

Dipartimento di Matematica e Informatica

Corso di Laurea Triennale in Informatica

*Lorenzo Maria Alberto Paoria*

Titolo

Relazione Progetto Finale

Relatore: Emiliano Tramontana

Chiar.mo Prof. ---

Correlatori:

Gabriella Verga

Alessandro Midolo

Anno Accademico 2023 – 2024

# Indice

[Indice 3](#_Toc175728921)

[Introduzione 4](#_Toc175728922)

[Capitolo 1 - Metodologia 6](#_Toc175728923)

[1.1 - Titolo. 6](#_Toc175728924)

[1.1.1 - Titolo. 6](#_Toc175728925)

[1.1.2 - Titolo. 6](#_Toc175728926)

[1.2 - Titolo. 6](#_Toc175728927)

[Capitolo 2 - Risultati. 7](#_Toc175728928)

[2.1 - Post con codice. 7](#_Toc175728929)

[2.2 - Post senza codice. 8](#_Toc175728930)

[Capitolo 3 - Conclusioni 10](#_Toc175728931)

[Riferimenti 11](#_Toc175728932)

# Introduzione

Questa tesi si propone di confrontare le risposte fornite dagli utenti di StackOverflow con quelle generate da sistemi di intelligenza artificiale avanzati, come ChatGPT, al fine di valutare la capacità di tali tecnologie nel fornire soluzioni tecniche di alta qualità. L'obiettivo principale è analizzare la pertinenza, l'accuratezza e l'efficacia delle risposte prodotte dall'AI rispetto a quelle fornite da esperti umani, per comprendere i limiti e le potenzialità delle moderne tecnologie di linguaggio naturale nel supportare la risoluzione di problemi complessi nel campo dell'informatica.

Nel seguito, verranno illustrati i contenuti dei capitoli della tesi.

* Font per i Titoli dei capitoli: Cambria 16 (Stile Titolo 1).
* Font per i Titoli dei sottocapitoli: Cambria 14 (Stile Titolo 2),
* Si prosegue con Titolo 3, ecc…
* L’utilizzo degli Stili “Titolo” permette di generare l’Indice automaticamente.
* Ogni capitolo deve terminare con una Interruzione di pagina.
* Font per il testo semplice: Calibri 12 (Stile Normale).
* Interlinea 1,5.
* Margine Sinistro 4 cm. Destro, Superiore e Inferiore 3 cm.
* Spaziatura paragrafo 12 pt dopo.
* Figure: Salvo particolari motivazioni, si suggerisce di inserire le figure “In linea con il testo” e centrate. Inoltre, bisogna aggiungere la didascalia con l’opportuno strumento in modo da numerarle automaticamente e poterle citare nel testo tramite Riferimenti incrociati. Vedere esempio.
* Bibliografia: la bibliografia va inserita come ultima sezione, dopo le conclusioni. Tutte le fonti vanno inserite con lo strumento Gestione fonti e citati con lo strumento Riferimenti. La bibliografia potrà così essere generata automaticamente. Vedere esempio.
* Si possono usare note a piè di pagina.
* Si possono inserire ringraziamenti.
* Si può essere flessibili rispetto alle regole sopraccitate.

# Metodologia

Questo è il capitolo 1. Questa è una citazione [1]

## Titolo.

Questo è un sottocapitolo del capitolo 1

### Titolo.

Questo è un sotto-sotto capitolo del capitolo 1.

### Titolo.

Questo è un altro sotto sotto capitolo del capitolo 1

## Titolo.

Questo è un altro sottocapitolo del capitolo 1.

# Risultati.

Una volta analizzati tutti i post e completate le richieste all’IA, abbiamo prodotto i risultati per i diversi tipi di catalogazione, principalmente suddivisi in post con e senza codice.

I risultati sono stati illustrati attraverso grafici ed esempi per facilitarne l'interpretazione.

## Post con codice.

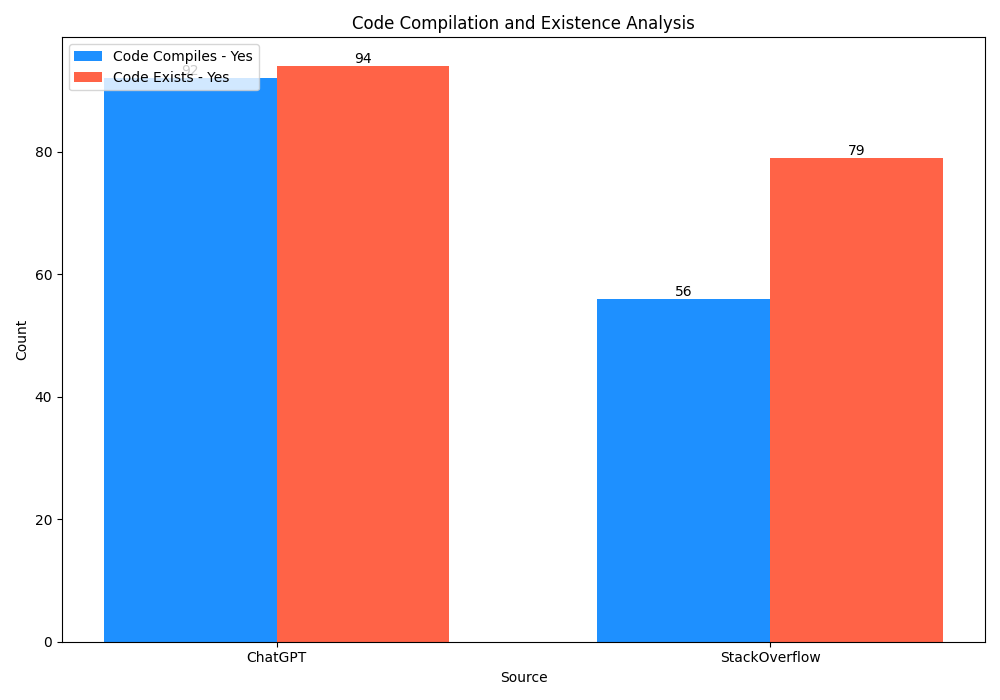
Per Per i post contenenti codice presenti nelle risposte, è stata valutata l'eseguibilità del codice o, nel caso di frammenti decontestualizzati, la correttezza semantica rispetto alla domanda. L'analisi è stata condotta su un dataset di 218 domande. Dai risultati è emerso che i post contenenti codice, già selezionati durante la creazione del dataset, erano:

* ChatGpt: 94 instanze
* StackOverflow: 79 instanze

Di queste:

* ChatGpt: 92 instanze compilano con successo
* StackOverflow: 56 instanze compilano con successo

Di seguito un grafico sulla situazione:



Per fornire un esempio testuale dei post analizzati e dei risultati ottenuti, presenteremo un esempio di un post contenente codice e la relativa analisi.

Domanda:

"ID":"937", "Question": "I know that they are implemented extremely unsafely in C/C++. Can't they be implemented in a safer way? Are the disadvantages of macros really bad enough to outweigh the massive power they provide? "

Per questa domanda, la milgiore risposta di StackOverflow risponde con del codice:

#define TWO 1+1

#define min(x, y) (((x) < (y)) ? (x) : (y))

Anche se questo codice è compilabile, può comportarsi in modo inaspettato quando viene utilizzato in espressioni. Pertanto, quando l'IA viene interpellata sulla sua correttezza semantica, lo considera non semanticamente corretto rispetto a quanto richiesto dalla domanda.

Per questa domanda, ChatGpt risponde usando del codice:

inline int square(int x) {

return x \* x;

}

ChatGPT sconsiglia l'uso delle macro in C++ e raccomanda invece di adottare un approccio basato su funzioni inline. Questo codice si compila correttamente ed è in linea con la domanda, poiché propone una soluzione alternativa ma più sicura rispetto a una che potrebbe comportare dei rischi.

## Post senza codice.

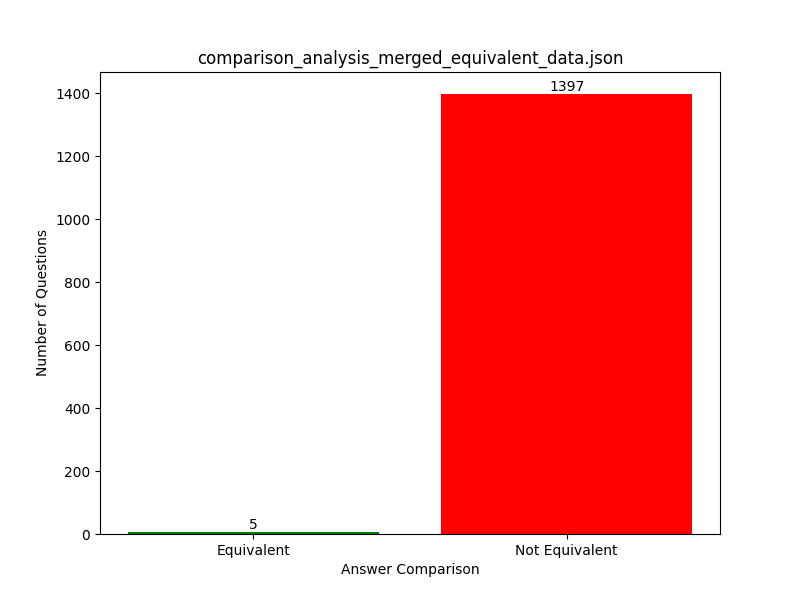
Per i post che non contengono codice, sono state raggruppate diverse categorie, tra cui:

* Post con un numero di caratteri superiore a limitChar
* Post con un numero di caratteri inferiore a limitChar
* Post contenenti un termine specifico

Sono state analizzate le equivalenze tra le migliori risposte di StackOverflow e le risposte di ChatGPT per tutti questi tipi di post. L'analisi è stata condotta su un dataset di 1402 domande e dai risultati è emerso che:

* 1397 domande non avevano risposte equivalenti
* 5 domande avevano risposte equivalenti

Di seguito un grafico della situazione:



Per illustrare il tipo di analisi effettuata e i risultati ottenuti, presenteremo un esempio che include una domanda con due risposte equivalenti e una domanda con due risposte non equivalenti.

Domanda con risposte equivalenti:

"ID":"21870", "Question": "Is is possible to correctly call yourself (or your team) \"Agile\" if you don't do TDD (Test-Driven Development)? "

La domanda chiede quando è appropiato utilizzare gli exception rispetto agli assert per la validazione degli input nelle funzioni e quale sia la prassi per migliore per gestire questi controlli.

Per questa domanda ChatGpt risponde:

"ChatGpt answer": "Yes, it is possible to call yourself or your team \"Agile\" even if you don't practice Test-Driven Development (TDD). Agile is a broader methodology that emphasizes principles and values outlined in the Agile Manifesto, such as collaboration, customer feedback, adaptive planning, and iterative development. TDD is a specific technique that some Agile teams choose to adopt, but it is not a requirement for being Agile.  Agile methodologies can incorporate various practices such as Scrum, Kanban, or Extreme Programming (XP), each of which may emphasize different techniques.

While TDD is beneficial for ensuring code quality and facilitating refactoring, Agile encompasses a range of practices and mindsets that can still promote responsiveness to change and customer-oriented development without necessarily using TDD.  Ultimately, what defines an Agile team is their adherence to Agile principles and their commitment to delivering value iteratively and collaboratively, rather than strictly following any one specific practice."

Per questa domanda la migliore risposta di StackOverflow risponde:

"Stack Overflow best answer": "Yes, yes, yes, a million times yes. Agile is a philosophy, TDD is a specific methodology. If I wanted to be really picky I could simply point out that there are quite a few variations of xDD - which their advocates will explain in depth are not TDD - but those are still substantially bound up with test first so that would be cheating. So lets says this - you can be agile without doing \"test first\" development (look at the way scrum works - nowhere in there are there specifics about how you write code). Look at a kanban board, look at all sorts of agile methodologies. Do you want unit tests? Of course you do, for all kinds of reasons - and you might well make an argument that you can't

be agile without unit tests (although I suspect that you can be) - but you don't have to write them first to be agile. And finally, its equally true that you could do Test First without being Agile and strong arguments for doing test first regardless of your overall dev philosophy.  It seems that others (with a more SOLID rep) have a similar opinion... Though it's not impossible to do   Agile without TDD and OOD, it is difficult.  Without TDD the iteration   rate of...  (The link in the tweet is to the full answer on LinkedIn) "

Entrambe le risposte concordano sul fatto che: essere Agile non richiede obbligatoriamente l'adozione di TDD, Agile si basa su principi generali di sviluppo iterativo e collaborazione mentre TDD è una tecnica specifica che può essere adottata ma non è essenziale.

Per fare un confronto di seguito troviamo una domanda con due risposte non equivalenti.

Risposta migliore di Stack Overflow:

Risposta migliore di ChatGpt:

Vediamo come queste due domande non sono equivalenti, quella di ChatGpt

# Conclusioni

# Riferimenti

|  |
| --- |
|  |