III. IMPLEMENTATION STATUS

Both learner-agents are in shadow-mode operation on a pilot tenant, processing every incoming PEC alongside the manual workflow without affecting protocol registration. The shadow-validation infrastructure — gold-label store, agreement-rate report, class-level precision per confidence band — is operational for the recognition agent, exercised over 268 real PECs from the pilot tenant; for the routing agent, gold-labelling is in progress. Agreement is tracked per confidence band $b \in \{HIGH, MED, LOW\}$:

$$A_b = \frac{|\{i : \hat{y}_i = y_i \wedge b_i = b\}|}{|\{i : b_i = b\}|} \tag{3}$$

with $\hat{y}_i$ the agent's proposal, y_i the operator-confirmed gold label, and b_i the band at proposal time; per-agent cutover is gated on A_b exceeding tenant-specific thresholds, with consolidation of a validated gold set for reliable per-band estimates ongoing. The monitoring agent is in early prototype, with functional integration but not yet validated against operator decisions. Per-tenant artifacts are configurable through an administrative interface for the records manager. The deployment relies entirely on open-weight models and open-source runtimes (Table II), removing API costs and runtime dependency on cloud providers.

IV. DISCUSSION AND FUTURE WORK

A. Limitations

Dedalo's current evaluation has three principal limits. First, validation is single-tenant: multi-tenant behaviour is not yet exercised. Second, no comparative evaluation against alternative architectures (single-LLM end-to-end, cloud-RAG, rule-only) is presented. Third, the reported envelope is measured on a development workstation; target-tier benchmarking is planned as future work.

B. Agentic trajectory and collaboration

The two operating learner-agents establish a pattern Dedalo extends symmetrically. Downstream, each new procedure-handling stage (deadlines, missing documents, response drafting) will be realised as an additional learner-agent inheriting the same governance, learning, and shadow-validation regime. Upstream, today's deterministic preprocessing services (subject normaliser, attachment relevance, sender resolution) are scheduled for the same uplift. The pattern admits two endpoints — a coordinated ensemble close to Branch-Train-Merge [7] or a true mixture-of-experts with learned sparse routing [8] — and the shadow-validation methodology generalises to both. In keeping with I-CiTies' tradition of welcoming projects in progress, we invite pilot deployments with municipalities willing to contribute operator feedback.

REFERENCES

[1] Regolamento (UE) 2024/1689 sull'intelligenza artificiale (AI Act); AgID, Linee Guida per l'adozione di IA nella PA, 2024–2025.

[2] D. Bianchini and M. Garda, "Integrating Large Language Models in Judicial Workflows: Designing an LLM-based Crime Reporting Application through Design Science Research," Digital Government: Research and Practice, vol. 7, no. 1, 2026.

[3] H. Muccini, K. Vaidhyanathan, R. Verdecchia et al., "A Synthesis of Green Architectural Tactics for ML-Enabled Systems," in Proc. ICSE-SEIS 2024, IEEE/ACM, 2024.

[4] A. Puliafito et al., "From Sensors to Data Intelligence: Leveraging IoT, Cloud, and Edge Computing with AI," Sensors, vol. 25, no. 6, 2025.

[5] H. Kautz, "The Third AI Summer: AAAI Robert S. Engelmore Memorial Lecture," AI Magazine, vol. 43, no. 1, pp. 105–125, 2022.

[6] A. d'A. Garcez and L. C. Lamb, "Neurosymbolic AI: the 3rd wave," Artificial Intelligence Review, vol. 56, pp. 12387–12406, 2023.

[7] M. Li et al., "Branch-Train-Merge: Embarrassingly Parallel Training of Expert Language Models," arXiv:2208.03306, 2022.

[8] N. Shazeer et al., "Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer," in Proc. ICLR 2017.