Via Nairobi 40
00144, Roma (RM)
info@zendata.it

Sommario

Via Nairobi 40
00144, Roma (RM)
info@zendata.it

Chi siamo

ZenData è una startup dinamica e innovativa, specializzata nello sviluppo di soluzioni basate sull'intelligenza artificiale, per aumentare l'efficienza e stimolare la crescita.

Le nostre soluzioni abbracciano una vasta gamma di settori e reparti, spaziando dallo sviluppo di assistente virtuale per pubbliche amministrazioni e professionisti autonomi, fino all'analisi e all'ottimizzazione dei processi produttivi aziendali.

La nostra strategia è orientata alla progettazione di soluzioni che si integrano in maniera semplice e fluida nei processi operativi dei nostri clienti, garantendo un'adozione senza attriti e un impatto positivo immediato.

Siamo convinti che il nostro approccio all'innovazione, unito alla capacità di personalizzare le soluzioni in base alle specifiche esigenze dei nostri clienti, ci permetta di offrire un valore aggiunto significativo e di contribuire al successo dei nostri partner commerciali.

Via Nairobi 40
00144, Roma (RM)
info@zendata.it

WWW.ZENDATA.IT

1. Executive Summary

1.1 Panoramica

Il progetto propone una **Proof of Concept (PoC) di 8 settimane** per dimostrare la fattibilità di un sistema basato su modelli LLM e grafi di conoscenza che **rilevi e segnali automaticamente i disallineamenti** fra documenti “madre” (interni) e relativi documenti “figlio” (pubblici o derivati).
Partendo da un gruppo pilota di documenti (es. *Policy ESG* + relativi estratti pubblici), la PoC mostrerà come:

1. l’AI estragga entità, sezioni e relazioni chiave;
2. le modifiche in un documento madre vengano propagate a un **knowledge graph**;
3. il sistema individui le parti incoerenti nei documenti collegati;
4. una **dashboard amministrativa** permetta di visualizzare l’alert, navigare il diff semantico e avviare il flusso di aggiornamento/redazione.

In alternativa – e come benchmark di complessità minore – potrà essere configurato un secondo scenario opzionale di **riassunzione automatica dei verbali** di riunioni, riutilizzando lo stesso stack tecnico.

1.2 Obiettivi del Sistema

- **Ingestione automatica e continua**
  Collegamento via IMAP o Microsoft Graph API a una casella condivisa, con polling schedulato per recuperare le email degli ultimi 5–7 giorni e gestione automatica di errori e duplicati.

- **Classificazione “base”**
  - Distinzione tra email che richiedono risposta diretta e email non operative.
  - Smistamento automatico verso i reparti in base al contenuto identificato.
- **Classificazione “avanzata”**
  Riconoscimento di specifiche tipologie di richiesta (anticipazioni fondi, informazioni contributive, variazioni anagrafiche, reclami), per facilitare un ulteriore livello di filtraggio e prioritizzazione.

- **Bozza di risposta automatica**
  Generazione di un testo preliminare, contestualizzato e conforme alle linee guida aziendali, che l'operatore potrà rivedere, completare e inviare con un click.
- **Feedback loop sui prompt**
  Ogni correzione o modifica apportata dall'operatore alle categorie o al testo della bozza viene raccolta per affinare in tempo reale i prompt utilizzati, garantendo un miglioramento progressivo della qualità delle risposte generate.
- **Interfaccia di revisione**
  Dashboard web intuitiva per esplorare le email processate, applicare filtri (categoria, data, stato di revisione), correggere le assegnazioni e inviare le risposte finali, il tutto con un'interfaccia user-friendly.

1.3 Funzionamento del Sistema

1. **Raccolta delle email**
   - Un componente di ingestion (Azure Function o Logic App) effettua il polling della casella condivisa ogni 5–10 minuti, recuperando solo le email degli ultimi 5–7 giorni e garantendo deduplica automatica.
2. **Pre-processing del contenuto**
   - Normalizzazione del testo: rimozione di HTML, firme e boilerplate, estrazione di mittente, oggetto, data/ora e allegati.
   - Preparazione del testo per la chiamata al modello LLM, con operazioni di pulizia e segmentazione.
3. **Classificazione e generazione bozza**
   - **Step 1 – Classificazione "base":** prompt template LLM per stabilire se rispondere o inoltrare e verso quale reparto.
   - **Step 2 – Classificazione "avanzata":** prompt template LLM per identificare la tipologia di richiesta.
   - **Step 3 – Bozza di risposta:** prompt template LLM per generare un draft di risposta coerente con il contenuto e le policy aziendali.
4. **Consegna dei risultati all'applicazione**

- Salvataggio in database dei metadati (categorie proposte, bozza di risposta, timestamp).
- Esposizione via API REST dei risultati per il frontend.

5. **Interfaccia di revisione e feedback**
   - Il frontend Next.js presenta le email classificate con snippet evidenziati, categoria proposta, bozza di risposta e controlli di filtro e ricerca.
   - Ogni correzione dell'operatore — sia sulle categorie sia sul testo della bozza — viene registrata per affinare immediatamente i prompt LLM, migliorando la coerenza e la qualità delle future generazioni.

2. Ambito e Deliverable

Proof of Concept (6 settimane)

Nell'ambito di questa PoC verranno realizzati e consegnati i seguenti elementi:

- **Setup infrastrutturale (Azure)**
  Configurazione di Azure OpenAI Service, Cosmos DB per i metadati email e Application Insights per logging e metriche.
- **Integrazione casella email**
  Collegamento via API (IMAP/Graph) a una inbox condivisa, con pre-processing di testo e gestione allegati.
- **Pipeline LLM**
  Definizione e test di prompt per:
  - classificazione base (rispondere / inoltrare)
  - classificazione avanzata (tipologie di richiesta)
  - generazione automatica di bozza di risposta
- **Dashboard di revisione**
  Interfaccia minimale (FastAPI + Next.js) per visualizzare email, categorie proposte, bozza risposta e invio feedback al sistema.
- **Report finale PoC**
  Riepilogo delle metriche operative (percentuale di email automatizzate, tempi di elaborazione) e raccomandazioni per un eventuale scaling.

3. Architettura

La seguente sezione descrive in dettaglio l'architettura prevista per la Proof of Concept di classificazione automatica delle email nell'area "assistenza iscritti". L'infrastruttura sarà ospitata su Microsoft Azure, sfruttando servizi cloud per garantire prestazioni elevate, sicurezza avanzata e la possibilità di evolvere in modo graduale senza interruzioni dell'operatività.

3.1 Panoramica dell'Architettura

L'architettura si articola in quattro layer funzionali, ciascuno specializzato in un compito chiave:

- **Layer di Ingestion:** responsabile dell'acquisizione e del pre-processing delle email in ingresso.
- **Layer di Storage:** dedicato alla persistenza sicura e indicizzata dei metadati e dei raw email file.
- **Layer di AI:** cuore computazionale, esegue la classificazione e genera le bozze di risposta sfruttando modelli LLM.
- **Layer Applicativo:** fornisce l'interfaccia di revisione, il feedback loop e gli endpoint API per orchestrare l'intero processo.

Nel complesso, questo design modulare favorisce l'indipendenza di ciascun componente, consentendo di scalare selettivamente i servizi in base al carico e di integrare in futuro ulteriori funzioni (es. analisi del sentiment, estrazione automatica di KPI).

3.2 Layer di Ingestion

Questo strato garantisce un flusso continuo e controllato di email in ingresso:

1. **Connessione alla casella email**

    - Azure Logic App o Azure Function si autenticano via IMAP (Exchange) o Microsoft Graph API (Office 365).
    - Configurazione di polling schedulato (es. ogni 5 minuti) per prelevare solo i nuovi messaggi.
2. **Filtro temporale e di deduplicazione**
    - Vengono considerate solo le email ricevute negli ultimi 5–7 giorni.
    - Controllo ID messaggio per evitare rielaborazioni multiple.
3. **Pre-processing del contenuto**
    - Rimozione di elementi non testuali (HTML tag, firme, boilerplate standard).
    - Estrazione di campi chiave: mittente, destinatario, oggetto, data/ora, allegati.
    - Normalizzazione del testo (tokenizzazione di base, rimozione stop-word) per ottimizzare le successive chiamate AI.
4. **Gestione errori e retry**
    - Log di tutte le email non processate per anomalie (formato non supportato, timeout).
    - Meccanismo di retry automatico su failure temporanee, con escalation via Application Insights in caso di errori persistenti.

3.3 Layer di Storage

I dati vengono organizzati e conservati in due servizi Azure:

- **Azure Cosmos DB for Mongo (NoSQL)**
    - **Metadati email:** ID univoco, mittente, oggetto, timestamp, categorie proposte, confidence score, stato di revisione.
    - **Log feedback:** ogni modifica operatore genera un record che alimenta il feedback loop.
    - **Indexing flessibile:** chiavi di partizionamento su tenant/area e indice secondario su stato di elaborazione per query rapide.
    - **Scalabilità dinamica:** throughput configurabile in base al peak di elaborazione durante la PoC.
- **Azure Blob Storage**

- **Raw emails (.eml):** conservazione dei file originali per audit e eventuale reprocessing.
- **Allegati:** storage separato per dati sensibili o documenti complessi da analizzare in futuro.
- **Lifecycle management:** regole di lifecycle per spostare in cold tier dopo 30 giorni, minimizzando i costi.

3.4 Layer di AI

Il layer di AI ospita la logica di classificazione e generazione delle bozze di risposta:

1. **Azure OpenAI Service**
   - Engine di fornitura di Large Language Models per utilizzare lo stato dell'arte dei modelli generativi.
2. **Pipeline di classificazione**
   - **Stage 1 – Categorizzazione superficiale:** template prompt per definire "da rispondere/non da rispondere" e "inoltro reparto X/Y/Z".
   - **Stage 2 – Categorizzazione approfondita:** prompt dedicati per riconoscere richieste specifiche (anticipazioni, contributi, variazioni anagrafiche, reclami).
   - **Stage 3 – Bozza di risposta:** generazione di un draft basato sull'analisi semantica dell'email, pronto per la revisione.
3. **Confidence Scoring & Thresholding**
   - Ogni output include un punteggio di affidabilità.
   - Soglie configurabili per decidere se inviare direttamente al layer applicativo o segnalare manual review.
4. **Logging e metriche**
   - Tutte le chiamate e le risposte AI vengono tracciate in Application Insights per analisi di latenza, consumo risorse e accuratezza.

3.5 Layer Applicativo

Questo layer costituisce il punto di contatto con l'operatore:

- **Backend (FastAPI su App Service)**
  - Endpoint REST per:
    - Avvio manuale o schedulato dei job di ingestion
    - Invio delle email al modulo AI e ricezione dei risultati
    - Registrazione del feedback operativo
  - Private Endpoint integrato nella VNet per massima sicurezza.

- **Frontend (Next.js su Static Web App)**
  - **Dashboard di revisione:** elenco email con: categoria proposta, confidence score, snippet evidenziati e draft di risposta.
  - **Filtri dinamici:** per stato, categoria, data, punteggio.
  - **Pannello di correzione:** modifica etichette e testo bozza, invio diretto al sistema di mailing.
  - **Reportistica in tempo reale:** grafici sul numero di email processate, percentuali di automazione, trend di accuratezza.

3.6 Autenticazione, Sicurezza e Osservabilità

- **Autenticazione e Autorizzazione**
  - OAuth2 / JWT per accesso API e dashboard, con ruoli distinti (operatore, admin).
  - Segreti e chiavi di accesso gestiti in Azure Key Vault, accessibile via Managed Identity.

- **Rete e isolamento**
  - VNet dedicata, con Private Endpoints per App Service, Cosmos DB e Blob Storage.
  - Nessuna risorsa esposta pubblicamente: tutto il traffico interno rimane confinato.

- **Monitoraggio e Alerting**
    - **Azure Application Insights:** raccolta di telemetria su performance, errori, tempi di risposta.
    - **Azure Monitor Alerts:** notifiche automatiche in caso di anomalie (es. throughput inferiore al previsto, errori > 1 %).
    - **Dashboard condivisa:** accessibile a team di sviluppo e operation per analisi continua.

4. Struttura Economica

Di seguito si riporta il dettaglio dei costi complessivi di progetto relativi alle macro attività di progetto:

Descrizione	Effort (gg)	Costo (€)
Analisi Funzionale	10	3.500
Sviluppo Interfaccia Web	20	7.000
Sviluppo app backend e Ai workflows	20	9.000
Totale		~~**19.500€**~~
Totale Scontato		**18.000**

Qualora il progetto dovesse risultare soddisfacente, predisporremo una valutazione dettagliata dell'effort necessario per un eventuale rollout in produzione, con focus sui processi di integrazione, sulle misure di sicurezza da rafforzare e sulla portabilità dell'intero sistema in ambienti on-premise o private cloud, a seconda delle esigenze del cliente.

È importante sottolineare che **il costo complessivo del sistema può variare sensibilmente in funzione del numero di utenti simultanei previsti**: un'analisi funzionale preliminare sarà necessaria per dimensionare correttamente le risorse e stimare l'impatto effettivo sull'infrastruttura, soprattutto in scenari di utilizzo intensivo.

Tutti i prezzi sono da considerarsi IVA esclusa.

La seguente offerta è **valida per 30 giorni** dalla data di emissione

zendata



info@zendata.it

www.zendata.it

Via Nairobi 40
00144, Roma (RM)
info@zendata.it

zendata

WWW.ZENDATA.IT