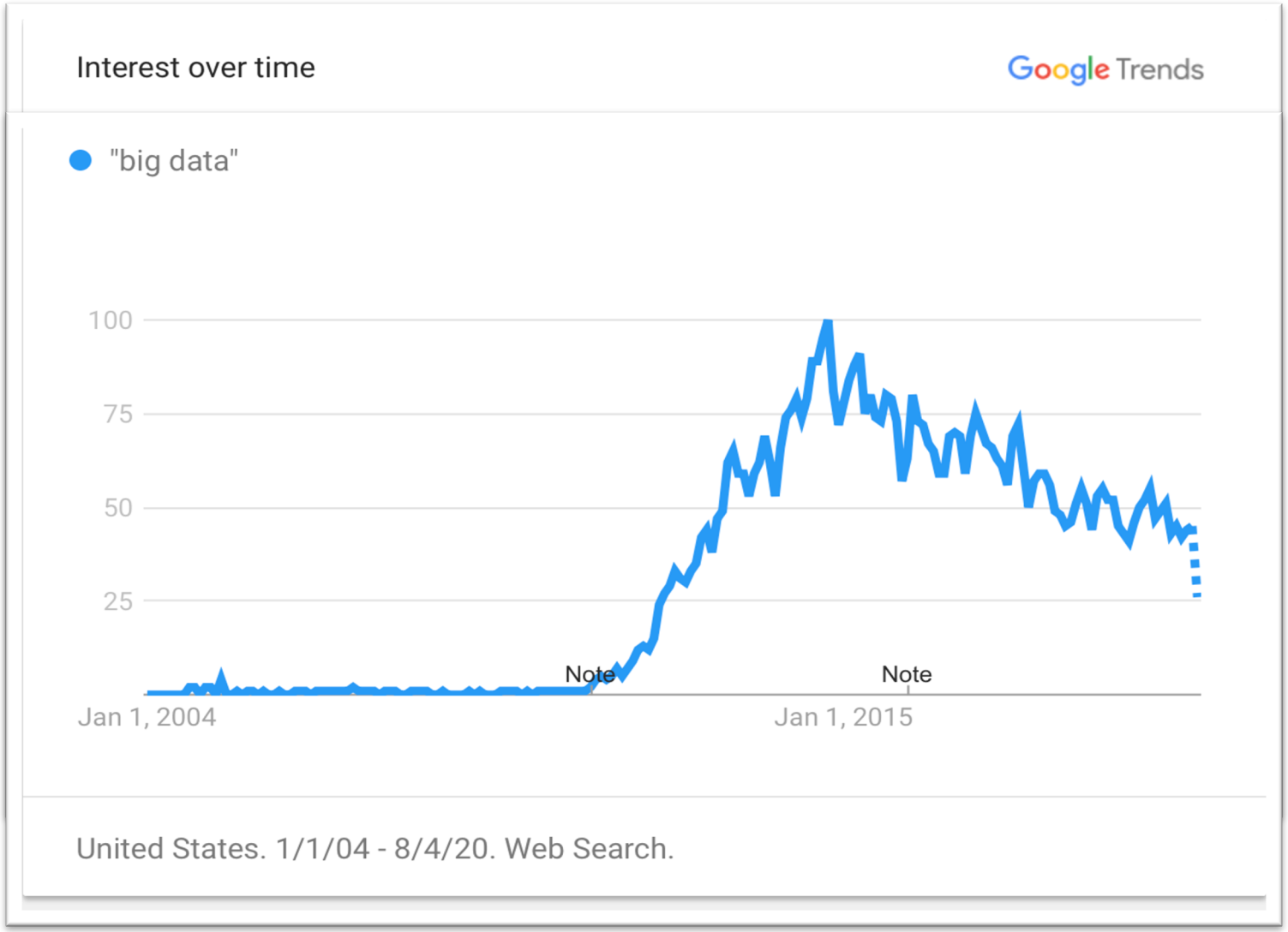
Introduction

Dr. Alex Williams

COSC 425: Introduction to Machine Learning
Fall 2020 (CRN: 44874)

THE UNIVERSITY OF TENNESSEE KNOXVILLE

Welcome!


Interest over time
Google Trends
"big data"
100
75
50
25
Note
Note
Jan 1, 2004
Jan 1, 2015
United States. 1/1/04 - 8/4/20. Web Search.

Learning representations by back-propagating errors

David E. Rumelhart*, Geoffrey E. Hinton† & Ronald J. Williams*

* Institute for Cognitive Science, C-015, University of California, San Diego, La Jolla, California 92093, USA
† Department of Computer Science, Carnegie-Mellon University, Pittsburgh, Philadelphia 15213, USA

We describe a new learning procedure, back-propagation, for networks of neurone-like units. The procedure repeatedly adjusts the weights of the connections in the network so as to minimize a measure of the difference between the actual output vector of the net and the desired output vector. As a result of the weight adjustments, internal 'hidden' units which are not part of the input or output come to represent important features of the task domain, and the regularities in the task are captured by the interactions of these units. The ability to create useful new features distinguishes back-propagation from earlier, simpler methods such as the perceptron-convergence procedure[1].

There have been many attempts to design self-organizing neural networks. The aim is to find a powerful synaptic modification rule that will allow an arbitrarily connected neural network to develop an internal structure that is appropriate for a particular task domain. The task is specified by giving the desired state vector of the output units for each state vector of the input units. If the input units are directly connected to the output units it is relatively easy to find learning rules that iteratively adjust the relative strengths of the connections so as to progressively reduce the difference between the actual and desired output vectors[2]. Learning becomes more interesting but more difficult when we introduce hidden units whose actual or desired states are not specified by the task. (In perceptrons, there are 'feature analysers' between the input and output that are not true hidden units because their input connections are fixed by hand, so their states are completely determined by the input vector: they do not learn representations.) The learning procedure must decide under what circumstances the hidden units should be active in order to help achieve the desired input–output behaviour. This amounts to deciding what these units should represent. We demonstrate that a general purpose and relatively simple procedure is powerful enough to construct appropriate internal representations.

The simplest form of the learning procedure is for layered networks which have a layer of input units at the bottom; any number of intermediate layers; and a layer the top. Connections within a layer or fro layers are forbidden, but connections can layers. An input vector is presented to the the states of the input units. Then the states layer are determined by applying equations connections coming from lower layers. All have their states set in parallel, but differer states set sequentially, starting at the bo upwards until the states of the output units

The total input, x_j, to unit j is a linear func y_i, of the units that are connected to j and on these connections

$$x_j = \sum_i y_i w_{ji}$$

Units can be given biases by introducing an unit which always has a value of 1. The w input is called the bias and is equivalent to opposite sign. It can be treated just like the

A unit has a real-valued output, y_j, whi function of its total input

$$y_j = \frac{1}{1+e^{-x_j}}$$

† To whom correspondence should be addressed.

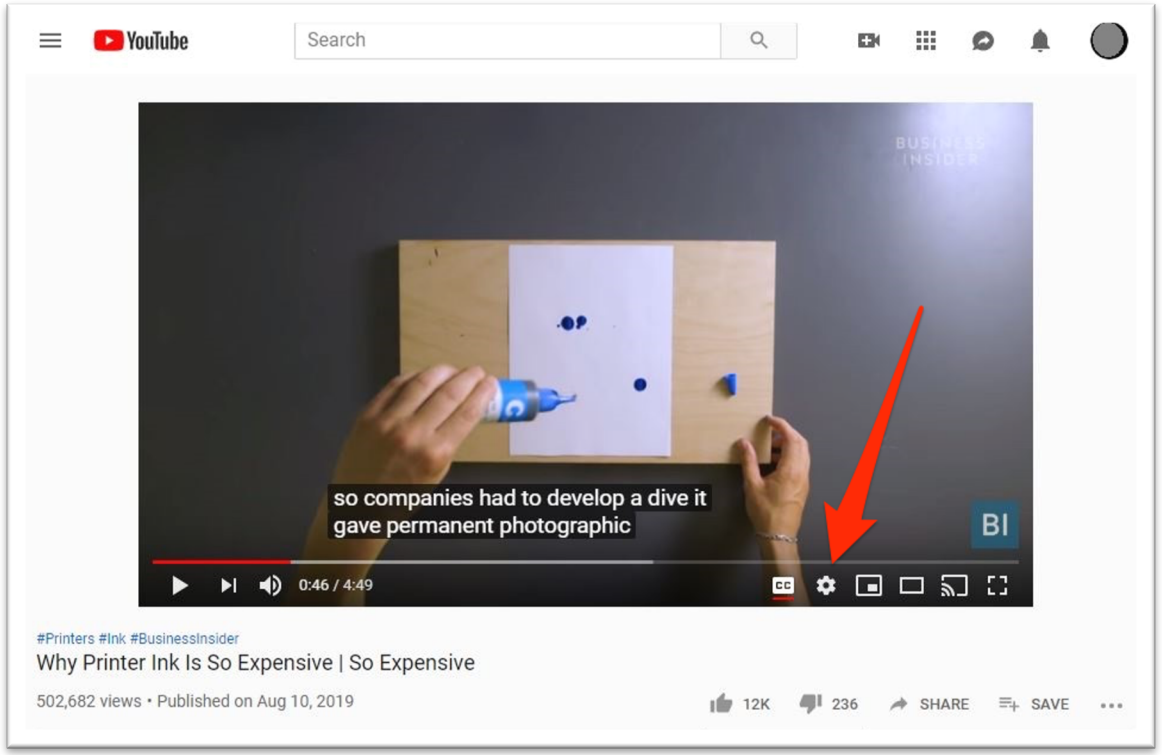
WHY IS SO MUCH MEMORY NEEDED FOR DEEP NEURAL NETWORKS?

Written by
Jamie Hanlon

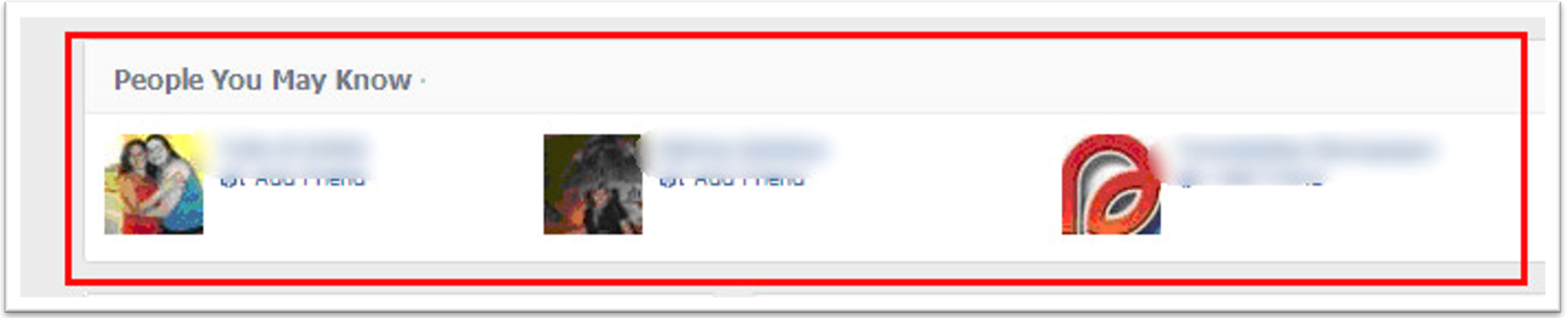
Posted
4 years ago

Memory is one of the biggest challenges in deep neural networks (DNNs) today. Researchers are struggling with the limited memory bandwidth of the DRAM devices that have to be used by today's systems to store the huge amounts of weights and activations in DNNs. DRAM capacity appears to be a limitation too. But these challenges are not quite as they seem.

Computer architectures have developed with processor


YouTube
Search
so companies had to develop a dive it
gave permanent photographic
BI
0:46 / 4:49
#Printers #Ink #BusinessInsider
Why Printer Ink Is So Expensive | So Expensive
502,682 views • Published on Aug 10, 2019
12K
236
SHARE
SAVE


Close
10:09
It looks like you've
taken a hard fall.
SOS
EMERGENCY
SOS
I'm OK

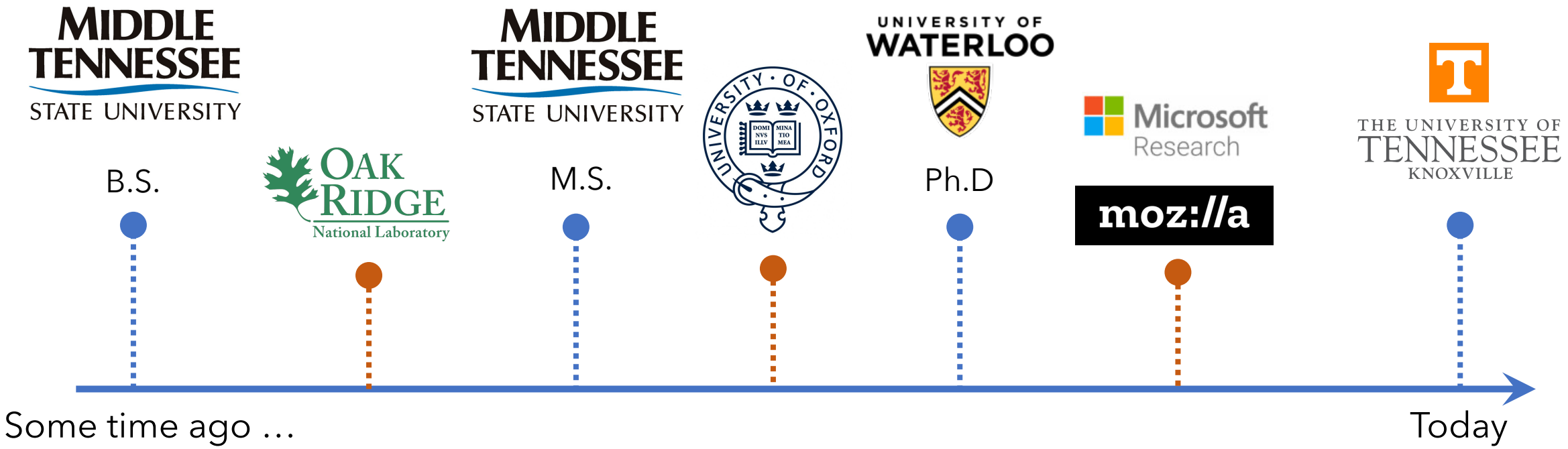

People You May Know

Today's Objectives

By the end of this video, you should:

1. Have a clear vision of expectations for COSC 425.
2. Understand the breadth of COSC 425's syllabus.

Who am I?


MIDDLE TENNESSEE STATE UNIVERSITY
B.S.
OAK RIDGE National Laboratory
MIDDLE TENNESSEE STATE UNIVERSITY
M.S.
UNIVERSITY OF OXFORD
UNIVERSITY OF WATERLOO
Ph.D
Microsoft Research
moz://a
THE UNIVERSITY OF TENNESSEE KNOXVILLE
Some time ago …
Today

What do I do?

How can we build interactive systems that fundamentally make the nature of work **better**?

What do I do?

Supporting Workplace Detachment and Reattachment with Conversational Intelligence

Alex C. Williams[1], Harmanpreet Kaur[2], Gloria Mark[3],
Anne Loomis Thompson[4], Shamsi T. Iqbal[4], Jaime Teevan[4]
[1]University of Waterloo, [2]University of Michigan, [3]University of California Irvine, [4]Microsoft Research
alex.williams@uwaterloo.ca, harmank@umich.edu, gmark@uci.edu,
{annelo, shamsi, teevan}@microsoft.com

ABSTRACT
Research has shown that productivity is mediated by an individual's ability to detach from their work at the end of the day and reattach with it when they return the next day. In this paper we explore the extent to which structured dialogues, focused on individuals' work-related tasks or emotions, can help them with the detachment and reattachment processes. Our inquiry is driven with *SwitchBot*, a conversational bot which engages with workers at the start and end of their work day. After preliminarily validating the design of a detachment and reattachment dialogue framework with 108 crowdworkers, we study SwitchBot's use in-situ for 14 days with 34 information workers. We find that workers send fewer e-mails after work hours and spend a larger percentage of their first hour at work using productivity applications than they normally would when using SwitchBot. Further, we find that productivity gains were better sustained when conversations focused on work-related emotions. Our results suggest that conversational bots can be effective tools for aiding workplace detachment and reattachment and help people make successful use of their time on and off the job.

Author Keywords
Detachment; reattachment; resumption; productivity; bot.

ACM Classification Keywords
H.5.m [Info. Interfaces and Presentation (e.g., HCI)]: Misc.

INTRODUCTION
Adequate recovery from work is vital for replenishing resources depleted during work hours and maintaining good psychological health and well-being [71]. Among the many influential factors that promote recovery, the ability to psychologically detach from work is recognized as particularly important for its core role in facilitating mental rejuvenation and refreshment in subsequent workdays [14,68]. Recent research has posited that rebuilding a mental connection with one's work before the start of the workday (i.e., reattaching with work) is equally as important for ensuring workplace engagement and productivity, particularly in the morning [64]. A variety of approaches, ranging from brief planning to extensive therapy, have been proposed and studied in support of these goals. The efficacy of these techniques ranges with much variation, making this an active and open area of research for novel interventions.

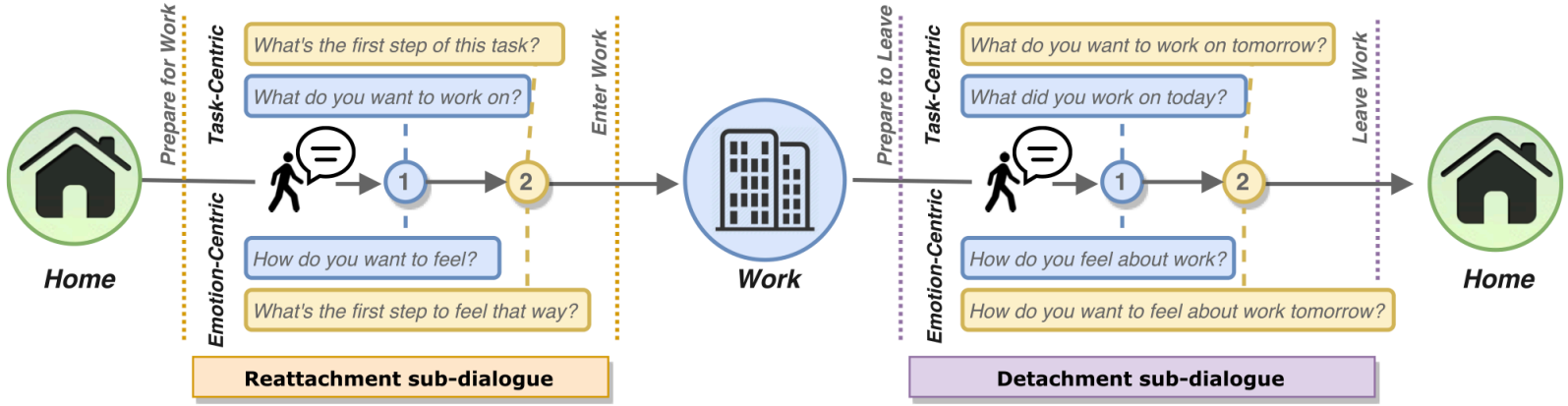
In this work, we study the extent to which structured dialogues, focusing on individuals' work-related tasks or emotions, can help them with the detachment and reattachment processes. Ranging from paper-based diaries to online surveys, an array of possible intervention types exists for administering such dialogues to individuals. Prior work, however, emphasizes the importance of social support that individuals may receive from others during the detachment process [27,61]. While this constraint belies many types of technical interventions, conversational intelligence, or *bots*, embraces these scenarios with prior research demonstrating their ability to provide such social support through active listening and guided conversation [31,70] as shown by systems such as ELIZA [70] and ALICE [31]. Further, conversational systems are known to offer the added benefit of inducing feelings of accountability in individuals when setting goals [7], a process that generally occurs during both the detachment process and the reattachment process.

We present and study *SwitchBot,* a conversational bot that helps workers detach from and reattach with their work. By identifying similarities between interruption and task resumption with detachment and reattachment, we leverage prior research to design two dialogue styles for SwitchBot, one that is task-centric and the other emotion-centric. We validated the practical value of each dialog via an online study with 108 crowd workers, and then conducted an in-situ study for 14 days where 34 information workers used SwitchBot as they began and concluded their workday. Our results show SwitchBot's dialogues were an effective intervention for supporting detachment from and reattachment with the workplace. In particular, we find that:

- Participants felt more productive and engaged during the first hour of their work when using SwitchBot;

CHI 2018, April 21–26, 2018, Montreal, QC, Canada

ACM 978-1-4503-5620-6/18/04...$15.00
https://doi.org/10.1145/3173574.3173662

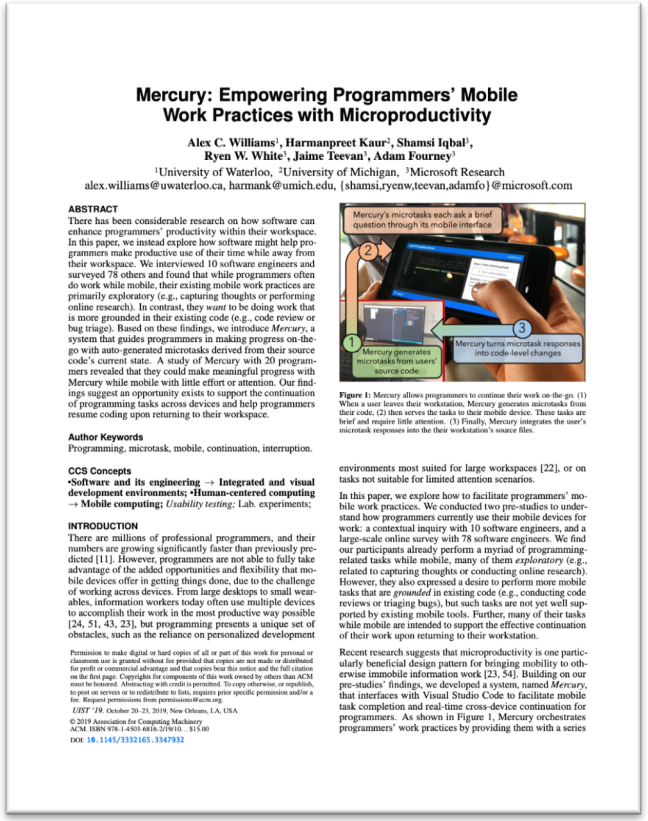


SwitchBot (*CHI '18)*

Conversational Agents for Work

Additional Information is available online.

What do I do?



Mercury: Empowering Programmers' Mobile Work Practices with Microproductivity

Alex C. Williams[1], Harmanpreet Kaur[2], Shamsi Iqbal[3], Ryen W. White[3], Jaime Teevan[3], Adam Fourney[3]
[1]University of Waterloo, [2]University of Michigan, [3]Microsoft Research
alex.williams@uwaterloo.ca, harmank@umich.edu, {shamsi,ryenw,teevan,adamfo}@microsoft.com

ABSTRACT
There has been considerable research on how software can enhance programmers' productivity within their workspace. In this paper, we instead explore how software might help programmers make productive use of their time while away from their workspace. We interviewed 10 software engineers and surveyed 78 others and found that while programmers often do work while mobile, their existing mobile work practices are primarily exploratory (e.g., capturing thoughts or performing online research). In contrast, they *want* to be doing work that is more grounded in their existing code (e.g., code review or bug triage). Based on these findings, we introduce *Mercury*, a system that guides programmers in making progress on-the-go with auto-generated microtasks derived from their source code's current state. A study of Mercury with 20 programmers revealed that they could make meaningful progress with Mercury while mobile with little effort or attention. Our findings suggest an opportunity exists to support the continuation of programming tasks across devices and help programmers resume coding upon returning to their workspace.

Author Keywords
Programming, microtask, mobile, continuation, interruption.

CCS Concepts
•**Software and its engineering** → **Integrated and visual development environments;** •**Human-centered computing** → **Mobile computing;** *Usability testing;* Lab. experiments;

INTRODUCTION
There are millions of professional programmers, and their numbers are growing significantly faster than previously predicted [11]. However, programmers are not able to fully take advantage of the added opportunities and flexibility that mobile devices offer in getting things done, due to the challenge of working across devices. From large desktops to small wearables, information workers today often use multiple devices to accomplish their work in the most productive way possible [24, 51, 43, 23], but programming presents a unique set of obstacles, such as the reliance on personalized development environments most suited for large workspaces [22], or on tasks not suitable for limited attention scenarios.

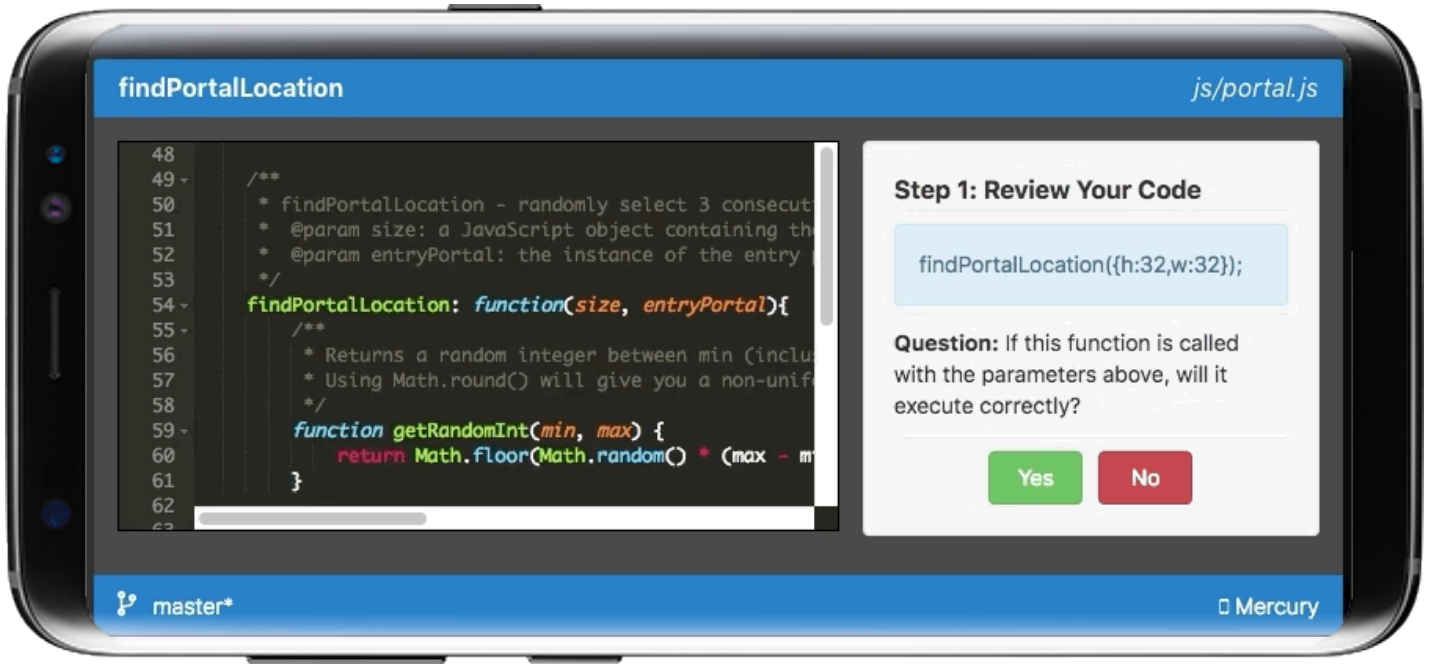
UIST '19, October 20–23, 2019, New Orleans, LA, USA
© 2019 Association for Computing Machinery
ACM. ISBN 978-1-4503-6816-2/19/10…$15.00
DOI: 10.1145/3332165.3347932

In this paper, we explore how to facilitate programmers' mobile work practices. We conducted two pre-studies to understand how programmers currently use their mobile devices for work: a contextual inquiry with 10 software engineers, and a large-scale online survey with 78 software engineers. We find our participants already perform a myriad of programming-related tasks while mobile, many of them *exploratory* (e.g., related to capturing thoughts or conducting online research). However, they also expressed a desire to perform more mobile tasks that are *grounded* in existing code (e.g., conducting code reviews or triaging bugs), but such tasks are not yet well supported by existing mobile tools. Further, many of their tasks while mobile are intended to support the effective continuation of their work upon returning to their workstation.

Recent research suggests that microproductivity is one particularly beneficial design pattern for bringing mobility to otherwise immobile information work [23, 54]. Building on our pre-studies' findings, we developed a system, named *Mercury*, that interfaces with Visual Studio Code to facilitate mobile task completion and real-time cross-device continuation for programmers. As shown in Figure 1, Mercury orchestrates programmers' work practices by providing them with a series

Figure 1: Mercury allows programmers to continue their work on-the-go. (1) When a user leaves their workstation, Mercury generates microtasks from their code, (2) then serves the tasks to their mobile device. These tasks are brief and require little attention. (3) Finally, Mercury integrates the user's microtask responses into the their workstation's source files.

Mobile Programming



Mercury (*UIST '19*)

Additional Information is available online.

What do I do?

The Future of Work

The Perpetual Work Life of Crowdworkers: How Tooling Practices Increase Fragmentation in Crowdwork

ALEX C. WILLIAMS, University of Waterloo, Canada
GLORIA MARK, University of California at Irvine, USA
KRISTY MILLAND, University of Toronto, Canada
EDWARD LANK, University of Waterloo, Canada
EDITH LAW, University of Waterloo, Canada

Crowdworkers regularly support their work with scripts, extensions, and software to enhance their productivity. Despite their evident significance, little is understood regarding how these tools affect crowdworkers' quality of life and work. In this study, we report findings from an interview study (N=21) aimed at exploring the tooling practices used by full-time crowdworkers on Amazon Mechanical Turk. Our interview data suggests that the tooling utilized by crowdworkers (1) strongly contributes to the fragmentation of microwork by enabling task switching and multitasking behavior; (2) promotes the fragmentation of crowdworkers' work-life boundaries by relying on tooling that encourages a 'work-anywhere' attitude; and (3) aids the fragmentation of social ties within worker communities through limited tooling access. Our findings have implications for building systems that unify crowdworkers' work practice in support of their productivity and well-being.

CCS Concepts: • **Human-centered computing** → **Computer supported cooperative work**.

Additional Key Words and Phrases: Crowdwork; Tooling; Work Practice; Interview; Fragmentation.

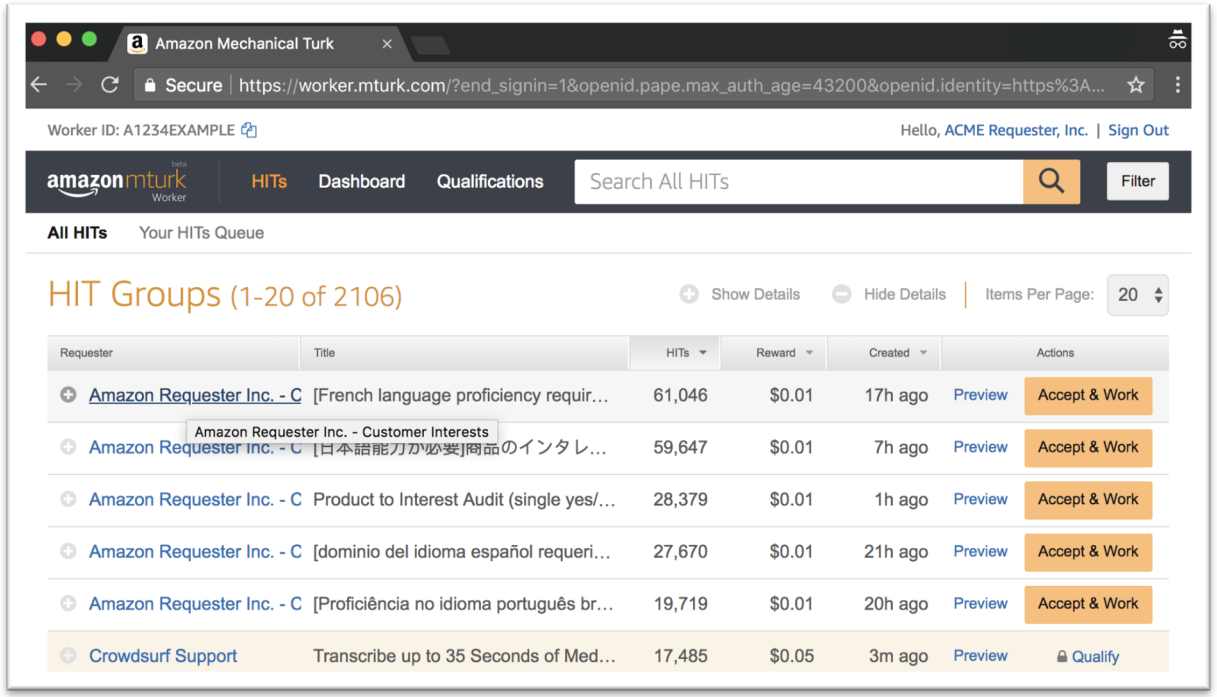
ACM Reference Format:
Alex C. Williams, Gloria Mark, Kristy Milland, Edward Lank, and Edith Law. 2019. The Perpetual Work Life of Crowdworkers: How Tooling Practices Increase Fragmentation in Crowdwork. *Proc. ACM Hum.-Comput. Interact.* 3, CSCW, Article 24 (November 2019), 28 pages. https://doi.org/10.1145/3359126

1 INTRODUCTION

Over the past decade, crowdwork has risen as an established and thriving work practice for thousands of people across the globe [20]. In a 2010 survey of US-based Amazon Mechanical Turk (MTurk) workers, less than 5% of the participants identified as spending 40 hours or more on crowdwork [42]. Recent data-driven analyses of long-term worker activity on MTurk found that the population of crowdworkers on the platform is generally stable, but that "tens of thousands of new workers that arrive on the platform each year" to replace workers that abandon the profession [21, 61]. Beyond MTurk, a 2016 survey found that more than 5 million individuals in the UK are actively engaged in crowdwork [39], and 53% of a 2017 survey's respondents with crowdworkers in Switzerland reported the profession as their full-time job [40]. Spurred by the changing nature

Authors' addresses: Alex C. Williams, University of Waterloo, Canada, alex.williams@uwaterloo.ca; Gloria Mark, University of California at Irvine, USA, gmark@uci.edu; Kristy Milland, University of Toronto, Canada, kristymilland@gmail.com; Edward Lank, University of Waterloo, Canada, lank@uwaterloo.ca; Edith Law, University of Waterloo, Canada, edith.law@uwaterloo.ca. *Disclaimer:* Kristy Milland was previously a crowdworker and a community manager of TurkerNation. She did not contribute to data collection or analysis of this research.

2573-0142/2019/11-ART24 $15.00
https://doi.org/10.1145/3359126



Amazon Mechanical Turk

What do I do?

The Future of Work

The Perpetual Work Life of Crowdworkers: How Tooling Practices Increase Fragmentation in Crowdwork

ALEX C. WILLIAMS, University of Waterloo, Canada
GLORIA MARK, University of California at Irvine, USA
KRISTY MILLAND, University of Toronto, Canada
EDWARD LANK, University of Waterloo, Canada
EDITH LAW, University of Waterloo, Canada

Crowdworkers regularly support their work with scripts, extensions, and software to enhance their productivity. Despite their evident significance, little is understood regarding how these tools affect crowdworkers' quality of life and work. In this study, we report findings from an interview study (N=21) aimed at exploring the tooling practices used by full-time crowdworkers on Amazon Mechanical Turk. Our interview data suggests that the tooling utilized by crowdworkers (1) strongly contributes to the fragmentation of microwork by enabling task switching and multitasking behavior; (2) promotes the fragmentation of crowdworkers' work-life boundaries by relying on tooling that encourages a 'work-anywhere' attitude; and (3) aids the fragmentation of social ties within worker communities through limited tooling access. Our findings have implications for building systems that unify crowdworkers' work practice in support of their productivity and well-being.

CCS Concepts: • **Human-centered computing** → **Computer supported cooperative work**.

Additional Key Words and Phrases: Crowdwork; Tooling; Work Practice; Interview; Fragmentation.

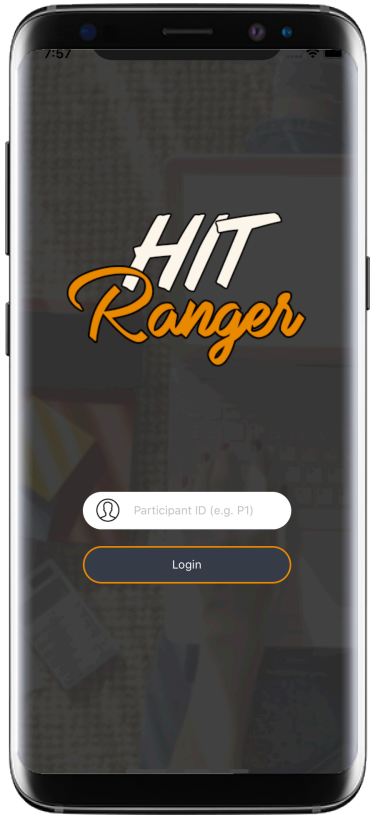
ACM Reference Format:
Alex C. Williams, Gloria Mark, Kristy Milland, Edward Lank, and Edith Law. 2019. The Perpetual Work Life of Crowdworkers: How Tooling Practices Increase Fragmentation in Crowdwork. *Proc. ACM Hum.-Comput. Interact.* 3, CSCW, Article 24 (November 2019), 28 pages. https://doi.org/10.1145/3359126

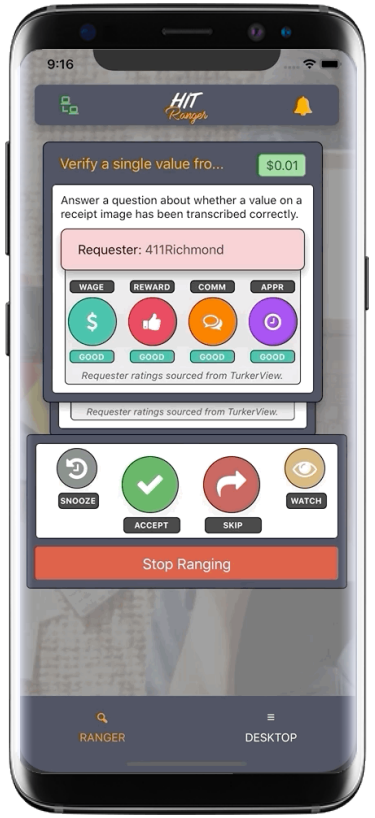
1 INTRODUCTION

Over the past decade, crowdwork has risen as an established and thriving work practice for thousands of people across the globe [20]. In a 2010 survey of US-based Amazon Mechanical Turk (MTurk) workers, less than 5% of the participants identified as spending 40 hours or more on crowdwork [42]. Recent data-driven analyses of long-term worker activity on MTurk found that the population of crowdworkers on the platform is generally stable, but that "tens of thousands of new workers that arrive on the platform each year" to replace workers that abandon the profession [21, 61]. Beyond MTurk, a 2016 survey found that more than 5 million individuals in the UK are actively engaged in crowdwork [39], and 53% of a 2017 survey's respondents with crowdworkers in Switzerland reported the profession as their full-time job [40]. Spurred by the changing nature

Authors' addresses: Alex C. Williams, University of Waterloo, Canada, alex.williams@uwaterloo.ca; Gloria Mark, University of California at Irvine, USA, gmark@uci.edu; Kristy Milland, University of Toronto, Canada, kristymilland@gmail.com; Edward Lank, University of Waterloo, Canada, lank@uwaterloo.ca; Edith Law, University of Waterloo, Canada, edith.law@uwaterloo.ca. *Disclaimer:* Kristy Milland was previously a crowdworker and a community manager of TurkerNation. She did not contribute to data collection or analysis of this research.

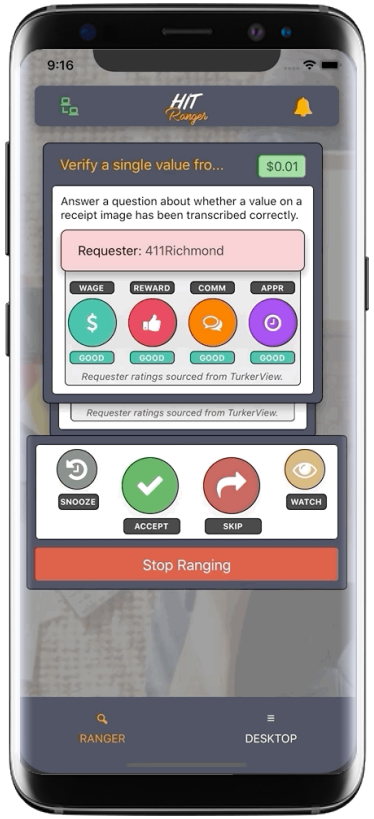
2573-0142/2019/11-ART24 $15.00
https://doi.org/10.1145/3359126





HIT Ranger (*CHI '21)*

A Future of Intelligence

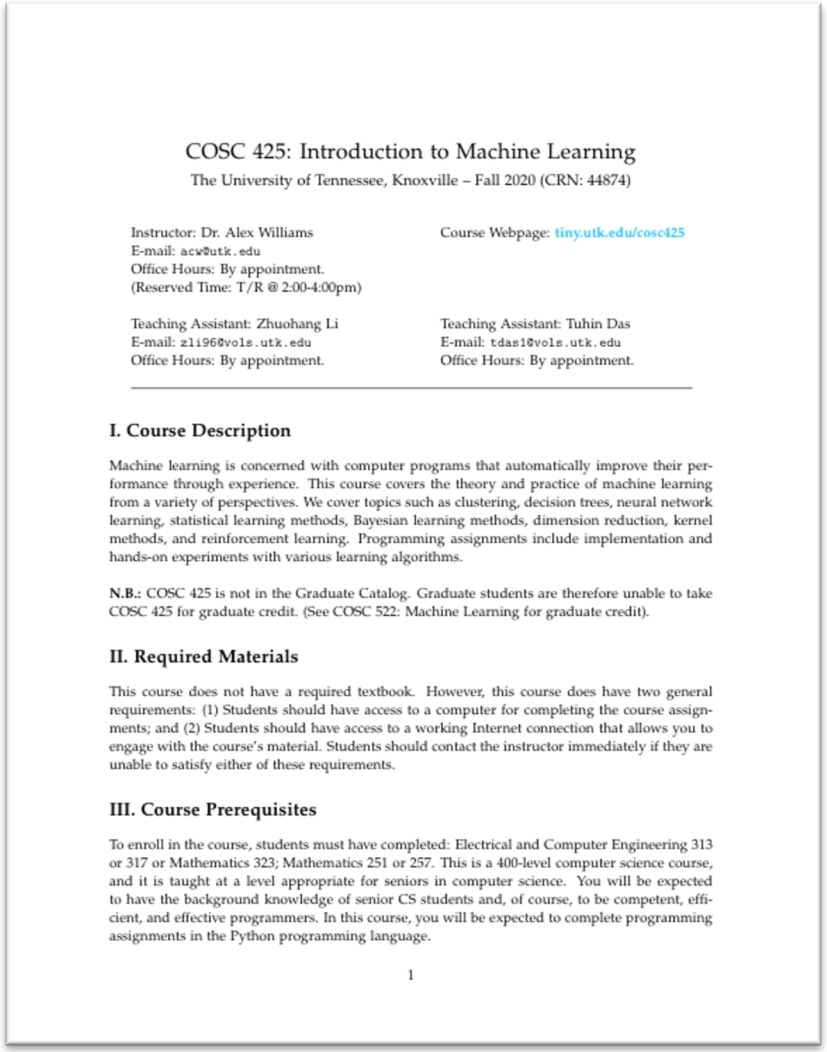


1. How can SwitchBot disengage proactively?
   → Sensing the user getting in a vehicle.

2. How can Mercury route a relevant microtask?
   → Understanding IDE context at the desktop.

3. How can HIT Ranger identify "happy" work?
   → Knowing work that makes the user happy.

Solution: Collect data → Learn trends.



COSC 425: Introduction to Machine Learning

The University of Tennessee, Knoxville – Fall 2020 (CRN: 44874)

Instructor: Dr. Alex Williams
E-mail: acw@utk.edu
Office Hours: By appointment.
(Reserved Time: T/R @ 2:00-4:00pm)

Course Webpage: tiny.utk.edu/cosc425

Teaching Assistant: Zhuohang Li
E-mail: zli96@vols.utk.edu
Office Hours: By appointment.

Teaching Assistant: Tuhin Das
E-mail: tdas1@vols.utk.edu
Office Hours: By appointment.

I. Course Description

Machine learning is concerned with computer programs that automatically improve their performance through experience. This course covers the theory and practice of machine learning from a variety of perspectives. We cover topics such as clustering, decision trees, neural network learning, statistical learning methods, Bayesian learning methods, dimension reduction, kernel methods, and reinforcement learning. Programming assignments include implementation and hands-on experiments with various learning algorithms.

N.B.: COSC 425 is not in the Graduate Catalog. Graduate students are therefore unable to take COSC 425 for graduate credit. (See COSC 522: Machine Learning for graduate credit).

II. Required Materials

This course does not have a required textbook. However, this course does have two general requirements: (1) Students should have access to a computer for completing the course assignments; and (2) Students should have access to a working Internet connection that allows you to engage with the course's material. Students should contact the instructor immediately if they are unable to satisfy either of these requirements.

III. Course Prerequisites

To enroll in the course, students must have completed: Electrical and Computer Engineering 313 or 317 or Mathematics 323; Mathematics 251 or 257. This is a 400-level computer science course, and it is taught at a level appropriate for seniors in computer science. You will be expected to have the background knowledge of senior CS students and, of course, to be competent, efficient, and effective programmers. In this course, you will be expected to complete programming assignments in the Python programming language.

1

Course Website + Syllabus

https://tiny.utk.edu/cosc425

Course Objectives

This course has three main objectives:

1. to develop an understanding of the fundamental concepts of modern machine learning;

2. to provide first-hand experience with implementing a breadth of learning algorithms; and

3. to empower students with the ability to evaluate, deploy, and critique learning algorithms.

Course Structure

Lectures

→ Asynchronous unless specified otherwise.
→ Available at 12:01am each “class day”.

Office Hours

→ Reserved Time: 2:00 – 4:00pm T/R)
→ Book online via Course Website.

Course Structure

50% Programming Assignments

→ Up to 5 assignments related to implementation.

→ Must use Python.

20% Homework

→ Up to 6 assignments related to ML articles.

→ Written responses, brief questions, etc.

30% Final Team Project

→ Create a machine learning solution for a "real" problem.

Final report → Purpose, Approach, Evaluation, Limits

Course Topics

- Decision Trees
- Instance-based Learning
- Perceptron
- Linear Regression
- Probabilistic Modelling
- Linear Methods of Classification
- Support Vector Machines
- Performance Evaluation
- Feature Construction / Selection
- Ensemble Learning
- Interpretable Machine Learning
- Active Learning
- Semi-Supervised Learning
- Unsupervised Learning
- Neural Networks
- Deep Learning
- Adversarial Generative Models
- Structured Prediction
- Reinforcement Learning
- Human-Centered Machine Learning
- The Future of Machine Learning

Course Scope

Biologically-Inspired Computation

Introduction to Machine Learning

Computational Cognitive Neuroscience

Machine Learning

Spring 2021

Special Topics:
Human-AI Interaction

What is “Interaction”?

→ Where HCI meets AI.

Designing Intelligent Interactivity.

→ Challenges and opportunities.

Evaluating Intelligent Interactivity.

→ User studies, walkthroughs, etc.

Course Policies

Communication Policy *

→ E-mail + Canvas

Attendance Policy *

→ No policy. (You are responsible adults!)

Assignment Submission Policy *

→ Hand-in via Canvas by 11:59pm on the due date.

Late Submission Policy*

→ 1 Day Late = 10%; 2 Days Late = 20% , etc.

*See Syllabus for details.

Course Policies

Academic Integrity & Plagiarism *

→ See Hilltopics Student Handbook.

→ Do not plagiarize, copy, etc.

Civility Policy *

→ I reserve the right to dismiss you from Zoom.

Accommodations for Disabilities*

→ Contact Office of Disability Services

→ (865-974-6087)

*See Syllabus for details.

Course Policies

Accommodations for Change Related to COVID-19

→ **Personal Change:** Consult with me immediately.*

→ **University Change**: I will try to keep you informed.

Other concerns (e.g. "absences"): Consult the Dean of Students (https://dos.utk.edu).

*See Syllabus for details.

Next Time

What is
Machine Learning?

Friday, 2:15pm. @
https://tennessee.zoom.us/j/6294683300