Kristy Emelyn Lee

E: kristylee@berkeley.edu | **C:** (408)784-1561 | **LinkedIn:** /kristylee1 | **GitHub:** kristyelee | **Google Scholar:** Kristy Lee

Education

University of California, Berkeley August 2018 - Expected: May 2022
Bachelor's, Computer Science, Applied Mathematics – Cumulative GPA: 3.933, Dean's List Honoree (top 10%), Honors to Date, EECS Honors Program [Mathematics], Upsilon Pi Epsilon CS Honor Society Member (top 33% of students in CS major)
Coursework: NLP (Graduate Class) [A+], Machine Learning [A+], Operating Systems*, Artificial Intelligence, Computer Vision + Computational Photography*, Computer Security, Algorithms, Data Structures, Computer Architecture, Data Science, Linear Algebra

Skills and Tools

Proficient/Expert: Java, Python
Familiar: Keras, PyTorch, TensorFlow, NumPy, Matplotlib, MATLAB, Swift, React.js, Flask, JavaScript, C, SQL, Go, HTML, Unix
Certifications: deeplearning.ai: Deep Learning, NLP; UMich: Using Databases with Python, Using Python to Access Web Data

Work Experience

AI/ML Domains Software Engineering Intern at Apple – Siri Experience (Siri Messages) Team May 2021–Aug. 2021
- Work on classification of important messages for Siri Messages.

Undergraduate Researcher at Berkeley NLP Group – Berkeley Artificial Intelligence Research Aug. 2020–Present
- Research with Ruiqi Zhong, PhD candidate under advisor Professor Dan Klein.
- Formulate zero-shot classification as a question answering task. Finetune the UnifiedQA model, which is based on a larger pretrained model T5 (Text-to-Text Transfer Transformer model). Use TPUs and Google Cloud Storage Bucket for model training. Contributed to EMNLP paper submission as second author.
- Work on inducing discrete representations from unlabeled data to improve models that translate language into executable commands in a grounded environment.

Software Engineering Intern at Apple – iOS System Experience Team May 2020–Aug. 2020
- Designed and implemented machine learning classifier for incoming text-based bug reports using Swift NLP APIs.
- Integrated text classification model into tools and apps for categorization of filed reports for iOS/iPadOS software issues, improving team's internal tools by allowing for the grouping and more efficient resolution of software issues.
- Built a MacOS app with a word tagging system tool used to quickly identify the subject/problematic element of every bug report.

Deep Learning Undergraduate Researcher at Simons Institute for the Theory of Computing – UC Berkeley Jan. 2020–May 2020
- Researched the generalization gap associated with large batch training under postdoctoral researcher Xiaowu Dai.
- Analyzed sharpness of minima associated with training across multiple batch sizes to examine sizes of generalization gap.
- Implemented and experimented with multiple neural network model architectures using PyTorch and Keras. Link to code.

Software Developer Intern at Berkeley Educators Engineering Software – UC Berkeley Jan. 2019–Mar. 2020
- Contributing to the programming of a wait-time calculation that provided better estimates of individual wait-time on CS61A's online office hour (OH) queue using past data. Contributed to front-end/back-end development for CS61A OH queue website using React.js, Flask.
- Identified and fixed bugs found in OH website used by 2000+ students taking the CS 61A course.

Deep Learning Undergraduate Researcher at Berkeley SETI Research Center Sep. 2019–Dec. 2019
- SETI: Breakthrough Listen Project under Dr. Steve Croft. Trained convolutional neural network and implement de-dispersion algorithms for classification and analysis of fast radio bursts (FRBs) related to extraterrestrial intelligence, distinguishing FRBs from radio frequency interference. Link to code.
- Achieved 90% testing accuracy. Integrated training models into Breakthrough Listen signal-retrieval pipeline.

Projects

Neural Machine Translation System
- Uses sequence-to-sequence model constructed with LSTM encoder and LSTM decoder with attention mechanism to translate German sentences from the Multi30K dataset to English sentences.
- Model attains 37.2 BLEU on test set, using beam search to generate the English sentence predictions.

Smart To-Do List iOS App
- Uses NLP techniques and classification to provide recommendations for tasks to add to to-do list based on user's previous activity.
- Implements creative idea of generating art in background as user completes tasks, subtly compelling users to finish to-do list tasks.

Teaching

Computer Science 170 (Efficient Algorithms and Intractable Problems) Course Reader Jan. 2021 – May 2021
Computer Science 61A (Structure/Interpretations of Computer Programs) Course Tutor Aug. 2020 – Dec. 2020