

# Aplikasi Berbasis Web untuk Identifikasi Distribusi Survival Menggunakan Model LLM

Andi Ardiansyah Nasir (222111890, 4SD3)

Dosen Pembimbing: Budi Yuniarto, S.S.T., M.Si

**Ringkasan—** Analisis survival adalah cabang statistik yang mempelajari waktu hingga terjadinya suatu peristiwa, seperti kesembuhan atau kegagalan. Identifikasi distribusi data yang tepat menjadi tantangan utama, terutama dalam model parametrik seperti Weibull dan eksponensial, karena kesalahan dapat menghasilkan analisis yang bias. Dengan kemampuan Large Language Model (LLM) seperti GPT-4o mini, pola data kompleks dapat dikenali tanpa asumsi distribusi yang kaku. Penelitian ini mengembangkan aplikasi berbasis web yang memanfaatkan prompt engineering dan fine-tuning untuk meningkatkan akurasi analisis distribusi survival. Aplikasi ini dirancang untuk mempermudah pengguna dalam melakukan analisis dan memberikan rekomendasi statistik lanjutan, menjadikannya solusi praktis dan inovatif untuk berbagai bidang seperti kesehatan, teknik, dan ilmu sosial.

**Kata Kunci—** LLM, GPT-4o Mini, Analisis Survival, Distribusi Data, Aplikasi Web

## I. LATAR BELAKANG

Analisis survival merupakan salah satu cabang statistik yang berfokus pada pemodelan dan analisis waktu hingga terjadinya suatu peristiwa. Secara umum, analisis survival merupakan kumpulan metode analisis longitudinal yang digunakan untuk menganalisis data dengan waktu sebagai variabel hasil. Contohnya adalah analisis waktu hingga kesembuhan pasien dengan penyakit jantung atau waktu hingga kegagalan pada komponen sebuah mesin [1,2]. Metode analisis survival ini umumnya digunakan pada bidang medis, biologi, pemasaran dan ilmu sosial [3,4,5]. Selain itu, analisis survival memiliki beberapa nama alternatif atau sinonim karena penggunaannya yang luas di berbagai bidang, seperti *event history analysis* dalam ilmu politik, *reliability theory* dalam bidang teknik, atau *duration analysis* dalam ekonomi [2].

Dalam analisis survival, metode yang digunakan dikategorikan menjadi tiga kelompok: parametrik, semi-parametrik, dan non-parametrik. Model parametrik memerlukan asumsi mengenai bentuk distribusi dari data tertentu, seperti distribusi eksponensial, distribusi weibull, atau distribusi gamma. Model nonparametrik tidak memerlukan asumsi distribusi, seperti model regresi *Cox Proportional Hazard* (Cox PH) yang mengasumsikan distribusinya bebas. Sementara model nonparametrik, seperti estimator *Kaplan-Meier*, bebas dari asumsi distribusi [2,6,7,8].

Salah satu tantangan terbesar dalam analisis survival adalah melakukan identifikasi terkait distribusi data. Identifikasi ini sangat penting untuk memilih model analisis yang tepat, terutama jika bekerja dengan model parametrik. Menurut Prajah dan Emen (2019), terdapat beberapa kesalahan yang sering terjadi dalam melaporkan analisis statistik, salah satunya yang paling sering adalah terkait distribusi data. Strategi dalam pemilihan model yang efektif dibutuhkan untuk memastikan

analisis yang akurat dan reliabel. Selain itu, data yang sering dihadapi mengandung tantangan seperti *censoring* atau distribusi kejadian yang tidak simetris. Pemilihan distribusi yang salah dapat mengakibatkan hasil yang bias dalam melakukan estimasi dan dapat mengakibatkan interpretasi yang salah. Oleh karena itu, kita memerlukan pendekatan yang lebih fleksibel dan canggih untuk mengatasi masalah ini [1,9,10].

Perkembangan teknologi *Artificial Intelligence* (AI), khususnya dalam bidang *Natural Language Processing* (NLP), telah menghadirkan solusi potensial melalui *Large Language Models* (LLM). Model seperti BERT [11], GPT-4 [12], PaLM [13], dan Llama [14]. Model-model seperti BERT [11], GPT-4 [12], PaLM [13], dan Llama [14] telah menunjukkan kemampuan yang luar biasa dalam memproses dan menganalisis data kompleks. Teknik seperti *prompt engineering* dan *fine-tuning* memungkinkan LLM untuk menandingi metode tradisional dalam menyelesaikan berbagai tugas dengan efisiensi tinggi [15]. Model LLM dapat mempelajari pola kompleks dalam data tanpa harus bergantung sepenuhnya pada asumsi distribusi yang kaku. Ini memberikan keuntungan signifikan dalam mengatasi ketidaksesuaian distribusi dan memperbaiki akurasi estimasi. LLM juga dapat menangani data yang tidak terstruktur atau tidak lengkap dengan lebih efektif, berkat kapasitas pemrosesan yang besar dan kemampuan generalisasi.

Salah satu pengembangan dari LLM adalah *Large Multimodal Model* (LMM), yang dapat menangani berbagai tugas secara bersamaan dengan memproses data dari berbagai format, seperti, teks, gambar, video, suara, dll [10,16]. Kemampuan ini membuat siapa saja memiliki peluang untuk memanfaatkan LLM dalam domain analisis data yang memiliki kompleksitas yang tinggi, termasuk dalam analisis *survival*, di mana distribusi data yang sulit untuk diidentifikasi atau diinterpretasikan dapat diatasi menggunakan LMM tersebut.

GPT-4o mini merupakan model LLM yang dirancang untuk tugas-tugas dengan biaya rendah dan efisiensi tinggi. Model ini memiliki keunggulan multimodal, mampu menerima input berupa teks dan gambar serta menghasilkan output dalam bentuk teks. Dengan kecerdasan yang lebih tinggi dibandingkan GPT-3.5-turbo, GPT-4o mini sangat cocok untuk penyesuaian model melalui teknik seperti *prompt engineering* dan *fine-tuning*. Kemampuannya yang fleksibel dalam mengenali pola kompleks membuat model ini efektif untuk tugas identifikasi distribusi data survival, terutama pada data dengan *censoring* atau distribusi yang asimetris.

Berdasarkan kebutuhan tersebut, dengan menggabungkan metode-metode statistik klasik dan menggunakan kekuatan dari LLM seperti GPT-4o-mini, penelitian ini bertujuan untuk mengembangkan model yang dapat mengidentifikasi distribusi data dalam survival analysis dengan lebih akurat menggunakan

fine-tuning. Melalui fine-tuning, GPT-4o mini dapat dioptimalkan untuk mengenali atau mengidentifikasi pola dari distribusi analisis survival dengan lebih akurat, terutama pada data yang memiliki distribusi yang kompleks seperti *censoring*. Selanjutnya, Model yang nantinya dihasilkan akan diintegrasikan ke dalam sebuah aplikasi berbasis web untuk memudahkan pengguna dalam menganalisis dan menginterpretasikan data survival dengan lebih baik. Pengembangan aplikasi berbasis web ini diharapkan dapat menjadi alat yang inovatif dan praktis bagi peneliti, praktisi, dan akademisi dalam melakukan analisis survival secara lebih efektif dan efisien. Dengan antarmuka yang intuitif serta kemampuan untuk memanfaatkan model LLM, pengguna dapat mengatasi kompleksitas data survival dengan lebih mudah, sehingga mendukung pengambilan keputusan yang lebih baik dalam berbagai bidang, termasuk kesehatan, teknik, dan ilmu sosial.

II. RUMUSAN MASALAH

Berdasarkan latar belakang, berikut rumusan masalah yang akan akan dibahas dalam penelitian ini:

1. Bagaimana mengembangkan model LLM yang dapat mengidentifikasi distribusi data survival secara akurat?
2. Seberapa efektif performa model LLM dalam mengidentifikasi distribusi data survival?
3. Bagaimana cara mengintegrasikan model LLM ke dalam aplikasi berbasis web yang user-friendly untuk analisis survival?
4. Bagaimana model dapat memberikan rekomendasi analisis statistik lanjutan berdasarkan distribusi yang teridentifikasi?

III. TUJUAN PENELITIAN

Berdasarkan latar belakang dan rumusan masalah pada uraian sebelumnya, dirumuskan tujuan penelitian sebagai berikut:

1. Mengembangkan model LLM untuk mengidentifikasi distribusi data survival.
2. Menguji performa model LLM dalam mengenali jenis distribusi data survival.
3. Memberikan rekomendasi statistik lanjutan berdasarkan hasil analisis.
4. Membuat aplikasi berbasis web dan mengintegrasikan dengan LLM untuk memberikan rekomendasi statistik lanjutan.

IV. BATASAN PENELITIAN

Pada penelitian ini permasalahan yang ada akan dibatasi berdasarkan keterbatasan peneliti. Pembatasan masalah meliputi:

1. Bentuk distrisbusi data yang akan dicakup meliputi distribusi Weibull, exponential, log-normal, log-logistic, gamma, gompertz dan pareto.

2. Model yang digunakan terbatas pada GPT-4o mini dengan ketergantungan koneksi internet dan API

V. PENELITIAN TERKAIT

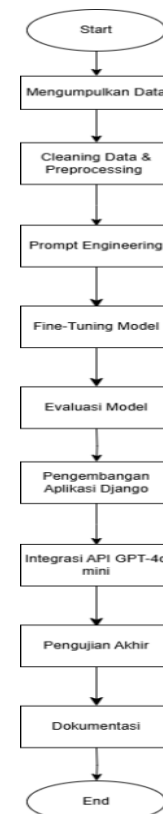
TABEL I  
TABEL LITERATUR PENELITIAN TERKAIT

No	Judul	Penulis, Publikasi	Tertulis	Hasil
1	Parametric Distributions for Survival and Reliability Analyses: A Review and Historical Sketch	Taketomi, Yamamoto, Chesneau, dan Emura, <i>Mathematics</i> , 2022	Tinjauan komprehensif tentang distribusi parametrik yang digunakan dalam analisis survival dan keandalan, termasuk sejarah, properti statistik, dan penerapan pada data nyata	Distribusi parametrik seperti eksponensial, Weibull, lognormal, dan gamma sangat penting dalam memodelkan waktu kejadian. Makalah ini menekankan perlunya pemilihan distribusi yang tepat untuk hasil analisis yang akurat, terutama saat data survival memiliki censoring atau distribusi tidak simetris. Pembahasan juga mencakup distribusi baru untuk memodelkan fenomena yang lebih kompleks.
2.	Review of Language Models for Survival Analysis	Vincent Jeanselme, Nikita Agarwal, Chen Wang, <i>AAAI Conference</i> , 2024	Ulasan mendalam tentang penggunaan LLM dalam analisis survival, termasuk strategi penerapan seperti fine-tuning, prompting, dan embedding.	LLM menawarkan potensi besar untuk meningkatkan efisiensi analisis data survival. Strategi fine-tuning menghasilkan kinerja superior, tetapi membutuhkan data pelatihan besar. Penggunaan embedding membantu model memahami struktur data yang rumit, sementara prompting memberikan fleksibilitas. Tantangan seperti distribusi yang sangat tidak simetris tetap menjadi perhatian.
3.	From Large Language Models to Large Multimodal Models: A Literature Review	Huang, Yan, Li, dan Peng, <i>Applied Sciences</i> , 2024	Tinjauan evolusi dari LLM ke LMM, termasuk arsitektur, strategi pelatihan, dan tantangan global.	LMM memperluas kapasitas LLM untuk tugas lintas modal (teks, gambar, audio, video). Strategi fine-tuning, seperti LoRA, meningkatkan efisiensi. Prompt engineering memperkuat

				kemampuan model dalam tugas kompleks. Namun, tantangan integrasi modal dan sumber daya masih signifikan.
4.	Prompt Engineering or Fine-Tuning? A Case Study on Phishing Detection with Large Language Models	Fouad Trad dan Ali Chehab, Machine Learning & Knowledge Extraction, 2024	Studi kasus membandingkan keefektifan prompt engineering dan fine-tuning untuk deteksi phishing URL menggunakan LLM.	Penelitian menunjukkan bahwa fine-tuning memberikan hasil yang lebih baik (F1-score hingga 97,29%) dibandingkan dengan prompt engineering. Prompt engineering memungkinkan pengembangan aplikasi yang cepat, tetapi terbatas dalam kinerja. Fine-tuning, meskipun lebih intensif sumber daya, mengungguli pendekatan state-of-the-art dalam deteksi URL phishing, menggarisbawahi pentingnya penyesuaian model untuk tugas spesifik
5.	Pembangunan Aplikasi Identifikasi Distribusi Data Survival Dengan Algoritma Genetika	Wulandari Fitri	Penelitian ini bertujuan untuk mengidentifikasi distribusi survival dengan menggunakan pendekatan analitis dan grafik. Penelitian ini juga menggunakan Algoritma Genetika untuk mendapatkan estimasi parameter untuk menghitung AIC	Hasil yang diperoleh nilai AIC yang dihasilkan dengan menggunakan Algoritma Genetika memberikan hasil yang tidak beda secara signifikan dengan menggunakan metode yang sering digunakan yaitu <i>Newton Raphson</i> .

## VI. METODE PENELITIAN

Gambar 1 menunjukkan rancangan metode penelitian yang mencakup berbagai tahapan penting dalam pengembangan aplikasi berbasis Large Language Model (LLM).



Gambar 1. Rancangan Penelitian

### A. Data dan Sumber Data

Pengumpulan data akan dilakukan dengan menggunakan dua jenis dataset, yaitu data sintetis dan data nyata. Dataset sintetis akan dibuat menggunakan teknik simulasi berbagai skenario. Data ini akan mencakup data survival dengan berbagai distribusi parametrik, seperti eksponensial, Weibull, dan gamma, dan akan dibuat menggunakan bahasa pemrograman Python. Sementara itu, dataset nyata akan dikumpulkan dari berbagai sumber, seperti data klinis atau data mengenai ketahanan suatu barang hingga mengalami kerusakan. Selain data teks, kami juga akan mengumpulkan gambar yang relevan untuk melatih dan menguji model. Gambar-gambar ini akan mencakup data sintetis yang dibuat dengan Python serta data yang dikumpulkan dari pencarian gambar di Google untuk distribusi survival.

### B. Pengembangan Model

#### 1. Prompt Engineering



Gambar 2. Fine-tuning model

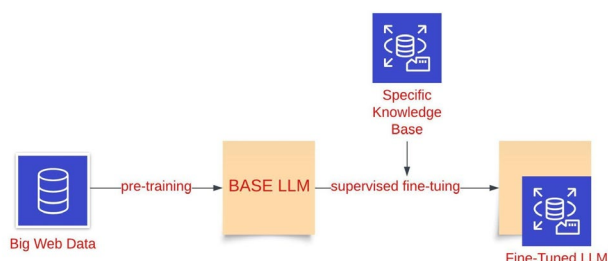
*Prompt engineering* merupakan langkah awal dalam melakukan pengembangan model LLM

yang berperan dalam memastikan model memberikan respons sesuai dengan kebutuhan pengguna. *Prompt engineering* melibatkan instruksi khusus agar model dapat mengeluarkan output yang sesuai dengan tugas tertentu tanpa harus mengubah parameter model. Pada tahap ini, kita akan membuat prompt sedemikian rupa agar model dapat secara efektif memahami dan memproses pertanyaan yang terkait dengan distribusi data *survival*. Salah satu teknik *prompt engineering* adalah *few-shot learning*. Teknik ini memanfaatkan kemampuan model untuk menggeneralisasi dari contoh yang sangat sedikit, sehingga membuat model lebih adaptif meskipun hanya menggunakan sejumlah kecil data pelatihan. Dengan menggunakan *few-shot learning* kita dapat membuat model lebih adaptif tanpa memerlukan banyak data pelatihan, dan menghemat sumber daya dan waktu.

Gambar 2. merujuk pada proses fine-tuning model, di mana model dioptimalkan menggunakan data spesifik dan prompt yang dirancang, memadukan pengetahuan eksternal untuk menghasilkan keluaran yang lebih akurat.

## 2. Fine Tuning

Setelah *prompt engineering*, tahap berikutnya adalah fine-tuning model GPT-4o mini. Proses ini melibatkan pelatihan model menggunakan dataset *survival* yang lebih spesifik, baik dari data sintesis maupun nyata. Tujuan fine-tuning adalah untuk meningkatkan akurasi dan kemampuan model dalam mengidentifikasi pola distribusi yang kompleks, termasuk menangani data *censored* dan distribusi yang asimetris.



Gambar 3. Fine-tuning model

## C. Pengujian Model

Dalam proses *prompt engineering* dan fine-tuning, kinerja model menjadi sangat penting, terutama karena tujuannya adalah melakukan klasifikasi distribusi data *survival*. Oleh karena itu, penelitian ini menggunakan beberapa metrik klasifikasi:

1. Akurasi: Akurasi adalah metrik yang paling umum digunakan dan dihitung sebagai rasio prediksi yang benar terhadap keseluruhan observasi. Meskipun metrik ini sangat berguna pada dataset yang seimbang, penggunaannya

terbatas dalam kondisi di mana terdapat ketidakseimbangan kelas, karena akurasi dapat memberikan gambaran yang menyesatkan.

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN}$$

2. Presisi: Dikenal juga sebagai positive predictive value, presisi dihitung sebagai rasio antara jumlah prediksi positif yang benar terhadap seluruh prediksi positif. Tingkat presisi yang tinggi penting dalam mengidentifikasi distribusi data, terutama untuk mengurangi kesalahan dalam mengklasifikasikan distribusi tertentu secara keliru.

$$Presisi = \frac{TP}{TP + FP}$$

3. Recall: Juga disebut sensitivity, recall dihitung sebagai rasio antara prediksi positif yang benar dengan semua observasi positif yang sebenarnya. Dalam konteks identifikasi distribusi, recall sangat penting untuk memastikan sebanyak mungkin distribusi yang benar dapat dikenali oleh model, mengurangi potensi terlewatnya distribusi tertentu.

$$Recall = \frac{TP}{TP + FN}$$

4. F1-score: F1-score adalah rata-rata harmonis dari presisi dan recall. Metrik ini lebih handal dibandingkan akurasi, terutama pada dataset yang tidak seimbang, karena mempertimbangkan baik positif palsu maupun negatif palsu. F1-score cocok untuk situasi di mana presisi dan recall sama-sama penting dalam proses identifikasi distribusi.

$$F1 - score = 2 \times \frac{Presisi \times Recall}{Presisi + Recall}$$

5. Area Under the Curve (AUC): Metrik ini mengukur kemampuan model untuk membedakan antara berbagai kelas distribusi dan diringkas dalam bentuk kurva Receiver Operating Characteristic (ROC). AUC memberikan gambaran umum tentang kemampuan model dalam membedakan distribusi yang berbeda.
6. True Positive Rate at a Given False Positive Rate: Metrik ini mengukur kemampuan model untuk secara akurat mengidentifikasi distribusi yang benar pada tingkat false positive rate tertentu. Metrik ini sangat bermanfaat dalam konteks identifikasi distribusi, terutama ketika kita perlu menjaga tingkat kesalahan positif palsu tetap rendah.
7. Akaike Information Criterion (AIC): adalah metrik untuk memilih model terbaik dengan menyeimbangkan kesesuaian data dan kompleksitas model. Dalam identifikasi distribusi, AIC membantu membandingkan beberapa model distribusi, dengan nilai AIC lebih rendah menunjukkan model yang lebih baik.

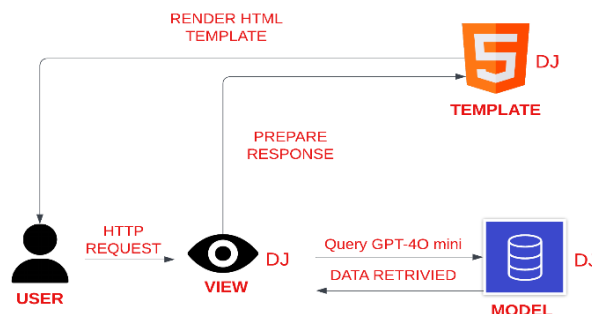
#### D. Pengemangan Aplikasi WEB

Dalam pengembangan aplikasi web ini, framework Django digunakan untuk membangun aplikasi berbasis arsitektur Model-View-Template (MVT). Arsitektur MVT memungkinkan pemisahan yang jelas antara logika bisnis, data, dan antarmuka pengguna, sehingga memudahkan pengelolaan serta pemeliharaan aplikasi. Berikut ini adalah komponen dari aplikasi yang dibangun:

**Model:** Bagian Model di Django akan mengelola interaksi dengan basis data yang menyimpan input pengguna, hasil analisis, dan prediksi distribusi data survival dari GPT-4o mini. Struktur basis data ini mencakup tabel untuk menyimpan tipe data, parameter distribusi yang diidentifikasi, serta metrik hasil analisis.

**View:** Bagian View bertindak sebagai penghubung antara logika aplikasi dan antarmuka pengguna. Pada tahap ini, View akan menerima data dari pengguna, mengirimnya ke GPT-4o mini melalui API, dan menerima hasil dari model untuk ditampilkan kepada pengguna.

**Template:** Template menyediakan antarmuka pengguna untuk memasukkan data dan menampilkan hasil analisis. Desain antarmuka ini dioptimalkan agar mudah dipahami oleh pengguna non-teknis.



Gambar 4. Architecture Model View Template (MVT)

#### VII. RANCANGAN JADWAL PENELITIAN

Tabel 2. merupakan tabel yang menunjukkan rancangan sementara jadwal penelitian. Rancangan jadwal tersebut dapat berubah di kemudian hari.

Kegiatan	2024			2025							
	Oktober	November	Desember	Januari	Februari	Maret	April	Mei	Juni	Juli	Agustus
Studi literatur											
Penyusunan proposal											
BAB I											
BAB II											
BAB III											
BAB IV & BAB V											
Makalah Seminar dan Pembuatan Aplikasi											
Seminar skripsi											
Revisi makalah pasca seminar											
Ujian skripsi											
Pengumpulan makalah & buku sidang											

#### DAFTAR PUSTAKA

- [1] Nanami Taketomi, K. Yamamoto, Christophe Chesneau, and Takeshi Emura, "Parametric Distributions for Survival and Reliability Analyses, a Review and Historical Sketch," Mathematics, vol. 10, no. 20, pp. 3907–3907, Oct. 2022, doi: <https://doi.org/10.3390/math10203907>.
- [2] F. Emmert-Streib and M. Dehmer, "Introduction to Survival Analysis in Practice," Machine Learning and Knowledge Extraction, vol. 1, no. 3, pp. 1013–1038, Sep. 2019, doi: <https://doi.org/10.3390/make1030058>.
- [3] C.-Y. Bae, B.-S. Kim, S.-H. Jee, J.-H. Lee, and N.-D. Nguyen, "A Study on Survival Analysis Methods Using Neural Network



- to Prevent Cancers,” *Cancers*, vol. 15, no. 19, pp. 4757–4757, Sep. 2023, doi: <https://doi.org/10.3390/cancers15194757>.
- [4] A. Hazra and N. Gogtay, “Biostatistics Series Module 9: Survival Analysis,” *Indian Journal of Dermatology*, vol. 62, no. 3, pp. 251–257, May 2017, doi: [https://doi.org/10.4103/ijd.IJD\\_201\\_17](https://doi.org/10.4103/ijd.IJD_201_17).
- [5] L. Mamudu and C. P. Tsokos, “Parametric and Non-Parametric Analysis of the Survival Times of Patients with Multiple Myeloma Cancer,” *Open Journal of Applied Sciences*, vol. 10, no. 04, pp. 118–134, 2020, doi: <https://doi.org/10.4236/ojapps.2020.104010>.
- [6] G. Rodríguez, “Parametric Survival Models 1 Survival Distributions 1.1 Notation,” 2001. Accessed: Nov. 05, 2024. [Online]. Available: <https://grodri.github.io/survival/ParametricSurvival.pdf>.
- [7] J. Gardiner, “Survival Analysis: Overview of Parametric, Nonparametric and Semiparametric approaches and New Developments,” 2010. Accessed: Nov. 05, 2024. [Online]. Available: <https://support.sas.com/resources/papers/proceedings10/252-2010.pdf>.
- [8] S. K. Praharaj and S. Ameen, “Common errors in reporting of statistical analyses,” *Kerala Journal of Psychiatry*, vol. 31, no. 1, Apr. 2019, doi: <https://doi.org/10.30834/kjp.31.1.2019.153>.
- [9] P. Wang, Y. Li, and C. K. Reddy, “Machine Learning for Survival Analysis: A Survey,” *arXiv.org*, Aug. 15, 2017. <https://arxiv.org/abs/1708.04649>
- [10] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” *arXiv.org*, May 24, 2019. <https://arxiv.org/abs/1810.04805#>
- [11] OpenAI, “GPT-4 Technical Report,” *arXiv (Cornell University)*, Mar. 2023, doi: <https://doi.org/10.48550/arxiv.2303.08774>
- [12] Aakanksha Chowdhery et al., “PaLM: Scaling Language Modeling with Pathways,” *arXiv (Cornell University)*, Apr. 2022, doi: <https://doi.org/10.48550/arxiv.2204.02311>.
- [13] H. Touvron et al., “Llama 2: Open Foundation and Fine-Tuned Chat Models,” *arXiv.org*, Jul. 19, 2023. <https://arxiv.org/abs/2307.09288>
- [14] S. Jaradat, R. Nayak, A. Paz, H. I. Ashqar, and M. Elhenawy, “Multitask Learning for Crash Analysis: A Fine-Tuned LLM Framework Using Twitter Data,” *Smart Cities*, vol. 7, no. 5, pp. 2422–2465, Sep. 2024, doi: <https://doi.org/10.3390/smartcities7050095>.
- [15] D. Zhang et al., “MM-LLMs: Recent Advances in MultiModal Large Language Models,” *arXiv (Cornell University)*, Jan. 2024, doi: <https://doi.org/10.48550/arxiv.2401.13601>.
- [16] OpenAI, “OpenAI API,” *platform.openai.com*. <https://platform.openai.com/docs/guides/fine-tuning>
- [17] F. Trad and A. Chehab, “Prompt Engineering or Fine-Tuning? A Case Study on Phishing Detection with Large Language Models,” *Machine learning and knowledge extraction*, vol. 6, no. 1, pp. 367–384, Feb. 2024, doi: <https://doi.org/10.3390/make6010018>.
- [18] “Do LLMs Play Dice? Exploring Probability Distribution Sampling in Large Language Models for Behavioral Simulation,” *Arxiv.org*, 2023. <https://arxiv.org/html/2404.09043v1> (accessed Nov. 05, 2024).
- [19] Django, “Django documentation | Django documentation,” Django Project, 2024. <https://docs.djangoproject.com/en/5.1/>