Brian **Lester**

99 E Middlefield Rd. Mountain View, CA. 94043

(734) 834-2546 | blester125@gmail.com | blester125.com | blester125 | blester125 | Brian Lester | Brian Lester

Work Experience

Google

Mountain View, California

AI Resident

2020–Present

Working on Natural Language Processing and Understanding

Interactions

Ann Arbor, Michigan

Machine Learning Engineer

2018–2020

Driving Deep Learning adoption. Built, maintained, designed, and productionized deep learning models to facilitate customer care interactions. Used for applications like dialogue systems and the retrieval, ranking, and categorization of social media posts (Twitter, Facebook, and web forums) to increase the effectiveness of customer service representatives.

- **Mead-Baseline:** I improve and maintain Mead-Baseline—our open-source, deep-learning toolkit and the sole method for training and deploying deep learning models in our entire company. Mead-Baseline doubles as our research platform and because of its extensible design I have used it for both novel research and to reproduce and vet current literature. Mead-Baseline includes my performant implementation of a CRF. My CRF is ten times faster than our original implementation making it undoubtedly one of the fastest ways to train a bLSTM-CRF for sequence tagging. My CRF also yields strong performance by helping a user inject constraints into the model as dictated by their tagging scheme. I have also accelerated the implementations of other complex modules such as Beam Search.
- **NLU Annotation Design:** Designed the label space for the Natural Language Understanding module of a customer self service dialogue system in the technical support domain. My design focused on using general intents, complex entities, and relations to cover a diverse and complex conversation space.
- **Deep Learning Training Platform:** Designed and created our cloud-native, model-training platform used by the whole company to train deep models. Training pipelines are built declaratively by specifying a directed acyclic graph where each node represents some unit of work to be run inside a Docker container. This graph is then executed on our Kubernetes cluster. Using Kubernetes helps us effectively manage the usage of shared GPUs as well as schedule many jobs in parallel allowing Hyper Parameter Optimization at scale. This platform enables building complex multi-step pipelines (often one step is training a model via Mead-Baseline) that transform raw data into a model that is ready for production.
- **Deep Learning Deployment Platform:** Designed and implemented our model server. This server, backed by TensorFlow Serving, enables rich Natural Language Understanding via cascading calls to a series of deep learning models. Both the model server itself and the TensorFlow Serving backend are deployed via Kubernetes. This model server is currently powering NLU for several production dialogue systems.
- **Production Models:** Created a wide range of machine learning models that are used in production. Model architectures and tasks range from ConvNets for classification and intent detection, bLSTM-CRF taggers for general NER and client-specific slot filling, mention-ranking models for relation extraction, to Transformer-based seq2seq models that generate suggested agent responses. Production models were also calibrated to produce reliable scores via post-hoc methods and secondary confidence models.

Trove

Ann Arbor, Michigan

Lead Machine Learning Research Engineer

2017–2018

Replaced heuristics with Machine Learning to enhance features. Created a model training and serving platform that processed 200 million emails a day. Provided technical leadership to the data science team.

- **Question Detection:** Text classification with a ConvNet to find sentences that contain questions. The presence of questions in an email was used as a feature in our email social graph.
- **Coreference Resolution:** A neural ranking model was used to find coreferent mentions in the text and provide context to users.
- **Bot detection:** Using lexical features, as well as connectivity information in the email social graph, we identified bot accounts.

Visteon Corporation

Van Buren, Michigan

Software Engineering Intern

2015

Optimized Voice Recognition in Mazda Cars. Designed an adaptive system to minimize voice recognition errors. We patented this system (US 9,984,688 B2) and it is used in production for Mazda Cars.

Conference Publications

Intent Features for Rich Natural Language Understanding

B. Lester, S. Ray Choudhury, R. Prasad, S. Bangalore

Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Papers, 2021, Online

The Power of Scale for Parameter-Efficient Prompt Tuning

B. Lester, R. Al-Rfou, N. Constant

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2021, Online

Constrained Decoding for Computationally Efficient Named Entity Recognition Taggers

B. Lester, D. Pressel, A. Hemmeter, S. Ray Choudhury, S. Bangalore

Findings of the Association for Computational Linguistics: EMNLP 2020, 2020, Online

An Effective Label Noise Model for DNN Text Classification

I. Jindal, D. Pressel, B. Lester, M. Nokleby

Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2019, Minneapolis, Minnesota

Workshop Publications

iobes: Library for Span Level Processing

B. Lester

Proceedings of Second Workshop for NLP Open Source Software (NLP-OSS), 2020, Online

Baseline: Strong, Extensible, Reproducible, Deep Learning Baselines for NLP

D. Pressel, B. Lester, S. R. Choudhury, M. Barta, Y. Zhao, A. Hemmeter

Neural Information Processing Systems Open Source Software Workshop (2018). 2018

Baseline: A Library for Rapid Modeling, Experimentation and Development of Deep Learning Algorithms targeting NLP

D. Pressel, S. Ray Choudhury, B. Lester, Y. Zhao, M. Barta

Proceedings of Workshop for NLP Open Source Software (NLP-OSS), 2018, Melbourne, Australia

Other Publications

SPoT: Better Frozen Model Adaptation through Soft Prompt Transfer

T. Vu, B. Lester, N. Constant, R. Al-Rfou, D. M. Cer

arXiv preprint arXiv:2110.07904 (Oct. 2021). 2021

Finetuned Language Models Are Zero-Shot Learners

J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, Q. V. Le

arXiv preprint arXiv:2109.01652 (Sept. 2021). 2021

Leader: Prefixing a Length for Faster Word Vector Serialization

B. Lester

arXiv preprint arXiv:2009.13699 (2020). 2020

Multiple Word Embeddings for Increased Diversity of Representation

B. Lester, D. Pressel, A. Hemmeter, S. R. Choudhury, S. Bangalore

arXiv preprint arXiv:2009.14394 (2020). 2020

Dynamically Adjusting a Voice Recognition System

B. Lester, S. M. Panainte

US Patent 9,984,688, 2018

Presentations

NeurIPS Spotlight Talk on Mead-Baseline *NeurIPS OSS Workshop*

Deep Learning *December 2018*

A spotlight talk, at the Open Source Software workshop at NeurIPS 2018, highlighting the advantages of using our open-source, model-building toolkit, Mead-Baseline, which provides high-level model abstractions, correct evaluation metrics, and fast runtimes. Mead-Baseline is the foundation of all Deep Learning work, in both research and production, at Interactions.

Confidence and Calibration of Neural Network Models *EMU ML Conference*

Deep Learning *March 2020,* [1]

In most settings a model is trained and evaluated on a held out test set; however, in some domains one would rather have no answer than an uncertain one. Modern Neural Networks are notoriously miscalibrated, that is, they are far too confident in their predictions. This talk discusses techniques used to evaluate model calibration and rejection of examples based on confidence as well as a summary of current work to produce high-fidelity confidence scores in the NLU component of a real world Dialogue System.

Padding in Neural Networks for Natural Language Processing *A2D-NLP*

Natural Language Processing *February 2020*

A deep dive into the implementation of many NLP building blocks focusing on the subtleties of the padding and masking needed to correctly process batches. It addressed the masking needed for operations like token level losses, mean pooling, and attention; complex operations such as the CRF forward algorithm and Viterbi Decoding; and subtle operations that you might not expect to need masking like max pooling following a 1D convolution.

Optimization via NumPy and Cython *Michigan Python Meetup*

Numerical Computation *January 2020*

Framed around the problem of calculating the pairwise Manhattan distance between a collection of points, this talk walks through a series of optimizations that introduce core NumPy concepts like Broadcasting and Vectorization before using a bespoke Cython implementation to scale up to 10s of thousands of points in just seconds compared to the multi hour runtime of a pure Python solution.

Input Representations of Deep Neural Networks *PyData Ann Arbor*

Deep Learning *October 2017*

A talk on creating robust models via learned character compositional input representations based on bLSTMs to support an open vocabulary for Deep Neural Networks.

Service & Public Scholarship

Reviewing

2019	CoNLL
2020–2021	Computer Speech and Language

Public Scholarship

2020 [1] **a^2-dlearn,** Helped organize, recruit speakers, and acquire funding though sponsorships

Skills

Machine and Deep Learning	Extensive experience using Pytorch, Tensorflow, and Scikit-learn for NLP applications.
Infrastructure	Building and deploying with Kubernetes, Docker, Flux, Prometheus, MongoDB, and GitLab CI/CD.
Toolkits	NumPy, Pandas, Faiss, Gensim, SpaCy, NLTK, SciPy, and Matplotlib.
Languages	Python, Cython, Java, C, Javascript, C++, and LaTeX.

[1] Canceled due to COVID-19

Projects

Text Rank *Personal Project*

Natural Language Processing

An implementation of Text Rank in Python that reproduces the results from the original paper. I currently use it as a vehicle for personal research: investigating if deep learning based sentence similarities will yield better summaries. Results forthcoming.

String Distance *Open Source Library*

Natural Language Processing

A collection of various minimum edit distance algorithms as well as token based methods like Jaccard overlap. The algorithms are implemented in Cython and scale well. The library can compute minimum edit distances between entire Wikipedia pages in under a second.

Quick KNN *Open Source Library*

Information Retrieval

Implementations of Locality Sensitive Hashing. Supports using MinHash to approximate Jaccard similarity and Random Hyperplanes for a cosine based LSH. This library enables users to find similar items in very large corpora quickly.

Dependency Parsing *Personal Project*

Natural Language Processing

Dependency parsing via Deep Learning models written in PyTorch. Supports training parsers via dynamic oracles using either the Arc Eager or the Arc Hybrid transition scheme or a Graph based parser using Biaffine Attention.

Decomposable Attention *Personal Project*

Natural Language Processing

A reimplementation of the paper “A Decomposable Attention Model for Natural Language Inference” in DyNet. I was able to reproduce the paper results via extensive hyper parameter tuning.

Steering Angles *Personal Project*

Deep Learning

Used a deep convolutional neural network written in TensorFlow to create a steering agent for an autonomous car. Reduced training time of network using architecture optimizations like residual connections and batch normalization.

Multi-Digit Recognition *Personal Project*

Deep Learning

Multi-Digit sequence recognition in natural scenes via a deep, convolutional neural network written with TensorFlow. The end-to-end network allowed optimization of the whole process rather than dividing it into localization, segmentation, and recognition tasks.

RFC 793 Transmission Control Protocol *University of Pittsburgh*

Computer Networking *Fall 2015*

A full implementation of RFC 793 TCP written in C++. It insures reliable communication between two hosts and supports multiple data packets in flight using a “Go Back N” strategy.

Particle Simulation *University of Pittsburgh*

High Performance Computing *Spring 2015*

Simulation of particle interactions written in MPI. The simulation was parallelized via a ring algorithm and was run on the Stampede super computer.

Secure File Sharing System *University of Pittsburgh*

Applied Cryptography and Network Security *Spring 2016*

A distributed, group-based file sharing system that is secured using cryptographic techniques such as symmetric and public key cryptography, ephemeral key exchange, digital signatures, and two factor authentication.

Education

University of Pittsburgh

Pittsburgh, PA

Double major, 3.51. **Computer Science** with Honors, 3.86. **Neuroscience**, 3.19.

2012–2016

Shanghai American School

Shanghai, China

High School

2008–2012

Advanced Placement Curriculum. Team lead for FIRST Robotics Competition.