Dai Documenti ai Dati: estrarre valore con LLM, Embeddings e Human-in-the-Loop senza farsi prosciugare il portafoglio

29 Luglio 2026 - 5 min. read

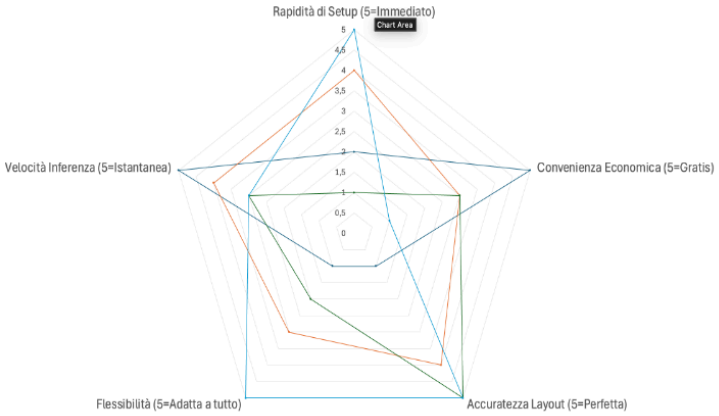
L'utilizzo dell'AI Generativa ormai sembra essere diventato un *must*, ma poi ci si scontra con la realtà: ogni azienda siede su una montagna di PDF, scansioni storte e bollette che nascondono dati preziosi. Tirarli fuori su scala *Enterprise* solleva due grandi problemi: come farlo in modo preciso e, soprattutto, come farlo senza che il CFO svenga quando arriva la fattura di AWS alla fine del mese?

Insieme a NeN, fornitore italiano di energia, 100% green e digitale, abbiamo affrontato proprio questa sfida. Ecco come siamo passati da un PoC "Pure LLM" (bellissimo, ma costoso) a un'architettura intelligente, ibrida e auto-apprendente.

La Sfida Iniziale: il fascino (e i limiti) del "Pure LLM"

Quando devi estrarre dati da un documento, il mercato ti mette davanti a quattro opzioni:

1. **OCR Tradizionale:** Economico e veloce. Se il layout cambia di un millimetro, però, va nel panico. Zero comprensione semantica.
2. **Servizi Managed (Feature extraction):** Pronti all'uso, ma poco flessibili se hai casistiche particolari (e occhio al vendor lock-in).
3. **Modelli ibridi Fine-tuned:** Ottimi per l'analisi spaziale, ma lenti su carichi massicci e poco inclini a digerire formati mai visti prima.
4. **Large Language Models (LLM):** La bacchetta magica. Flessibilità totale, capiscono il contesto e beccano i dati anche se la scansione è fatta male. Unico difetto? Costano cari.



Nel nostro early-stage, puntavamo tutto sulla velocità di setup e sulla massima versatilità. Quindi, abbiamo scelto la strada "tutto LLM". Abbiamo costruito un *ground truth* di 20 bollette (luce, gas, dual, PDF nativi e scansioni) e abbiamo messo alcuni modelli sul ring.

LLM Selection: vietato tirare a indovinare

Scegliere un modello "a sensazione" non è un'opzione accettabile in ingegneria. È il metodo rigoroso basato sui dati che trasforma una scommessa in una decisione architetturale solida. Ecco il processo che abbiamo seguito:

- **Set di validazione:** Senza una verità di riferimento (*ground truth*), non stai misurando assolutamente niente. Abbiamo usato le nostre 20 bollette eterogenee per coprire casistiche reali: provider diversi, versioni multiple della bolletta per lo stesso fornitore, mix di formati (immagine e PDF) e tipologie di fornitura (gas, luce, dual).
- **Prompt Engineering:** Abbiamo scritto il prompt per fornire istruzioni chiare e inequivocabili al modello.
- **La selezione dei contendenti:** Abbiamo individuato un subset di 4 modelli generativi della stessa "portata", ideali per il task. Nello specifico, abbiamo calato nell'arena di test **Amazon Nova 2 Lite**, **Amazon Nova Pro**, **Claude 3 Haiku** e **Claude 4.5 Sonnet**.
- **Il Benchmark spietato:** Abbiamo fatto girare i modelli sul set di validazione per estrarre risultati oggettivi.

Il vincitore assoluto per rapporto qualità/prezzo? **Claude 4.5 Sonnet**, orchestrato ovviamente tramite **Amazon Bedrock**.

Tutto molto bello. Ma quanto mi costa?

Il PoC è andato su in un attimo: disaccoppiamento con **Amazon API Gateway**, storage su **Amazon S3**, **AWS Lambda** a fare da collante e Bedrock a fare la magia. Il sistema estraeva i dati con una precisione spaventosa. Ma c'era un problema: usare un LLM di fascia alta per *ogni singola bolletta*, specialmente su volumi di produzione, non è finanziariamente sostenibile. È come usare una Ferrari per andare a fare la spesa dietro l'angolo.

Dovevamo ottimizzare. E per farlo, abbiamo guardato in faccia i nostri dati. Le bollette hanno due caratteristiche chiave:

1. I formati di input sono pochi (PDF nativo o immagine).
2. I template sono ripetitivi: le bollette provenienti da stesso provider ragionevolmente avranno layout comune (a meno di un cambio di versione).

Perché pagare l'LLM per far fatica a capire una struttura che in realtà abbiamo già visto?

Il pivot: Embeddings e il ritorno dell'OCR Classico

Serviva un modo per capire istantaneamente se un documento apparteneva a un template noto. I modelli di computer vision puri sono veloci ma fragili, i modelli multimodali sono troppo pesanti. La soluzione? **L'algoritmo di Embedding**. Trasformiamo il template della bolletta in un vettore numerico (un'impronta digitale oggettiva). Per capire se una nuova bolletta corrisponde a un layout noto, basta calcolare la *cosine similarity* tra vettori. Rapido, indolore ed economico.

Abbiamo così progettato un'architettura **Versione 2.0** a due velocità:

- **Routing:** Una **AWS Lambda** prende il file, lo valida e, se serve, fa pre-processing sulle immagini.
- **Orchestrazione:** Qui entra in gioco **AWS Step Functions Express**, che gestisce la logica di confronto vettoriale cercando le "firme" dei template già salvati su **Amazon DynamoDB**.

Da qui, il flusso si divide:

- **La Fast Track (Layout noto):** Il template esiste già. Invece di svegliare l'LLM, deviamo il traffico su una Lambda *self-contained* che usa il caro, vecchio OCR classico.

Sapendo già dove guardare (le coordinate spaziali sono salvate a DB), l'OCR estrae il dato in millisecondi a un costo ridicolo.

- **La Discovery Track (Layout sconosciuto):** Il template è nuovo. Chiamiamo **Amazon Bedrock**. L'LLM estrae semanticamente le info, ma non fa solo quello: mappiamo i *bounding boxes* (le coordinate delle chiavi) e calcoliamo l'embedding del nuovo documento. Salviamo questa "nuova firma" su DynamoDB. Dalla bolletta successiva, questo stesso layout andrà in Fast Track!

Il risultato? Un ecosistema auto-apprendente. I costi crollano man mano che il sistema immagazzina nuovi template, e l'LLM lavora solo quando serve davvero materia grigia.

Human-in-the-Loop: la fiducia è bene, il controllo è meglio

Delegare tutto a un algoritmo in produzione è il modo più rapido per non dormire la notte. Per questo abbiamo inserito un pattern **Human-in-the-Loop (HITL)** come perno della governance:

- **Tracciabilità:** Ogni dato estratto ha la sua impronta digitale spaziale e un rigoroso *confidence score*.
- **Backoffice Serverless:** Abbiamo tirato su una Single Page Application in React (hostata su **S3** e distribuita globalmente in CDN con **CloudFront**). Qui gli operatori possono validare visivamente i dati sovrapposti all'immagine originale della bolletta. Le API backend? Ovviamente Lambda.
- **Alerting Proattivo:** Cosa succede se Eni cambia improvvisamente il layout delle fatture? Il confidence score medio crolla. **Amazon CloudWatch** se ne accorge, fa scattare un allarme e **Amazon SNS** inonda di email il team di ingegneria prima che il business se ne renda conto.

Takeaways

L'AI generativa non è la soluzione a tutto, è uno strumento. La maturità architetturale nel Cloud non si dimostra sparando Foundation Models su ogni singolo task sperando che funzioni. La vera Ingegneria (con la I maiuscola) sta nell'orchestrazione: usate i **Large Language Models** solo per la pura capacità cognitiva, affidatevi all'**OCR classico** per spingere l'efficienza sui task deterministici, e usate gli **Embeddings** per il routing intelligente.

In questo modo avrete un'architettura che non solo scala da paura, ma impara da sola, vi lascia il pieno controllo sui dati e fa sorridere il vostro FinOps.

About Proud2beCloud

Proud2beCloud is a blog by beSharp, an Italian APN Premier Consulting Partner expert in designing, implementing, and managing complex Cloud infrastructures and advanced services on AWS. Before being writers, we are Cloud Experts working daily with AWS services since 2007. We are hungry readers, innovative builders, and gem-seekers. On Proud2beCloud, we regularly share our best AWS pro tips, configuration insights, in-depth news, tips&tricks, how-tos, and many other resources. Take part in the discussion!

Paolo Serale

Senior Data & AI Project Manager in beSharp. Guido aziende e team nell'adozione di soluzioni Cloud avanzate e AI.Con una solida esperienza nella gestione di progetti complessi, trasformo grandi volumi di dati in valore strategico e innovazione concreta.Speaker ed evangelist tech, condivido la mia passione per il futuro della tecnologia attraverso eventi di settore e community.

Copyright © 2011-2026 by beSharp spa - P.IVA IT02415160189