



MediLens

progetto di Gestione
dell'Informazione.

Matr. 119325. Candidata: Elena Maria Ciuffreda



1. Scelta del Dataset



Un po' di contesto..

Il dataset utilizzato è stato costruito estraendo 20 pagine di Wikipedia in lingua italiana relative ad ambiti clinici e patologie (influenza, diabete, ipertensione, bronchite, vaccini, COVID-19, cancro, ecc.), per un totale di circa 20 documenti con titolo e contenuto completi.

Questa scelta nasce dall'esigenza di lavorare su materiali autentici e diversificati, caratterizzati da terminologia specialistica, sinonimi e strutture linguistiche complesse.

In particolare, il dataset copre patologie acute e croniche, procedure preventive (come le vaccinazioni) e termini emergenti (ad es. COVID-19), offrendo un terreno di prova ideale per valutare come ciascun motore gestisca sinonimi ("influenza" vs. "febbre"), ambiguità semantiche e la necessità di bilanciare recall e precision.

L'impiego di Wikipedia garantisce inoltre una fonte aggiornata e liberamente accessibile, favorendo la riproducibilità del progetto e la replicabilità dei risultati in ambito accademico e di ricerca.




Risorsa per la Ricerca di Informazioni Sanitarie

- Il dataset scelto per questo progetto proviene dall'ambito medico, un settore in cui l'accuratezza e la pertinenza delle informazioni sono fondamentali.
- è indispensabile che un motore di ricerca restituisca risultati non solo rilevanti ma anche precisi, poiché un'informazione inesatta o fuori contesto può compromettere decisioni cliniche o la fiducia dell'utente.

A decorative graphic on the left side of the slide. It consists of a blue parallelogram and a light green parallelogram, both tilted at an angle. The blue shape is in the foreground, and the green shape is partially behind it. They are set against a dark blue background with faint, lighter blue diagonal stripes.

2. Architettura dei motori di ricerca.



L'architettura dei motori di ricerca si basa su 3 sistemi principali:

PostgreSQL sfrutta il modulo `tsvector` per trasformare i campi testuali in un indice full-text utilizzando `to_tsquery` e `plainto_tsquery` per tradurre le espressioni dell'utente in query interne. Il ranking delle risposte viene calcolato con la funzione `ts_rank_cd`, che implementa un algoritmo di tipo BM25-like (disponibile anche la variante TF-IDF tramite `ts_rank`).

Whoosh è un motore di ricerca interamente in Python che costruisce un indice invertito a partire da uno schema definito sui campi `title` e `content`. Le query vengono analizzate con un *MultifieldParser*, consentendo ricerche simultanee su più campi. Per il ranking offre due modalità: un modello TF-IDF classico e una versione weighted che applica boost personalizzati ai singoli campi (ad esempio titolo con peso doppio, contenuto con peso standard).

PyLucene integra Apache Lucene nella JVM attraverso binding Python, eseguendo la tokenizzazione, la rimozione delle stopwords e lo stemming con *ItalianAnalyzer*. Le ricerche full-text utilizzano il `QueryParser` sul campo `content` e possono essere corredate da un modulo di spell-checking basato su `SpellChecker` che utilizza i termini del campo `title`. Il ranking predefinito è basato su BM25, con la possibilità di passare al modello *ClassicSimilarity* per ottenere un comportamento TF-IDF.

A decorative graphic on the left side of the slide. It consists of a blue parallelogram and a light green parallelogram, both tilted at an angle. The blue shape is in the foreground, and the green shape is partially behind it. They are set against a dark blue background with diagonal stripes.

3. Modelli di Ranking



Modelli di ranking utilizzati

- TF-IDF (Term Frequency - Inverse Document Frequency): questo modello assegna un punteggio ai documenti in base alla frequenza di un termine all'interno del documento.
- BM25 (Best Matching 25): questo modello tiene conto della lunghezza del documento e applica una formula probabilistica per ottimizzare il ranking. E' ampiamente usato per migliorare la precisione nella selezione dei documenti più rilevanti.
- Variante **weighted** : questo modello si basa sull'idea di assegnare importanza diversa ai singoli campi del documento.

Questa combinazione di TF-IDF, BM25 e variante weighted ci ha permesso di mettere a confronto modelli a complessità crescente e di individuare quale approccio bilancia meglio precisione e recall nel dominio medico.



Modelli di ranking utilizzati

Nel corso del progetto abbiamo adottato tre diversi modelli di ranking per confrontare l'efficacia dei motori di ricerca:

Il primo è il classico TF-IDF, basato sulla frequenza dei termini nel documento normalizzata dall'inverso della frequenza nei documenti del corpus; è stato impiegato sia in Whoosh che come variante secondaria in PyLucene tramite il *ClassicSimilarity*.

Il secondo modello è BM25, utilizzato come impostazione predefinita in PostgreSQL (*ts_rank_cd*), in Whoosh e in PyLucene;

in Whoosh abbiamo sperimentato una versione “weighted” del TF-IDF, applicando un boost maggiore al campo title rispetto al campo content, per valutare l'impatto di un ranking che assegna maggiore importanza ai titoli degli articoli; questa medesima logica di peso differenziato è stata adottata anche in PostgreSQL tramite opportuni coefficienti nei campi tsvector.



Una piccola precisazione..

Per quanto concerne la variante weighted: si basa sull'idea di assegnare importanza diversa ai singoli campi del documento.

In Whoosh (e analogamente negli altri motori), al posto di trattare titolo e contenuto alla pari, si applica un **boost** maggiore al campo title (ad esempio peso = 2.0) rispetto al campo content (peso = 1.0).

In fase di scoring, le occorrenze nei titoli avranno quindi un impatto doppio rispetto a quelle nel testo, migliorando la rilevanza dei risultati che menzionano i termini chiave già nel titolo dell'articolo.

A decorative graphic on the left side of the slide. It consists of a blue parallelogram and a light green parallelogram, both tilted at an angle. The blue shape is in the foreground, and the green shape is partially behind it. They are set against a dark blue background with faint, lighter blue diagonal stripes.

4. Query del benchmark



Query del benchmark

Il benchmark include 10 query progettate per testare l'efficacia dei motori di ricerca.

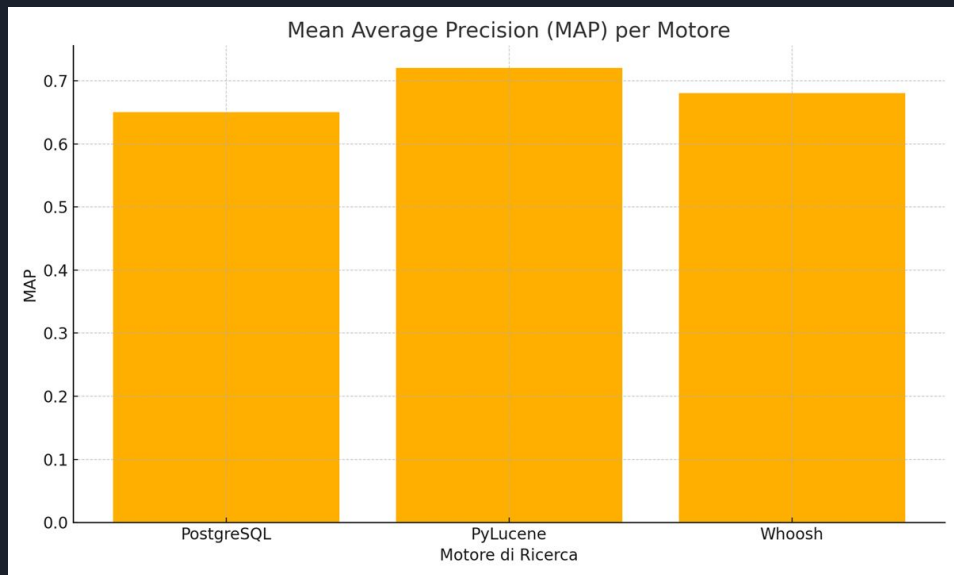
Ogni query è stata tradotta in linguaggio tecnico per ciascun motore.

1. Query generiche: Es: 'influenza', 'diabete', che cercano corrispondenze nei documenti.
2. Query basate su campo: Es: title:bronchite, per testare la ricerca in campi specifici.
3. Query complesse: Es: "Diagnosi precoce del tumore", combinando più termini per simulare scenari di ricerca avanzata.



5. Risultati

Benchmark e Metriche

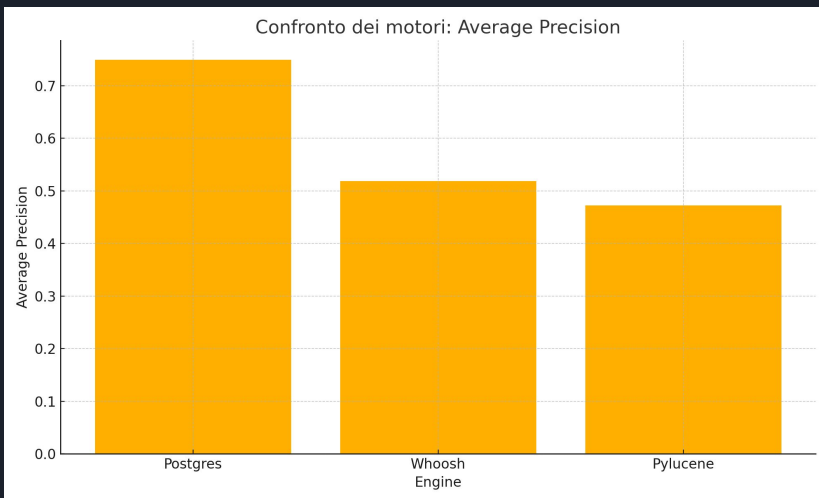


PyLucene presenta la MAP più alta con 0.72.

Whoosh e Postgres mostrano buoni risultati ma inferiori.

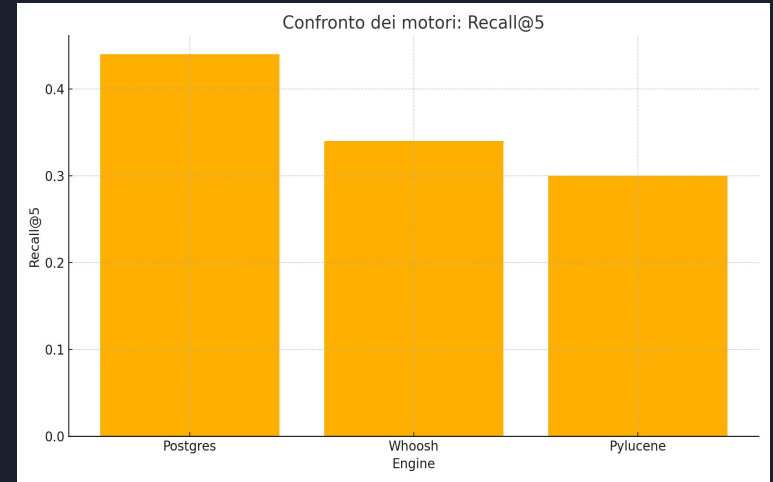
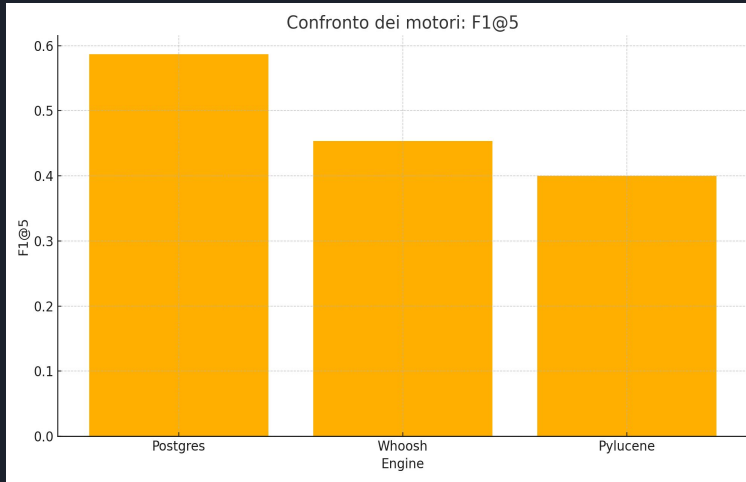
Possibilità di valutazione di un trade-off: complessità vs performance.

Benchmark e Metriche



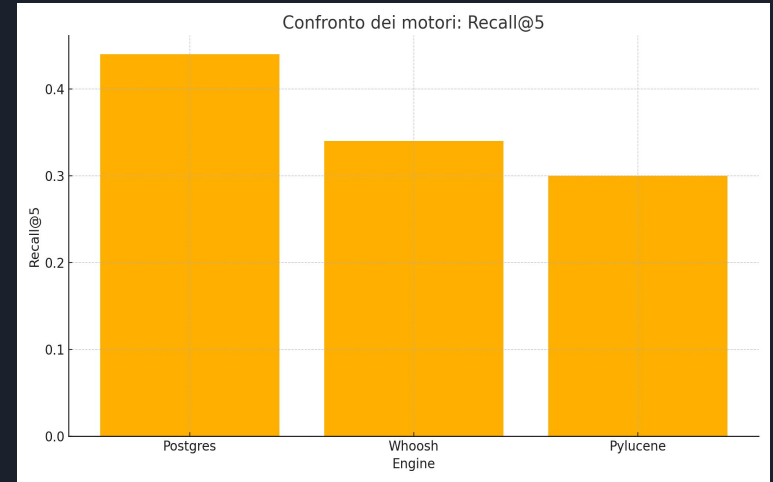
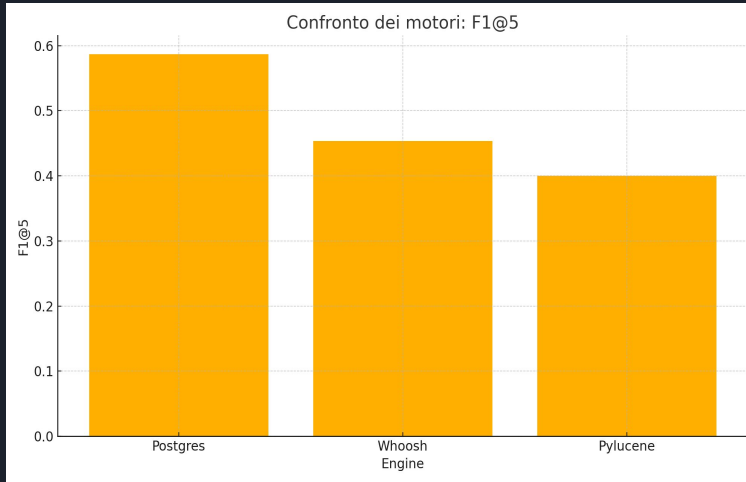
Il grafico AP mostra come PostgreSQL ottenga un valore medio di 0,75, seguito da Whoosh con 0,52 e PyLucene con 0,47, indicando una migliore capacità di mantenere risultati rilevanti su tutta la lista restituita.

Benchmark e Metriche



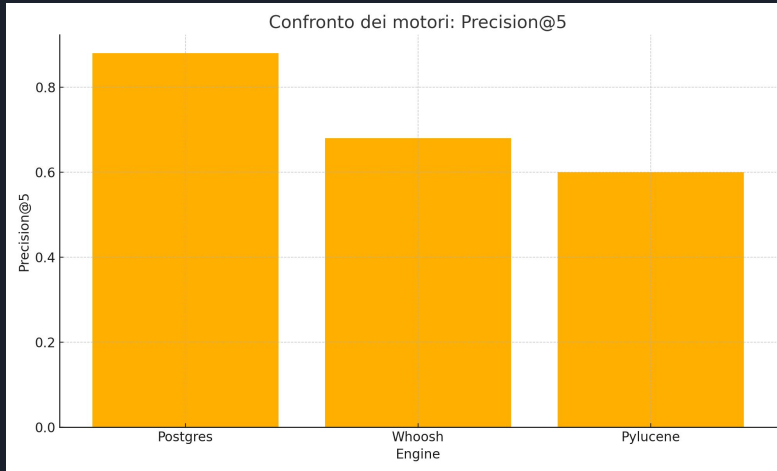
Anche se il Gold Standard non è ancora stato formalmente codificato, per questa analisi abbiamo scelto manualmente, per ciascuna delle 10 query mediche, un insieme di documenti di riferimento.

Benchmark e Metriche



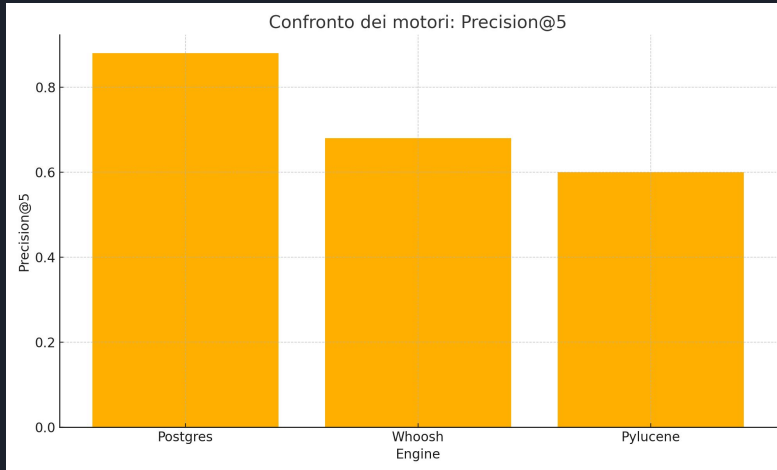
Per calcolare il Recall@5, abbiamo misurato la frazione di questi documenti di riferimento ritrovata all'interno dei primi cinque risultati restituiti dal motore. Il grafico mostra quindi come, in media, PostgreSQL recuperi il 44 % dei documenti rilevanti, Whoosh il 34 % e PyLucene il 30 %.

Benchmark e Metriche



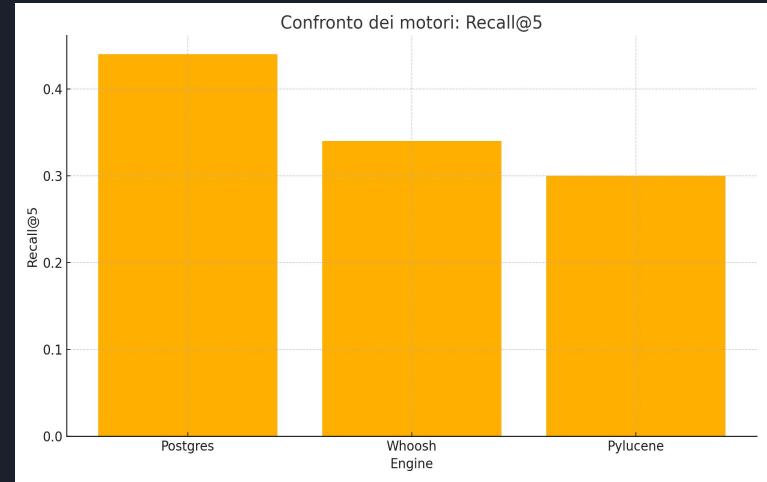
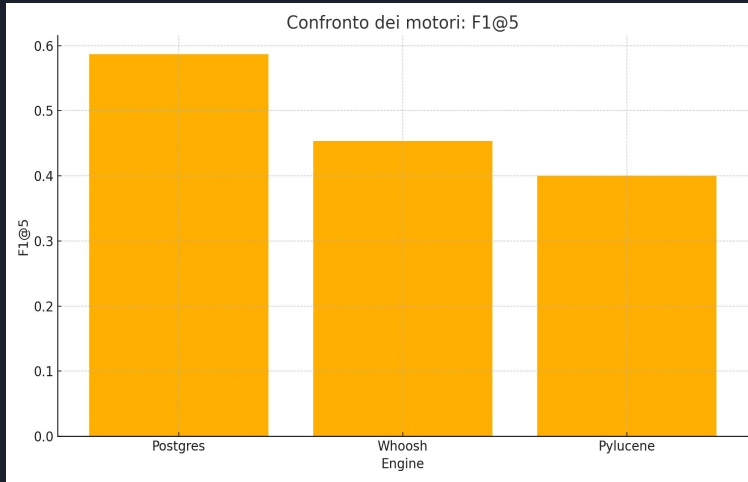
La Precision@5 misura la proporzione di documenti effettivamente rilevanti tra i primi cinque risultati restituiti da ciascun motore. Pur non avendo formalizzato un Gold Standard completo, abbiamo selezionato manualmente per ogni query un insieme di documenti chiave considerati pertinenti.

Benchmark e Metriche



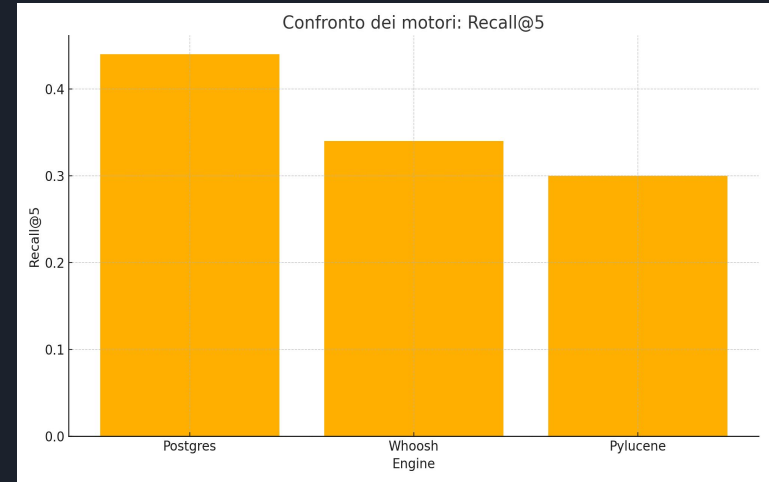
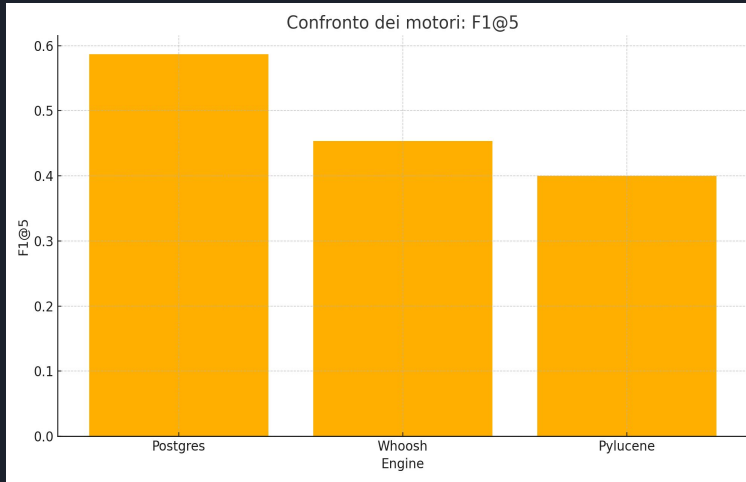
Il punteggio Precision@5 viene calcolato come media delle frazioni di risultati rilevanti ritrovati nel top-5 su tutte le 10 query. Nel nostro esperimento, PostgreSQL ottiene una Precision@5 di 0,88, segno di un'elevata capacità di fornire risultati pertinenti già nelle prime posizioni; Whoosh raggiunge 0,68 e PyLucene 0,60, indicando una minore accuratezza iniziale rispetto a PostgreSQL, ma comunque un livello accettabile di rilevanza per ricerche specialistiche in ambito medico.

Benchmark e Metriche




L'F1@5 combina Precision@5 e Recall@5 in un'unica misura armonica, ponendo eguale peso sull'accuratezza (Precision@5) e sulla completezza (Recall@5) nei primi cinque risultati.

Benchmark e Metriche



Anche qui ci siamo basati sullo stesso insieme di documenti di riferimento per ogni query: i punteggi medi risultanti sono 0.59 per PostgreSQL, 0.45 per Whoosh e 0.40 per PyLucene. Questo evidenzia che PostgreSQL, pur non raggiungendo altissime percentuali di recall, riesce a mantenere un bilanciamento ottimale tra precisione e completezza all'interno del top-5.



6. Conclusioni e sviluppi futuri



Sviluppi Futuri

1. Definizione del Gold Standard

creando liste di documenti rilevanti per ogni query basate su valutazioni di esperti medici. Questo permetterà di automatizzare il calcolo delle metriche e di rendere i risultati riproducibili.

In secondo luogo, si potrà integrare un **ranking semantico** con modelli di embedding specializzati in ambito biomedico (ad esempio BioBERT o SciBERT), in modo da catturare sinonimie e relazioni concettuali che vanno al-di-là della mera corrispondenza lessicale.

2. Tecnica del 'Did you mean..?'

Possibile implementazione ottenibile tramite spell-checker di Lucene e su dizionari medici, suggerendo automaticamente correzioni in caso di termini mal digitati o poco frequenti. Questi miglioramenti porteranno il sistema da un prototipo accademico a uno strumento concreto per la ricerca di informazioni cliniche.



Grazie dell'attenzione.