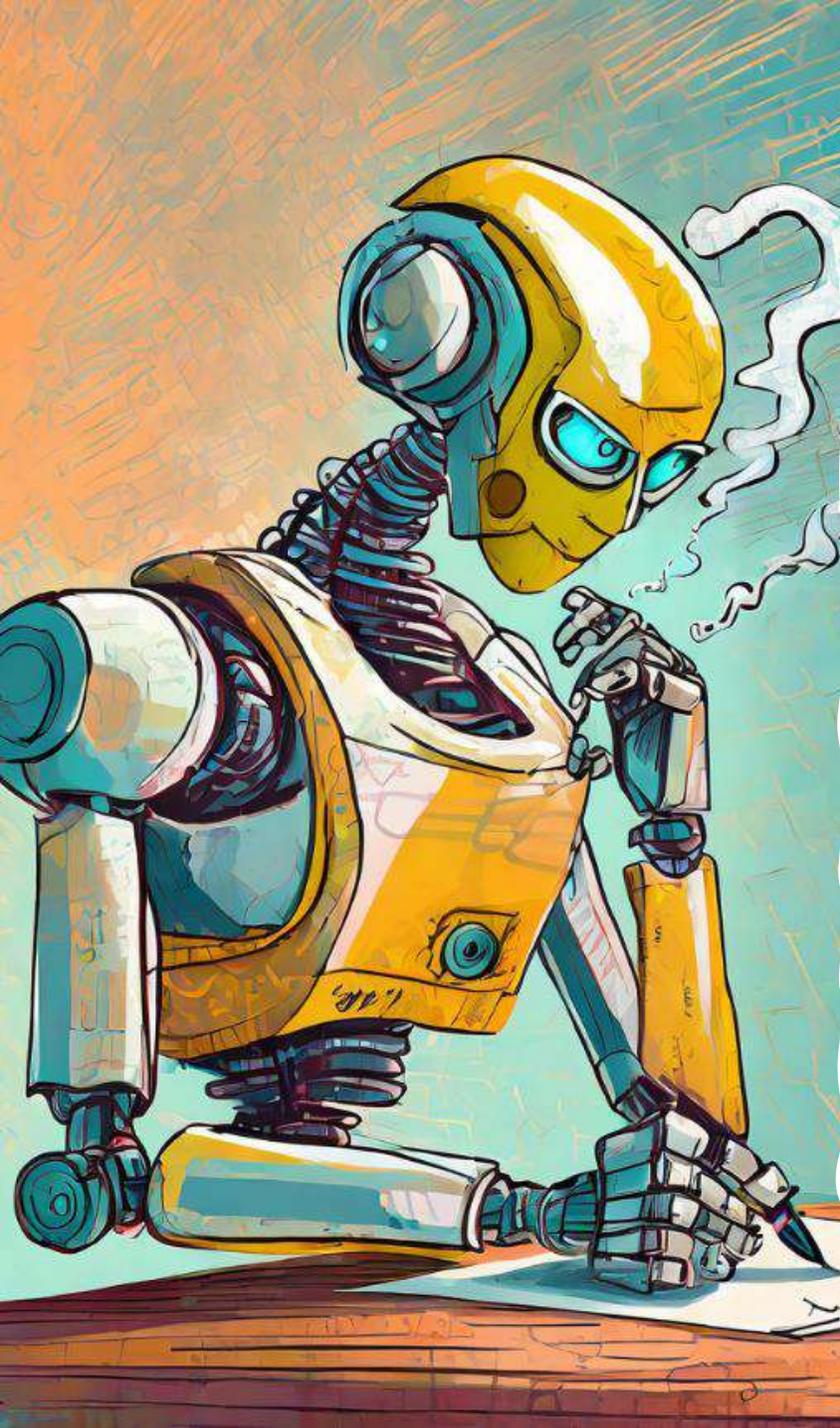


ChatGPT: lavorare con il testo

Scrittura di un breve intervento accademico dallo stile "insolito"



Obiettivo: Far produrre ad un LLM un intervento per una rivista accademica.

- La rivista nel caso specifico è *Indiscipline*, una rivista di scienze sociali.
- "Intervento"? Non un classico paper scientifico, ma un contributo di circa 1500 parole su temi specifici. Il macrotema individuato nel caso specifico era quello del ruolo pubblico della sociologia contemporanea.
- Non un classico paper poiché il contributo è discorsivo e breve, non diviso in sezioni con titoli. In pratica un flusso di testo dalla prospettiva di un sociologo accademico che si esprime brevemente, ma in modo "personale" sul tema.

Scelte di metodo:

- Livello base-intermedio: ovvero rivolto a chi ha una panoramica sul funzionamento di un LLM e delle principali opzioni (servizi esterni; utilizzo API di un servizio; hosting locale; ecc.), ma non ha mezzi e/o competenze e/o tempo per un utilizzo avanzato di un LLM.

Scelta: ChatGPT-4 (premium).

- Necessità di accedere a "fonti accademiche" al di là di quelle già possedute dal modello nella fase di addestramento e necessità di fornire direttamente articoli e documenti (**R**etrieval **A**ugmented **G**eneration).

Scelta: Consensus.



PROMPT ENGINEERING

- Siamo partiti da un prompt iniziale semplice, ma che fosse il più esaustivo possibile (a destra).
- Per poi provare template forniti da plugin ad-hoc di ChatGPT (e.g. costar).

★★★★ CO-STAR ★★★★★

Context
Clarify the scenario for relevance

Objective
Specify the goal to focus LLM's response

Style
Guide the style for tailored writing

Tone
Set the attitude for response

Audience
Identify the target recipients

Response
Define the required output format

Reset

Copy to Clipboard

Setting Your Context

- Identify your role or purpose for using the extension.
- *E.g. I am a teacher and I want to generate a teaching plan for my students. The teaching plan is to allow students to understand what is AI*

Stating Your Objective

- Summarize your goal in one/two sentence.
- *E.g. I want to generate a one week teaching plan.*

PROMPT:

- You are an academic researcher in the field of sociology, aiming to present your unique perspective. Write an academic article of about 12000 characters.

Following these rules:

- While integrating the provided context, maintain an academic and coherent style.
- Use your autonomous bibliography and the one I gave you in the context with my papers.
- Be precise and use appropriate citations and sources as necessary (maximum of 3).
- Endeavor to offer your vision on the topic, starting from the context, what you consider significant in the subject matter, and reflecting on the role of the intellectual.
- Avoid using section titles. Do not use title for sections.
- Describe a potential theory on the role of the intellectual in today's world.
- Maintain an academic tone, demonstrate a deep understanding of the topic, and offer original insights.

Risultati:

- Non c'è una formula magica per compiti (task) insoliti.

Alla fine siamo ripartiti da un prompt direttamente in italiano che sintetizzasse i precedenti:

"Sei un esperto sociologo accademico. ha studiato approfonditamente il ruolo pubblico della sociologia. Il tuo compito è scrivere un discorso rivolto ai tuoi colleghi accademici. nel tuo discorso devi argomentare in modo convincente quale ruolo pubblico la sociologia dovrebbe giocare rispetto al recente sviluppo dei sistemi di intelligenza artificiale. usa uno stile raffinato e accattivante, ma adatto all'esposizione orale. produci un testo di circa duemila parole usando tutti i materiali forniti nel contesto e la tua personale bibliografia. non commentare il contesto e non fornire riferimenti bibliografici. Usa il contesto solo per la produzione di un testo originale."

E siamo andati avanti selezionando i migliori contenuti generati e aggiungendo dettagli:

"riscrivi il testo sotto forma di articolo in 2000 parole. espandilo argomentando il ruolo dell'intellettuale nel quadro da te fornito."

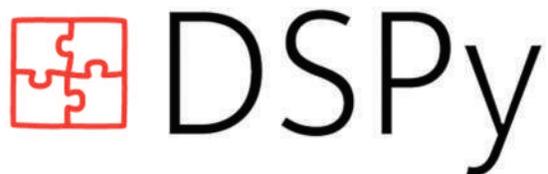
"espandi l'ultimo testo concentrandoti sul ruolo pubblico della sociologia, approfondendo il tema degli approcci anti-disciplinari in relazione all'intelligenza artificiale".

- I LLMs si confermano "pappagalli stocastici" in compiti che richiedono un forte punto di vista personale ed il loro output migliora con il migliorare dei prompts, dei dettagli forniti e del materiale fornito.

- Le tecniche di RAG stanno migliorando e le allucinazioni diminuendo: i LLMs si confermano ottimi strumenti per il *dialogo con* e la *manipolazione del testo*.



Ricerca sul tema attualmente in rapido movimento...



DSPy: *Programming*—not prompting—Foundation Models

- [Jun'24] [Optimizing Instructions and Demonstrations for Multi-Stage Language Model Programs](#)
- [Oct'23] [DSPy: Compiling Declarative Language Model Calls into Self-Improving Pipelines](#)
- [Jul'24] [Fine-Tuning and Prompt Optimization: Two Great Steps that Work Better Together](#)
- [Jun'24] [Prompts as Auto-Optimized Training Hyperparameters](#)
- [Feb'24] [Assisting in Writing Wikipedia-like Articles From Scratch with Large Language Models](#)
- [Jan'24] [In-Context Learning for Extreme Multi-Label Classification](#)
- [Dec'23] [DSPy Assertions: Computational Constraints for Self-Refining Language Model Pipelines](#)
- [Dec'22] [Demonstrate-Search-Predict: Composing Retrieval & Language Models for Knowledge-Intensive NLP](#)

DSPy: Compiling Declarative Language Model Calls into Self-Improving Pipelines

Omar Khattab, Arnab Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, Ashutosh Sharma, Thomas T. Joshi, Hanna Moazam, Heather Miller, Matei Zaharia, Christopher Potts

The ML community is rapidly exploring techniques for prompting language models (LMs) and for stacking them into pipelines that solve complex tasks. Unfortunately, existing LM pipelines are typically implemented using hard-coded "prompt templates", i.e. lengthy strings discovered via trial and error. Toward a more systematic approach for developing and optimizing LM pipelines, we introduce DSPy, a programming model that abstracts LM pipelines as text transformation graphs, i.e. imperative computational graphs where LMs are invoked through declarative modules. DSPy modules are parameterized, meaning they can learn (by creating and collecting demonstrations) how to apply compositions of prompting, finetuning, augmentation, and reasoning techniques. We design a compiler that will optimize any DSPy pipeline to maximize a given metric. We conduct two case studies, showing that succinct DSPy programs can express and optimize sophisticated LM pipelines that reason about math word problems, tackle multi-hop retrieval, answer complex questions, and control agent loops. Within minutes of compiling, a few lines of DSPy allow GPT-3.5 and llama2-13b-chat to self-bootstrap pipelines that outperform standard few-shot prompting (generally by over 25% and 65%, respectively) and pipelines with expert-created demonstrations (by up to 5-46% and 16-40%, respectively). On top of that, DSPy programs compiled to open and relatively small LMs like 770M-parameter T5 and llama2-13b-chat are competitive with approaches that rely on expert-written prompt chains for proprietary GPT-3.5. DSPy is available at [this https URL](#).

Subjects: [Computation and Language \(cs.CL\)](#); [Artificial Intelligence \(cs.AI\)](#); [Information Retrieval \(cs.IR\)](#); [Machine Learning \(cs.LG\)](#)
Cite as: [arXiv:2310.03714 \[cs.CL\]](#)
(or [arXiv:2310.03714v1 \[cs.CL\]](#) for this version)
<https://doi.org/10.48550/arXiv.2310.03714>

Submission history

From: Omar Khattab [[view email](#)]
[v1] Thu, 5 Oct 2023 17:37:25 UTC (77 KB)



TextGrad: Automatic "Differentiation" via Text

Mert Yuksekgonul, Federico Bianchi, Joseph Boen, Sheng Liu, Zhi Huang, Carlos Guestrin, James Zou

AI is undergoing a paradigm shift, with breakthroughs achieved by systems orchestrating multiple large language models (LLMs) and other complex components. As a result, developing principled and automated optimization methods for compound AI systems is one of the most important new challenges. Neural networks faced a similar challenge in its early days until backpropagation and automatic differentiation transformed the field by making optimization turn-key. Inspired by this, we introduce TextGrad, a powerful framework performing automatic "differentiation" via text. TextGrad backpropagates textual feedback provided by LLMs to improve individual components of a compound AI system. In our framework, LLMs provide rich, general, natural language suggestions to optimize variables in computation graphs, ranging from code snippets to molecular structures. TextGrad follows PyTorch's syntax and abstraction and is flexible and easy-to-use. It works out-of-the-box for a variety of tasks, where the users only provide the objective function without tuning components or prompts of the framework. We showcase TextGrad's effectiveness and generality across a diverse range of applications, from question answering and molecule optimization to radiotherapy treatment planning. Without modifying the framework, TextGrad improves the zero-shot accuracy of GPT-4o in Google-Proof Question Answering from 51% to 55%, yields 20% relative performance gain in optimizing LeetCode-Hard coding problem solutions, improves prompts for reasoning, designs new druglike small molecules with desirable in silico binding, and designs radiation oncology treatment plans with high specificity. TextGrad lays a foundation to accelerate the development of the next-generation of AI systems.

Comments: 41 pages, 6 figures
Subjects: [Computation and Language \(cs.CL\)](#); [Artificial Intelligence \(cs.AI\)](#); [Machine Learning \(cs.LG\)](#)
Cite as: [arXiv:2406.07496 \[cs.CL\]](#)
(or [arXiv:2406.07496v1 \[cs.CL\]](#) for this version)
<https://doi.org/10.48550/arXiv.2406.07496>

Submission history

From: Mert Yuksekgonul [[view email](#)]
[v1] Tue, 11 Jun 2024 17:32:21 UTC (766 KB)

Grazie dell'attenzione!