



STUDIO
LEGALE
FABIANO



CNEF Consiglio
Nazionale
Forense



UNU

Legal Prompting

Come "*parlare*" ai sistemi di AI

per ottenere risposte giuridiche precise, verificabili e utili

Avv. Nicola Fabiano
Studio Legale Fabiano

Obiettivi dell'incontro

- comprendere **perché** gli LLM si comportano in modo diverso dai software giuridici tradizionali;
- imparare una struttura di **prompt legale efficace**;
- distinguere tra uso utile, uso rischioso e uso improprio degli strumenti di AI;
- collegare prompting, **RAG**, privacy, sicurezza e **AI Act**;
- definire un metodo operativo: **prompt** → **verifica** → **utilizzo professionale**.

Il punto centrale non è "usare l'AI", ma usarla con metodo giuridico, limiti chiari e controllo umano.

Come lavoreremo oggi

Guida essenziale → Prove sul campo → Metodo finale

1. Guida essenziale

- cosa sono LLM e legal prompting
- come si costruisce un buon prompt
- quali rischi evitare

2. Prove sul campo

- stesso task con prompt diversi
- stesso prompt su modelli diversi
- verifica errori, omissioni, allucinazioni, utilità pratica

3. Ritorno alla guida

- cosa abbiamo imparato dalla demo
- regole operative per lo studio
- checklist finale di uso professionale

Perché gli LLM sono diversi dai software giuridici tradizionali

Caratteristica	Software tradizionale	LLM generativo
Logica	Regole deterministiche	Modello probabilistico
Output	Prevedibile	Variabile e contestuale
Errore tipico	Tecnico e riproducibile	Plausibile ma non sempre corretto
Aggiornamento	Nuova versione	Nuovo prompt / nuovo contesto
Valore professionale	Esecuzione	Supporto al ragionamento

Un LLM può sbagliare in modo convincente. Per questo il legal prompting e la verifica umana non sono opzionali.

Che cos'è il legal prompting

Prompt = insieme di istruzioni fornite al modello per ottenere un output utile.

Nel diritto, il prompt non è solo un comando tecnico: è una forma di **argomentazione giuridica strutturata**.

Elemento	Equivalente professionale
Prompt	quesito / istruzione
Contesto	fatti, documenti, materiali rilevanti
LLM	interprete linguistico
Output	bozza, sintesi, parere preliminare
Verifica umana	controllo professionale finale

I prompt diventano artefatti di lavoro: documentabili, migliorabili, versionabili e auditabili.

Il prompting è già nel DNA dei giuristi

Metodo del giurista	Equivalente nel prompting
qualificazione dei fatti	descrizione accurata del caso
individuazione della norma	definizione del quadro normativo
sussunzione	richiesta di applicazione della regola
motivazione	richiesta di reasoning verificabile
conclusione	output strutturato e utilizzabile

Il giurista non deve imparare un linguaggio alieno: deve rendere esplicito il metodo che già usa.

IRAC come struttura naturale del prompt giuridico

Lettera	Significato	Funzione nel prompt
I	Issue	definire la questione
R	Rule	indicare la regola applicabile
A	Application	applicare la regola ai fatti
C	Conclusion	formulare la conclusione

Esempio sintetico:

Issue: il trattamento di dati sanitari per ricerca è ammissibile?
Rule: art. 9(2)(j) GDPR + art. 89 GDPR
Application: il trattamento richiede garanzie adeguate e misure tecniche/organizzative
Conclusion: ammissibile solo entro il perimetro delle deroghe e con adeguate salvaguardie

Prima della demo: la domanda giusta

Non chiedere: "Qual è il migliore LLM?"

Chiedi invece:

- **Quale modello ragiona meglio su un caso giuridico concreto?**
- **Quale segue davvero le istruzioni?**
- **Quale inventa meno?**
- **Quale gestisce meglio fonti, limiti e formato?**
- **Quale è compatibile con privacy, budget e workflow dello studio?**

***Nella demo non guardiamo chi "scrive meglio".
Guardiamo chi è più **utile, controllabile e verificabile**.***

Modelli disponibili: una mappa essenziale

Categoria	Esempi	Punti forti	Attenzione
Cloud proprietari	Claude 4, GPT-5.4, Gemini 2.5 Pro, Grok 4	Qualità elevata, tool integrati, context window ampia	Dati verso server terzi, policy di trattamento
Open-weight / locali	Llama 4, DeepSeek-R1, Qwen3, Mistral	Controllo totale, privacy, fine-tuning possibile	Setup tecnico, requisiti hardware
Runtime locali	Ollama, LM Studio	Esecuzione locale, testing rapido, costo zero	Sono strumenti di esecuzione, non modelli

Regola pratica: dati sensibili o coperti da segreto professionale → preferire anonimizzazione forte o soluzioni locali. **Token** e **temperature** sono i parametri chiave: token = lunghezza contesto, temperature = riproducibilità output (usare 0.0 per atti professionali).

Cosa osservare nella demo

Non guardiamo se l'AI "scrive bene".

Guardiamo se è **professionalmente affidabile**.

Criteri di valutazione

- **Aderenza al task** — fa esattamente quello che gli è stato chiesto?
- **Rispetto dei vincoli** — segue le istruzioni o le aggira?
- **Qualità del ragionamento** — il ragionamento è verificabile?
- **Gestione delle fonti** — cita fonti reali o plausibili?
- **Formato realmente utile** — l'output è direttamente utilizzabile?
- **Rischio di invenzione** — quante verifiche sono necessarie?
- **Bisogno di correzione umana** — quanto lavoro resta all'avvocato?

⚠ Un output elegante non è necessariamente un output corretto.

Materiali della prova pratica

Durante la demo vedremo sempre tre elementi

1. **Input**

- testo del prompt
- eventuali documenti o estratti forniti
- eventuali vincoli normativi

2. **Output**

- risposta del modello
- formato prodotto
- eventuali fonti o riferimenti citati

3. **Valutazione**

- cosa funziona
- cosa manca
- cosa va verificato prima dell'uso professionale

Non guardiamo la demo come spettatori.

La guardiamo come se dovessimo decidere se usare davvero quell'output in studio.

Primo test sul campo

Stesso contratto. Due prompt. Due risultati diversi.

Prompt debole

Analizza questo contratto.

Effetto atteso:

- risposta generica;
- focus incerto;
- nessun controllo sulle fonti;
- formato poco utile.

La demo parte da qui.

Prompt efficace

Sei un avvocato senior B2B. Analizza questa bozza SaaS lato cliente. Individua: 1. limitazioni di responsabilità sbilanciate; 2. criticità GDPR; 3. lacune su sicurezza e continuità. Output: tabella (Clausola | Rischio | Revisione). Cita solo norme fornite; se manca, scrivi "da verificare".

Effetto atteso:

- output mirato e strutturato;
- formato subito usabile;
- rischio ridotto.

Demo 1 – Stesso modello, prompt diversi

Obiettivo


Vedere quanto cambia l'output quando cambia il prompt.

Task

Analizzare una clausola SaaS lato cliente.

Confronteremo

- **Prompt A** → generico
- **Prompt B** → strutturato
- **Prompt C** → strutturato + vincoli + formato + fonti

 **Domanda guida:** il modello è "bravo" o è il prompt a guidarlo bene?

Demo 2 – Stesso prompt, modelli diversi

Obiettivo

Vedere come modelli diversi reagiscono allo stesso task legale.

Testeremo

- aderenza alle istruzioni
- precisione del linguaggio
- tendenza a inventare
- chiarezza del formato
- necessità di revisione umana

 **Domanda guida:** tutti gli LLM sembrano forti. Ma si comportano davvero allo stesso modo su un caso giuridico concreto?

Demo 3 – Quando l'AI sembra sicura ma sbaglia

Obiettivo

Mostrare un caso in cui l'output è plausibile, ben scritto, ma giuridicamente fragile o falso.

Cosa cercheremo

- riferimenti normativi non verificati
- falsa sicurezza espositiva
- omissioni su eccezioni o limiti
- conclusioni troppo assertive

**⚠ Il rischio professionale non è l'errore grossolano.
È l'errore **credibile**.**


Demo 4 – Lo stesso caso con e senza anonimizzazione

Obiettivo

Mostrare che il legal prompting non riguarda solo la qualità dell'output, ma anche la **protezione del dato**.

Confronteremo

- prompt con dati identificativi
- prompt anonimizzato con framework LegalGuardian
- differenza di utilità pratica
- differenza di esposizione al rischio GDPR e deontologico

 **Punto chiave:** un buon prompt legale è anche un prompt **privacy-aware**.

La griglia che abbiamo usato nella demo

Quando costruiamo o valutiamo un prompt legale, verifichiamo sempre:

1. **Ruolo** → chi deve essere il modello
2. **Contesto** → fatti, settore, ordinamento
3. **Destinatario** → per chi è scritto l'output
4. **Task** → cosa deve fare esattamente
5. **Vincoli** → norme, limiti, esclusioni
6. **Metodo** → IRAC, checklist, tabella, confronto
7. **Formato** → tabella, memo, elenco, JSON
8. **Fonti** → solo quelle fornite / "da verificare"
9. **Verifica** → punti critici, limiti, confidence
10. **Negative constraints** → cosa non deve fare

Mnemonic: Ruolo → Contesto → Destinatario → Task → Vincoli → Metodo → Formato → Fonti → Verifica → Limiti

Tecniche avanzate di legal prompting

- **Role prompting**: assegna expertise e prospettiva;
- **Few-shot prompting**: mostra esempi del formato desiderato;
- **Chain of Thought (CoT)**: chiedi ragionamento strutturato e verificabile;
- **Prompt chaining**: scomponi task complessi in passaggi verificabili;
- **Meta-prompting**: usa il modello per migliorare il prompt;
- **IRAC prompting**: struttura il task per fasi giuridiche esplicite.

Task complesso in un prompt unico = più rumore.

Task complesso in più passaggi = più controllo, più qualità, più verificabilità.

RAG – Definizione semplice per lo studio legale

Domanda del professionista



Recupero di documenti rilevanti (sentenze, contratti, norme)



Lettura dei documenti trovati



Sintesi / risposta basata sulle fonti



Verifica finale dell'avvocato

Senza RAG: il modello risponde sulla base del training (rischio allucinazioni).

Con RAG: il modello usa documenti esterni recuperati per quel task specifico.

RAG non elimina automaticamente le allucinazioni. Migliora il contesto, ma non sostituisce la verifica. Stanford Law School (2025): i sistemi RAG legali allucinano nel 17-33% dei casi.

Tre forme di RAG che l'avvocato deve distinguere

Tipologia	Vantaggio	Rischio
Generativo	rapido, apparentemente brillante	può inventare riferimenti normativi
Retrieval-only	recupera solo documenti nel database	richiede curatela della base documentale
Ibrido con validazione umana	più flessibile e contestuale	mai usare senza controllo finale

Vantaggi reali:

- accesso rapido a fascicoli, contratti, norme e precedenti interni;
- maggiore tracciabilità delle fonti usate.

Limiti reali:

- qualità dipendente dalla qualità del database;
- retrieval rumoroso, fonti obsolete o conflittuali;
- aumento della fiducia del modello anche quando sbaglia.

Privacy-Preserving Prompting – Framework LegalGuardian

Ogni documento incollato in un LLM cloud espone dati personali dei clienti con implicazioni su GDPR, segreto professionale e AI Act.

Il framework LegalGuardian (ricerca 2025) – precisione fino al **97%**:

- [1] Testo grezzo con dati personali
- ↓
- [2] NER locale → rileva nomi, CF, indirizzi, dati sanitari
- ↓
- [3] Anonimizzazione → "Mario Rossi" diventa "[PERSONA_1]"
- ↓
- [4] Prompt anonimizzato → LLM cloud
- ↓
- [5] Re-integrazione dei dati reali nel risultato finale

Livello	Metodo	Costo
Base	Anonimizzazione manuale prima del prompt	Zero
Intermedio	Tool locali (spaCy + regex) per NER automatica	Basso
Avanzato	Pipeline LegalGuardian-style su VM locale	Setup tecnico

Prompt injection – Il rischio che l'avvocato deve conoscere

Definizione: istruzione malevola nascosta in un documento che induce l'agente AI a comportarsi scorrettamente, all'insaputa dell'avvocato.

Tipologie di attacco:

- **Direct injection:** nel prompt stesso;
- **Indirect injection:** in documenti, email, pagine web analizzate dall'agente;
- **Data exfiltration:** l'agente viene indotto a trasmettere dati sensibili.

Contromisure minime:

- separare istruzioni di sistema e contenuti analizzati;
- limitare permessi di scrittura e invio;
- introdurre guardrails e approvazione umana;
- **Regola d'oro:** mai dare a un agente accesso a dati sensibili + capacità di trasmissione esterna simultaneamente.

Guardrails – Controllo dell'output

I guardrails bloccano o segnalano output problematici prima che raggiungano l'avvocato.

```
if contains_personal_data(output):  
    block() # Guardrail 1: No dati personali  
if citations_not_verified(output):  
    flag_for_review() # Guardrail 2: No citazioni inventate  
if action_requires_approval():  
    wait_human_confirmation() # Guardrail 3: Azione non autorizzata
```

Tool di riferimento:

- [NeMo Guardrails \(NVIDIA\)](#)
- [Guardrails AI \(open source\)](#)

In ambito legale il guardrail più efficace non è solo tecnico: è l'**obbligo di revisione umana** sui passaggi critici.

Agentic AI e MCP – Il prompting che agisce

LLM classico: riceve un prompt e genera testo.

Agente AI: riceve un obiettivo, pianifica, usa tool, verifica e produce un risultato.

Model Context Protocol (MCP) – standard Anthropic (2024): protocollo aperto che connette LLM a tool esterni (database, email, calendari, API).


Scenario	Cosa fa l'agente
Due diligence	legge fascicoli → estrae dati → compila report → salva in archivio
Monitoraggio normativo	controlla EUR-Lex → segnala novità rilevanti per i clienti
Gestione scadenze	legge calendario → redige bozza → invia reminder al cliente

Con agenti autonomi gli errori si propagano in cascata. La supervisione umana su ogni azione critica resta obbligatoria.

AI Act e studi legali – Obblighi operativi

Quando uno studio usa LLM internamente si qualifica come **deployer** ai sensi dell'AI Act (art. 3, n. 4).

Obbligo	Riferimento	Applicazione pratica
AI Literacy	Art. 4	Formazione obbligatoria per chi usa LLM
Trasparenza	Art. 50	Informare il cliente se il documento è stato redatto con supporto AI
DPIA	Art. 9 GDPR + Art. 26 AI Act	Analisi impatto se si trattano dati sanitari o giudiziari
Registro	Art. 49	Documentare i sistemi AI usati per supporto decisionale

 Redigere ora una **AI Policy interna di studio** che regoli: quali LLM sono autorizzati, per quali task, con quali limiti sui dati trattabili. È già una best practice deontologica e sarà presto un obbligo.

Cosa ci ha mostrato la prova pratica

Le evidenze emerse

- il modello non basta da solo;
- il prompt cambia davvero il risultato;
- l'output migliore è quello più **controllabile**, non quello più brillante;
- fonti, vincoli e formato riducono il rischio;
- la verifica umana resta il vero presidio professionale.

Dopo la demo, torniamo al metodo.

Il problema non è l'AI

Le tre cause reali degli errori AI in ambito legale:

1. **Prompt scadenti** – istruzioni vaghe, senza ruolo, senza vincoli normativi.
2. **Contesto incompleto** – il modello non conosce il caso, l'ordinamento, il cliente.
3. **Mancanza di verifica umana** – l'output viene usato senza controllo professionale.

Cambiare modello aiuta meno di quanto si creda. Migliorare **prompt, contesto e processo** aiuta molto di più.

Cosa abbiamo visto davvero nella prova pratica

Le evidenze emerse

- lo stesso modello cambia molto se cambia il prompt;
- modelli diversi non reagiscono allo stesso modo allo stesso task;
- l'output più elegante non è sempre il più affidabile;
- fonti, vincoli e formato riducono il rischio;
- la verifica umana resta il vero presidio professionale.

💡 La domanda corretta non è:

"Questa AI è intelligente?"

La domanda corretta è:

"Questo output è utilizzabile, controllabile e verificabile?"

Tre principi del legal prompting

 1. Il modello non conosce il diritto come un giurista.

 2. Il modello conosce il linguaggio del diritto.

 3. Il giurista deve guidarne il ragionamento.

Checklist finale per usare l'AI in studio

Prima di usare un output AI chiediti sempre:

- Ho definito bene **ruolo, contesto e task**?
- Ho imposto **vincoli normativi e formato**?
- Ho escluso dati che non dovevano uscire dallo studio?
- Le **fonti** sono reali o solo plausibili?
- L'output è **verificato** prima di essere usato?

✓ Il legal prompting non serve a "farsi scrivere le cose".

Serve a ottenere un primo output **controllabile, verificabile e professionalmente gestibile**.

Legal Prompting

Il futuro della professione legale è Human + AI

**"Non è l'AI che sostituirà gli avvocati,
ma gli avvocati che usano bene l'AI
supereranno chi la usa male o non la usa affatto."**

Keep learning, keep prompting, keep verifying.



Grazie per l'attenzione

Avv. Nicola Fabiano

 <https://www.fabiano.law> |  <https://www.nicfab.eu> |  <https://links.nicfab.eu>

