Peter Danenberg

+1 650 889 0679   ∘   danenberg@post.harvard.edu   ∘   in   github

Experience

Senior Software Engineer Google DeepMind (2024–)

Moved to DeepMind to lead rapid prototyping for Gemini, where I work with leadership to create feature prototypes within 24 hours based on insights from external biweekly meetups. Some prototypes gained visibility with top executives, including Demis Hassabis and Sundar Pichai; and advanced to production as Gemini Canvas and Gemini Agent.

I am responsible for full-stack development (front-end, back-end, modeling, and deployment), including:

- Built computer-use capabilities for autonomous agents (Gemini controlling browsers, booking flights, managing tasks) which became Gemini Agent: "order those Lululemon yoga pants I tried on last week—you remember which ones and my size."
- Built end-to-end app for rendering self-contained HTML, PDF, and SVG artifacts which became Gemini Canvas: "create a playable 3D voxel world with procedural terrain generation, block-building mechanics, and real-time physics."
- Co-created Little PAIPer (lilpaiper.ai) with Peter Norvig: a Socratic dialogue-based learning platform for AI education, replacing linear textbook reading with interactive, personalized conversations guided by Gemini.
- Organize biweekly Gemini workshops (geminimeetup.ai) with ∼ 350 attendees; created 18+ open-source demonstrations including AI gaming agents, voice companions, and fine-tuning pipelines (github.com/google-gemini/workshops).
- Built game and entertainment experiences for AR/XR through Project Astra, integrating prototype glasses with live vision and conversational AI.

Senior Software Engineer Google, Inc. (2021–2024)

Worked on autonomous agents in Bard; launched Bard Extensions to 1.8 million DAU using a self-orchestrating LLM. Wrote C++ backend serving neural-semantic models for Google Assistant, handling ∼ 120 million queries per day. Automated training and evaluation for 54 models in 18 languages.

Software Engineer Google, Inc. (2013–2021)

Developed a pipeline for Google Assistant's machine-translation models, launching models for multiple languages. Built Data QnR, a natural language interface to Google Analytics, supporting custom knowledge-graphs and metrics.

Intern Google, Inc. (Summer 2008)

Created Roxygen, a documentation system for R, now widely used.

Research Assistant Information Sciences Institute (2005–2008)

Developed multi-agent simulation environments for computational social science.

Talks and Publications

Danenberg, P. (2025). Towards a Positive Ethics of AI. *Oxford Institute for Ethics in AI.*

Danenberg, P. (2025). The Future of AI. *Magdalen College, Oxford University.*

Danenberg, P. (2025). Competence in the Age of LLMs. *TEDxCSTU.*

Danenberg, P. (2024). What is AI? *Westminster Technology Forum, House of Lords.*

Danenberg, P. (2024). Learnings on Building Gemini. *Imagination in Action, MIT; Forbes.*

Danenberg, P. (2023). Prompt-Engineering with Bard. *Devfest, Google.*

Danenberg, P. (2023). *Single Parameter Driven Tunable Resource Allocation for Machine Learning Tasks* (tech. rep. No. 5812). Technical Disclosure Commons, Defensive Publication Series.

Kahn, J., Danenberg, P., & et al. (2010). Model simulation and decision analysis with the SimR package in R. *The R User Conference.*

Danenberg, P., & Marsella, S. (2010). Data-Driven Coherence Models. *19th Annual Conference on Behavior Representation in Modeling and Simulation 2010, BRiMS 2010.*

Education

2004–2008	University of Southern California	Ph.D. candidate, Computer Science (Research Assistant)
2002–2004	Albert-Ludwigs-Universität, Freiburg	Ph.D. candidate, Philosophy
2000–2002	Harvard University	Ph.D. candidate, Comparative Literature
1995–2000	University of Southern California	B.A., Computer Science, Classics, Philosophy, German, Piano

Studied abroad at Cambridge University, Διεθνές Κέντρο Ελληνικών (Athens), Alliance Française (Paris), and Humboldt-Universität (Berlin).

Skills

Skill	Experience	Skill	Experience
C++	9 yrs.	NLU Infrastructure	6 yrs.
Python	8 yrs.	Java	5 yrs.
Full-Stack Development	7 yrs.	AI Prototyping	2 yrs.
TypeScript	6 yrs.	LLMs	2 yrs.