School Address:
110 Sproul Hall
Berkeley, CA 94720

Peitong Duan

(919) 423-6431
peitongd@berkeley.edu

Home Address:
955 Escalon Ave
Sunnyvale, CA 94085

Education

University of California, Berkeley
Candidate for PhD in Computer Science ***Aug 2020 - Jun 2025 (Expected)***

Massachusetts Institute of Technology
M.Eng. in Computer Science (Artificial Intelligence Concentration) ***Feb 2017***
Thesis Title: Beagle: Automated Extraction and Interpretation of Visualizations from the Web
GPA: 5.0/5.0
B.Sc. in Computer Science and Engineering, with a Minor in Mathematics ***Jun 2016***
GPA: 4.9/5.0

Publications

"Optimizing User Interface Layouts via Gradient Descent", **Peitong Duan**, Casimir Wierzynski, Lama Nachman. *CHI 2020* [paper][arXiv][video]

"Beagle: Automated Extraction and Interpretation of Visualizations from the Web", Leilani Battle, **Peitong Duan**, Zachery Miranda, Dana Mukusheva, Remco Chang, Michael Stonebraker, *CHI 2018* [paper]

Work and Research Experience

Google
Researcher (6-month contract) ***Mar 2020 – Sep 2020***
- Developed deep learning methods to decode input touchpoints (typos included) to English text and also for transliteration of English to other languages
- Utilized Tensorflow and Sonnet to build, train, and evaluate neural network models
- Wrote code to perform distributed model training on TPUs to significantly reduce training time

Intel
AI Research Resident, Intel AI Labs (1-year program) ***Sep 2018 – Oct 2019***
- Lead a research project to develop a technique using gradients of a trained neural network model to optimize layouts of mobile user interfaces (UI)
- Programmatically generated 100 random layouts and hand-crafted 5 good layouts and 3 bad layouts for a single UI to build the dataset used to fit the neural network model. A layout is defined by the size and location of each element (e.g. button) in the UI.
- Utilized psiTurk, HTML, Javascript, and CSS to build a mobile app to crowdsource task completion times and error rates for each of the 108 layouts on Amazon Mechanical Turk. Performance data for over 100K tasks were crowdsourced.
- Extended an existing hierarchical LSTM model that predicts completion times of item selections on vertical menus (lists) to predict task performance (a metric combining task completion time and error rate) of a variety of task types on mobile UIs with an assortment of elements
- Used PyTorch to build, train, and evaluate this extended model. After evaluation via 6-fold cross validation on the crowdsourced dataset, the model achieved a target-level R^2 that is better than the current state of the art.
- Developed a gradient-descent based algorithm that uses gradients of the trained neural network to make updates to the UI's layout parameters optimizing for predicted task performance. Penalty functions are added to the algorithm to prevent undesirable situations (e.g. overlapping UI elements) and to enforce customizable desired characteristics in the output layout (e.g. a minimum size constraint for a particular UI element)
- Applied this optimization algorithm on several layouts of two different mobile UIs. A UI that the model has not been trained on was included to assess the model's generalizability
- Verified task performance improvements in the optimized layouts for both UIs experimentally via crowdsourcing. The crowdsourced results showed improvements of up to 9.2 percent.
- Published a first-author paper of this research to CHI 2020 [paper][arXiv][video]
- Gave a 60-minute talk about this research at Intel with over 60 attendees

Google

Software Engineer, Local Search Quality ***Jul 2017 – Sep 2018***

- Enhanced the quality of answer cards (e.g. knowledge cards) for queries in several non-English languages by improving the features for machine learning models in various stages of the answering pipeline and evaluating these changes
- Worked on a project that aims to improve answers to queries pertaining to local businesses (e.g. queries asking for the business's phone number)
- Utilized machine learning techniques to build an answer scoring model for the local businesses project and using Google's internal tools to construct pipelines for training/test data collection from logs

Software Engineer, Newsstand ***Jan – Jul 2017***

- Created an Android card that allows users to opt into receiving email notifications from the Newsstand App
- Added support for storing Ad settings on the Magazine level, so when a user modifies the Ad setting on a particular issue, these settings would be applied to all issues under the same magazine
- Implemented the capture of various user engagement metrics from logs (e.g. the duration users spent on each article) and analyzed these metrics to evaluate the efficacy of particular features of the App

Software Engineering Intern, Structured Results ***Jun – Aug 2015***

- Utilized C++, Java, and Google's internal frameworks to create a date filter for the city events list that appears below the search bar when a user searches for a city's events, and prepared a design document used for its launch approval meeting
- Fixed a bug in Google Search to ensure the venue link for each event in the Knowledge Panel links to the correct venue

MIT Computer Science and Artificial Intelligence Laboratory (CSAIL)

Master's Researcher, Database Group, Supervisor: Prof. Michael Stonebraker ***Sep 2015 – Feb 2017***

- Collected large corpora of charts in SVG (Scalable Vector Graphics) format by scraping visualization rich sites and the general web
- Constructed a random forest classifier that labels a SVG visualization with its type (e.g. scatter plot) that supports any set of chart types (with training), and it achieved high accuracy (up to 97 percent) when evaluated on the collected corpora
- Incorporated the classifier into a system that collects all SVG visualizations from a list of URLs, annotates the visualizations by chart type, and extracts and stores useful information about each visualization into a PostgreSQL database. This system aims to help users analyze how data is visualized on the web.
- Contributed this research work (as second author) to a paper accepted by CHI 2018 and wrote my MEng thesis on this research [thesis link]

Undergraduate Researcher, Haystack Group, Supervisor: Prof. David Karger ***Jan – May 2015***

- Tested out a new language developed at the lab called "Cascading Tree Sheets" (CTS) that allows users to map designs onto one's websites without editing HTML via a set of mapping rules
- Wrote CTS rules to map various designs onto a bare-bone Wordpress theme, including that of another Wordpress theme

Octopart Inc.

Software Engineering Intern ***Jun – Aug 2014***

- Wrote Python code to crawl and scrape websites for unstructured data on electronic parts and collaborated on modifying the web-crawling infrastructure so that crawl jobs can run on spot instances
- Wrote and updated geo-targeting code to redirect international user clicks on the Octopart site to distributor sites with the appropriate language and country settings
- Created a new Octopart blog on Wordpress and edited CSS files of the Theme to customize blog layout and wrote plug-ins to add additional custom features to the blog
- Wrote scripts to migrate all posts from the old blog to the new Wordpress blog

MIT Department of Electrical Engineering and Computer Science

Discrete Math for Computer Science Grader ***Sep – Dec 2013***

- Graded students' solutions to problem sets and wrote new problems for use during class problem solving sessions

Principal Financial Group
Data Management Intern ***Jun – Aug 2013***

- Collaborated on a project analyzing data on differences between two databases that would ideally be identical
- Utilized SQL, VBA, and Python to process data and apply analysis metrics, and used Excel to generate visualizations
- Participated in a Code Jam and collaborated on the creation of the winning Android App for United Way Volunteers

Selected Projects

Predicting Chemical Reactions using Neural Networks and Molecular Convolutional Graphs: We constructed a neural network that predicts if a given molecule is a product of two input reactants. The network took molecular convolutional graph representations of the molecules as input. I came up with a novel way of generating negative samples by replacing actual products with their structural isomers. [final project paper link]

Analysis of Noncoding Mutations in Cancer: We used two Convolutional Neural Networks to compute the functional impact of the non-coding mutations of 4000 cancer patients, and I analyzed the functional effects with K-Means Clustering, Correlation Matrices, and Hierarchical Clustering. [final project paper link]

Security Analysis of the Boston Symphony Orchestra's (BSO) Website and Mobile Application: With permission from the BSO, we analyzed the security vulnerabilities of their website and App to common attacks such as Cross Site Request Forgery, SQL Injection, and Cross Site Scripting. We reported our findings and suggestions for security improvements to the BSO. [final project paper link]

Rainforest Rescue Mission: For a web programming competition, we built a website that encourages children to develop environmentally friendly habits, while teaching them about the rainforest. The website is in the form of a game where users navigate through the rainforest and rescue animals along the way by completing a number of eco-friendly tasks. I designed and implemented all front-end features and assisted with back-end development.

Awards

UC Berkeley EECS Excellence Award, for an outstanding academic record **2020**

Honorable Mention in MIT IAP 6.470 Web Programming Competition (Main Division) **2014**

US National Chemistry Olympiad Top 20 National Finalist and attended Study Camp to train for the International Chemistry Olympiad **2010**

Leadership and Activities

Reviewer for CHI 2020 (ACM CHI Conference on Human Factors in Computing Systems) **2020**

Outreach Committee Member of the MIT Undergraduate Society of Women in Mathematics **2011 – 2012**

Figure Skating **2006 - 2017**

Skills

Python, Machine Learning, PyTorch, Tensorflow, Crowdsourcing, Amazon Mechanical Turk, psiTurk, Java, Natural Language Processing, C++, Scikit-learn, Pandas, MATLAB, Julia, OpenCV, LaTeX, HTML, Javascript, CSS, MongoDB, Node, Bootstrap