Modello di Costo per un Messaggio in Chat

Di seguito è presentato il modello di calcolo del costo per singolo messaggio (domanda+risposta), includendo tutti i *parametri* necessari a determinare la dimensione dell'*input* al modello GPT (storia della chat, risultati di ricerca, ecc.) e dell'*output* generato, oltre ai costi di *retrieval* e *salvataggio*. Questo *focalizza la stima* esclusivamente sul *costo di un messaggio* e *non ripete* il resto dei calcoli mensili o dell'ingestion dei contenuti.

1. Componenti di Costo

Per ciascun **messaggio** (inteso come "turno utente + risposta AI"), il costo totale C_{msg} è dato dalla somma di:

$$C_{\text{msg}} \;=\; C_{\text{LLM}} \;+\; C_{\text{retrieval}} \;+\; C_{\text{store}}.$$

- C_{LLM}: costo di inferenza del modello GPT-4o/Mini, calcolato in base ai **token di input** (prompt) e **token di output** (risposta).
- $C_{\text{retrieval}}$: costo relativo alla **ricerca vettoriale** (embedding della query, query su KBox, lettura chunk).
- C_{store}: costo di **scrivere** nel database il testo del messaggio e la risposta (o eventuali embedding aggiuntivi se indicizziamo la conversazione).

2. Calcolo dei Token di Input

2.1 Storia della Chat Si assume che la chat mantenga un **numero massimo di coppie** di messaggi (utente+AI) in contesto — ad esempio, *25 coppie*. Se ciascuna coppia di messaggi ha una lunghezza media di T_{history} token per messaggio (sommando utente e AI, o separandoli a seconda del design), allora la parte di "storia" inclusa nel prompt vale:

- Numero di messaggi di "storia" effettivamente inclusi: fino a $25 \times 2 = 50$ messaggi, ma in molti casi si condensano i precedenti o si limita a 25 scambi totali.
- Token totali di storia $T_{\text{history\_total}} = (\#\text{ messaggi di contesto}) \times T_{\text{history}}$.

Se semplifichiamo assumendo 25 coppie complete e lunghezza media $\bar{t}_{\text{hist}}$ per messaggio, i token "storia" sono:

$$T_{\text{history\_total}} \;=\; 25 \;\times\; 2 \;\times\; \bar{t}_{\text{hist}} \;=\; 50\,\bar{t}_{\text{hist}}.$$

(Se la chat è più breve, si userà un numero minore; se superasse 25 coppie, si taglia la parte più vecchia.)

2.2 Input Utente Corrente Al di là della storia passata, l'utente invia il **nuovo messaggio** (la "domanda"):

- Se ha lunghezza media $\bar{t}_{\text{user}}$ token, questa si somma direttamente al prompt.

2.3 Risultati di Retrieval dalle KBox Se l'app fa **RAG** (Retrieval Augmented Generation), si calcolano i chunk di testo provenienti da una o più KBox. Supponiamo:

- N_{kbox} = numero medio di KBox interrogate (es. 1,5).
- $R_{\text{per\_kbox}}$ = numero di chunk ("risultati") recuperati per ogni KBox (es. 3).
- $\bar{t}_{\text{chunk}}$ = lunghezza media di ciascun chunk (es. 300 token).

Il totale di token "di contesto" aggiunti dal retrieval è:

$$T_{\text{retrieval}} \;=\; N_{\text{kbox}} \;\times\; R_{\text{per\_kbox}} \;\times\; \bar{t}_{\text{chunk}}.$$

(Esempio: 1,5 KBox × 3 chunk × 300 token = 1350 token di contesto da documenti.)

2.4 Totale Token di Input Sommando storia + messaggio utente + chunk di retrieval:

$$T_{\text{in}} \;=\; T_{\text{history\_total}} \;+\; \bar{t}_{\text{user}} \;+\; T_{\text{retrieval}}.$$

3. Calcolo dei Token di Output

La risposta generata dal modello ha una lunghezza media $\bar{t}_{\text{out}}$ (ad esempio $\sim$300 token). In caso di risposte più lunghe, ovviamente il costo cresce linearmente.

4. Costo di Inferenza LLM (C_{LLM})

Avendo T_{in} token in input e T_{out} token in output, e definendo:

- p_{in} = costo per token di input (es.: GPT-4o 0,005 \$/1k, GPT-4o Mini 0,00015 \$/1k),
- p_{out} = costo per token di output (es.: GPT-4o 0,015 \$/1k, GPT-4o Mini 0,00060 \$/1k),

allora:

$$C_{\text{LLM}} \;=\; \frac{T_{\text{in}}}{1000}\, p_{\text{in}} \;+\; \frac{T_{\text{out}}}{1000}\, p_{\text{out}}.$$

*(Se usiamo un modello **misto** con frazione f di messaggi su GPT-4o full e $(1-f)$ su Mini, si fa la media pesata dei costi. Ma per **il singolo messaggio** in quell'istante useremo i parametri del modello selezionato.)*

5. Costo di Retrieval ($C_{\text{retrieval}}$)

- **Embedding** della query utente ($\sim \bar{t}_{\text{user}}$ token, costo $\sim \bar{t}_{\text{user}} \times C_{\text{embed}}$). Spesso ¡ 10^{-5} \$.
- **Query vettoriale** su Atlas (v3): $\sim 10^{-6}$–10^{-5} \$.
- **Lettura chunk** (d4): qualche microcentesimo su base di dimensioni ridotte.

In genere si approssima $C_{\text{retrieval}} \approx 10^{-5}$ \$ (trascurabile rispetto a C_{LLM}).

6. Costo di Salvataggio (C_{store})

- **Scrittura** del nuovo messaggio utente + della risposta AI (2 doc), ciascuno $\sim \bar{t}_{\text{user}}$ e $\bar{t}_{\text{out}}$ token. In Atlas, 1–2 WPU totali $\sim 10^{-6}$ \$.
- Eventuale **embedding** della conversazione se la si indicizza $\rightarrow$ qualche token in embedding. Spesso è anch'esso dell'ordine di 10^{-5} \$.

7. Formula Riassuntiva per il Costo del Messaggio

$$\boxed{C_{\text{msg}} \;=\; \underbrace{\frac{T_{\text{in}}}{1000}\, p_{\text{in}} \;+\; \frac{T_{\text{out}}}{1000}\, p_{\text{out}}}_{C_{\text{LLM}}} + C_{\text{retrieval}} \;+\; C_{\text{store}}.}$$

Dove:

- $T_{\text{in}} \;=\; 50\,\bar{t}_{\text{hist}} \;+\; \bar{t}_{\text{user}} \;+\; \big(N_{\text{kbox}} \times R_{\text{per\_kbox}} \times \bar{t}_{\text{chunk}}\big)$ (nell'esempio con 25 coppie massime),
- $T_{\text{out}} \;\approx\; \bar{t}_{\text{out}}$,
- $C_{\text{retrieval}}$ e C_{store} sono piccoli (embedding query + ricerche + scritture DB), di solito $\sim 10^{-5}$ \$ complessivi.

Esempio Numerico

- **25 coppie** di messaggi in storia, ciascuno $\bar{t}_{\text{hist}} = 100$ token $\rightarrow 50 \times 100 = 5000$ token.
- **Input utente** $\bar{t}_{\text{user}} = 50$ token.
- **KBox**: $N_{\text{kbox}} = 1,5$, $R_{\text{per\_kbox}} = 3$, $\bar{t}_{\text{chunk}} = 300 \rightarrow 1,5 \times 3 \times 300 = 1350$ token di contesto.
- **Totale input** $= 5000 + 50 + 1350 = 6400$ token.
- **Output**: $\bar{t}_{\text{out}} = 300$ token (esempio).

Caso GPT-4o Full Con $p_{\text{in}} = 0,005$ e $p_{\text{out}} = 0,015$:

$$C_{\text{LLM}} = \frac{6400 \times 0{,}005}{1000} + \frac{300 \times 0{,}015}{1000} = 0{,}032 + 0{,}0045 = 0{,}0365 \text{ \$.}$$

(3,65 centesimi). A cui si aggiunge retrieval/store $\sim 10^{-5}$ \$ $\rightarrow$ totale $\sim 0{,}03651$ \$.

Caso GPT-4o Mini Con $p_{\text{in}} = 0,00015$ e $p_{\text{out}} = 0,00060$:

$$C_{\text{LLM}} = \frac{6400 \times 0{,}00015}{1000} + \frac{300 \times 0{,}00060}{1000} = 0{,}00096 + 0{,}00018 = 0{,}00114 \text{ \$.}$$

(0,114 centesimi). Sommando retrieval/store $\approx 0{,}00115$ \$ per messaggio.

In sintesi, questo è il *modello dettagliato* per calcolare il costo di **un singolo messaggio in chat**, tenendo conto di:

1. **Storia massima** (ad es. 25 coppie) $\rightarrow T_{\text{history\_total}}$.
2. **Input utente** $\rightarrow \bar{t}_{\text{user}}$.
3. **Contenuto di retrieval** (numero KBox $\times$ chunk) $\rightarrow T_{\text{retrieval}}$.
4. **Output** generato $\bar{t}_{\text{out}}$.
5. **Costi** di embedding query, database, salvataggio ($C_{\text{retrieval}} + C_{\text{store}}$).

Il risultato finale è la formula (riportata nel riquadro) che, sostituendo i vari parametri, produce **il costo unitario** (in dollari) per la singola interazione (turno di domanda-risposta).

Descrizione della Formula Totale Generale

Vogliamo ora **descrivere tutti i parametri** coinvolti e presentare la *formula generale* con una *versione sintetica* e una *versione estesa* che mostra ogni sotto-parametro nel dettaglio.

Versione Sintetica

Indichiamo con:

- T_{in}: **token totali di input** (storia chat + messaggio utente + chunk retrieval),
- T_{out}: **token di output** (risposta generata),
- p_{in}, p_{out}: **costi per token** di input/output (dipendono dal modello GPT-4o vs GPT-4o Mini),
- $C_{\text{retrieval}}$: **costo retrieval** (embedding query + query store + letture),
- C_{store}: **costo salvataggio** (scritture DB).

Allora la formula *generica* è:

$$C_{\text{msg}} = \underbrace{\frac{T_{\text{in}}}{1000}\, p_{\text{in}} + \frac{T_{\text{out}}}{1000}\, p_{\text{out}}}_{\text{Costo LLM}} + C_{\text{retrieval}} + C_{\text{store}}.$$

Versione Estesa (con sotto-parametri)

Approfondiamo i *dettagli* di ciascun termine:

1. $T_{\text{in}} = T_{\text{history\_total}} + \bar{t}_{\text{user}} + \left(N_{\text{kbox}} \times R_{\text{per\_kbox}} \times \bar{t}_{\text{chunk}}\right)$
   - $T_{\text{history\_total}} = (\#\text{ coppie di storia} \times 2) \times \bar{t}_{\text{hist}}$, tipicamente ≤ 25 coppie,
   - $\bar{t}_{\text{user}}$: lunghezza media del messaggio utente in token,
   - N_{kbox}: numero di KBox coinvolte in media (es. 1,5),
   - $R_{\text{per\_kbox}}$: chunk restituiti da ciascuna KBox (es. 3),
   - $\bar{t}_{\text{chunk}}$: token medi per chunk (es. 300).
2. $T_{\text{out}} = \bar{t}_{\text{out}}$, lunghezza media della risposta LLM in token (es. 300).
3. $p_{\text{in}}, p_{\text{out}}$: costi per token *in* e *out* (dipende dal modello). Ad esempio:

$$\begin{aligned} \text{GPT-4o: } & p_{\text{in}} \approx 0,005 \text{ \$/1k}, \quad p_{\text{out}} \approx 0,015 \text{ \$/1k}, \\ \text{GPT-4o Mini: } & p_{\text{in}} \approx 0,00015 \text{ \$/1k}, \quad p_{\text{out}} \approx 0,00060 \text{ \$/1k}. \end{aligned}$$

4. $C_{\text{retrieval}}$ copre:
   - l'embedding della query (costo $\approx \bar{t}_{\text{user}} \times C_{\text{embed}}$),
   - la query vettoriale (pochi *RPU* su Atlas),
   - la lettura dei chunk dal DB (pochi *RPU*).

   Spesso stimato come costante di $\sim 10^{-5}$ \$.
5. C_{store} copre:
   - la scrittura di messaggio utente + risposta nel DB (1–2 WPU),
   - eventuale embedding della conversazione.

   Anch'esso $\sim 10^{-5}$ \$.

Formula Estesa Finale:

$$\boxed{C_{\text{msg}} = \left[\left(T_{\text{history\_total}} + \bar{t}_{\text{user}} + N_{\text{kbox}} \times R_{\text{per\_kbox}} \times \bar{t}_{\text{chunk}}\right) \frac{p_{\text{in}}}{1000}\right] \; + \; \left[\bar{t}_{\text{out}} \times \frac{p_{\text{out}}}{1000}\right] \; + \; C_{\text{retrieval}} \; + \; C_{\text{store}}.}$$

Significato dei Parametri (riassunto):

- $\bar{t}_{\text{hist}}$: n. token medi per messaggio nella *storia* (p.es. 100).
- $\bar{t}_{\text{user}}$: n. token medi di un messaggio utente (p.es. 50).
- $\bar{t}_{\text{out}}$: n. token medi della risposta LLM (p.es. 300).
- N_{kbox}: media di KBox coinvolte per query (p.es. 1,5).
- $R_{\text{per\_kbox}}$: chunk per KBox (p.es. 3).
- $\bar{t}_{\text{chunk}}$: n. token in un chunk (p.es. 300).
- $p_{\text{in}}, p_{\text{out}}$: costi per 1k token in input/output.
- $C_{\text{retrieval}}$: stima fissa per embedding della query e RPU (p.es. 10^{-5} \$).
- C_{store}: stima fissa per scritture DB e embedding conversazione (p.es. 10^{-5} \$).