

Pramod Kotipalli // Portfolio



I'm Pramod, a fourth-year Computer Science at the [Georgia Institute of Technology](#).

I research *wearable computing* and *augmented reality* under [Professor Thad Starner](#).

I've developed full-stack experiences across web, mobile, and wearable platforms.

I have significant professional experience, most recently at **SPACEX** 

Table of Contents

RF-Pick : Awarded Best Paper, ISWC 2018	3
// Novel wearable RFID verification system for logistical order picking	
dARts	4
// Play darts in augmented reality	
RichCaptions	5
// Symbolic math captions for educational videos	
Twitter Sentiment Analysis	6
// Symbolic math captions for educational videos	
Safely	8
// Smart wearable technology to end campus violence	
Modeling and Analysis of BCG and ECG Signals	10
// Work with Professor Jim Rehg and UC San Francisco	
Algorithmic Implementations - Constraint Satisfaction	11
// Follow up from Georgia Tech's A.I. course for undergraduates	
Algorithmic Implementations - Traveling Salesperson	12
// Python Package created to support Contextual Computing Group	

RF-Pick : Awarded Best Paper, ISWC 2018

// Novel wearable RFID verification system for logistical order picking

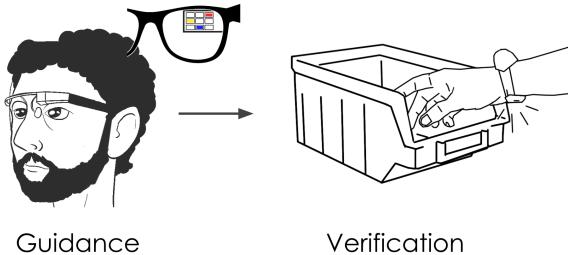
ISWC 2018: 24% Paper Acceptance Rate

One of 62 Accepted Papers to be **Awarded Best Paper**

ISWC 2018
8-12 October 2018 SINGAPORE

Abstract:

- Order picking accounts for 55% of the annual \$60 billion spent on warehouse operations in the United States.
- Reducing human-induced errors in the order fulfillment process can save warehouses and distributors significant costs.
- We investigate a radio-frequency identification (RFID)-based verification method wherein wearable RFID scanners, worn on the wrists, scan passive RFID tags mounted on an item's bin as the item is picked; this method is used in conjunction with a head-up display (HUD) to guide the user to the correct item.
- We compare this RFID verification method to pick-to-light with button verification, pick-to-paper with barcode verification, and pick-to-paper with no verification.
- We find that pick-to-HUD with RFID verification enables significantly faster picking, provides the lowest error rate, and provides the lowest task workload.



Guidance

Verification

Contributions:

- Integration of novel wearable system with HUDs
- Comprehensive and rigorous HCI user study
- Embedded systems and networking programming
- Application of software engineering principles to increase development speed and system robustness

Advisor:

Professor Thad Starner: Technical Lead, Google Glass



View through Google Glass



Georgia Tech School of Interactive Computing
College of Computing

dARts

// Play darts in augmented reality

Created an iOS application, in one weekend, to have fun in the virtual space!

Uses ARKit to overlay SceneKit objects in the virtual world.

Allows user to select detected walls to place a dartboard.

User then taps on dartboard to throw virtual "darts" at the virtual dartboard.



RichCaptions

// Symbolic math captions for educational videos

Problem identified

- Online education is rapidly gaining momentum
- Video captioning systems are limited to displaying simple plain-text
- Math/science students learn better by reading semantically-useful symbols

Solution created

- Create web application where content creators can easily caption their videos in LaTeX
- Allows anyone on the internet to watch these captioned videos without cost

Illustrative example

The screenshot shows a video player interface. The main content area displays a handwritten mathematical derivation for integration by parts. The formula $\int f(x)g'(x) dx = f(x)g(x) - \int f'(x)g(x) dx$ is written in green ink. Below it, the integral $\int x^2 e^x dx$ is shown with x^2 circled in blue and e^x circled in pink. A cursor points to the right side of the equation. At the bottom of the video frame, a caption in white text on a black background reads: "f of x is equal to x squared, in which case f prime of x". The video player has a progress bar at 1:20 / 6:45, volume controls, and a closed captioning icon.

so we are saying $f(x) = x^2$ in which case $f'(x) = 2x$

Design paradigms & technologies used

- REST API design and docs with Django REST framework
- Material Design front-end with AngularJS + Javascript/jQuery
- YouTube <iframe> API
- LaTeX, KaTeX



Twitter Sentiment Analysis

// Symbolic math captions for educational videos

Motivation

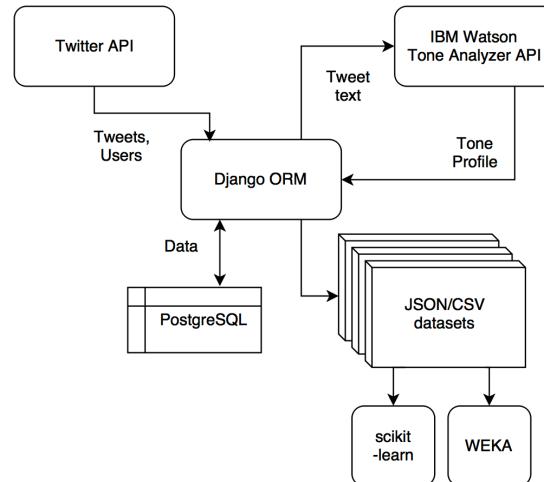
- The United States, in particular, features extremely high costs for healthcare
- Public awareness and support mental health care is increasing

Solution created

- Pipeline to gather tweets on two polar topics to understand users' sentiment towards them
- Demonstration that identifies those users who use depression-indicative language
- Useful to mental health professionals to identify long-term trends in user's mental health

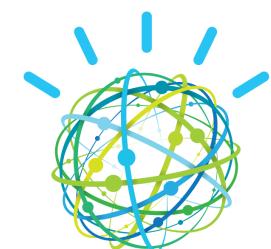
Pipeline overview

1. Data collection: Collected nearly 4,000 tweets from the Twitter Developer API and labelled them based on hashtags present. For example, tweets containing "depressed" (or related hashtags) will be labelled as belonging to the "depressive-indicative" class; tweets containing "happy" (or related hashtags) will be labelled as part of the "non-depressive-indicative" class.
2. Understand the user's position in the Twitter community: Call the Twitter API to gain information about the user's followers, followees, average retweet counts, and more.
3. Data analysis: Send each of the 4,000 tweets through IBM Watson's Tone Analyzer API to gain more dimensions of sentiment information about each tweet.
4. Classification model: Use the labelled data to discriminate between tweets that are "depressive-indicative" or not in terms of their language characteristics. Trained classification model with scikit-learn's k-Nearest Neighbors implementation.
5. Classify an unknown user: Given an unknown user, generate visualizations and an overall classification of their Twitter tweet language.



Technologies used

- Python + Django web framework
- scikit-learn & IBM Watson intelligence APIs
- chart.js & Material Bootstrap



Twitter Sentiment Analysis (cont.)

Visualizations

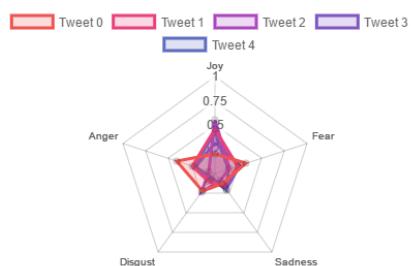
Analyzing tweets for [@barackobama](#)

Profile overview

Followers: 77510571
Followees: 634144
Tweets/day: 4.3833
Average favorites: 0.0500
Average retweets: 1878.0950

IBM Watson Tone Analyzer

Emotional tone



Tweet-focused analytics @barackobama

Senate leaders have blocked Judge Garland's hearing for over six months now. Keep pushing: <https://t.co/peV6aCYswa>
#DoYourJob

Word count: 70 Retweets: 625 Favorites: #1 @0

Depressive: No

Standard Depressive



Senate leaders need to work for the good of the American people, not seek out political points. <https://t.co/Tdx3xk6gYi>

Word count: 72 Retweets: 1753 Favorites: #0 @0

Depressive: No

Standard Depressive



Overall classification

From his/her language and status within the Twitter community, **@barackobama** is likely Ambiguous (56.57)

On a scale of 0 (indicative) to +100% (standard) for this particular class with an 85% accuracy.

Safely

// Smart wearable technology to end campus violence



Problem identified:

- Almost one in four women and one in 20 men experience sexual assault in college
- Students are often in situations where a smartphone is not accessible for help

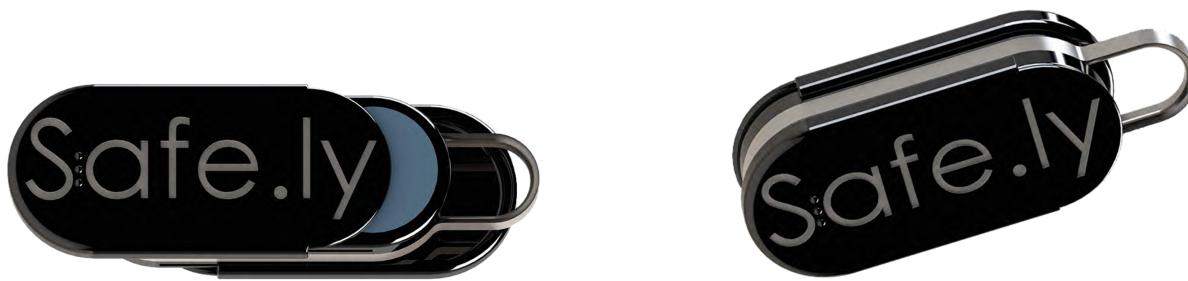
Solution created:

- Discrete, wearable button programmed to alert friends, family members, & campus police
- Complimentary smartphone app that manages student's notification prefs. & tracking
- Comprehensive business and financial plan for producing physical products

Achievements:

- Won competition of dozens of teams & six semifinalists, presented to venture capitalists
- Identified an underserved market need in need for a real & permanent solution
- Co-founded a startup ([GoSafely, LLC](#)) that has now raised over \$20,000 in funding

Hardware renderings:



- Users slide open and press the shielded blue button. Pressing the button once alerts friends and family. Pressing the button twice alerts local police through Wi-Fi or a cellular network.
- The device can be added to a keychain for discrete and easy access in times of emergency.

Mobile app:



- Users can manage their Safely tags from their mobile phones via Bluetooth.
- Users can also trigger a more specific safety warning (e.g. EMS) from their mobile app. After an event is triggered, the Safely web API is notified and interested parties can track a user's location in real-time.

Technologies used:

- Ionic Framework
- AngularJS in TypeScript + HTML/CSS
- SolidWorks CAD + 3D Printing

Modeling and Analysis of BCG and ECG Signals

// Work with Professor Jim Rehg and UC San Francisco

Problem identified:

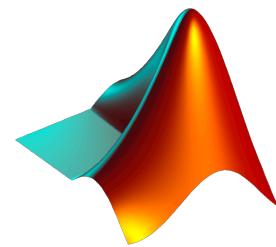
- Increased adoption of wearable heart-rate sensors opens opportunity to understand mobile health behavior
- Classification of arrhythmias and other heart complications saves lives and costs

Solution created:

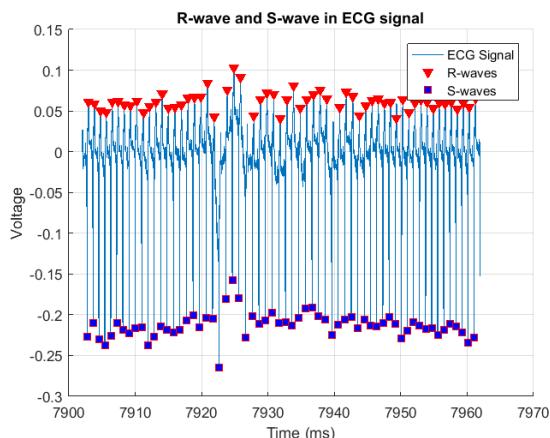
- Pipeline to input, clean, de-trend, process, and fit ballistocardiogram and electrocardiogram waveforms to Hidden Markov Models

Technologies used:

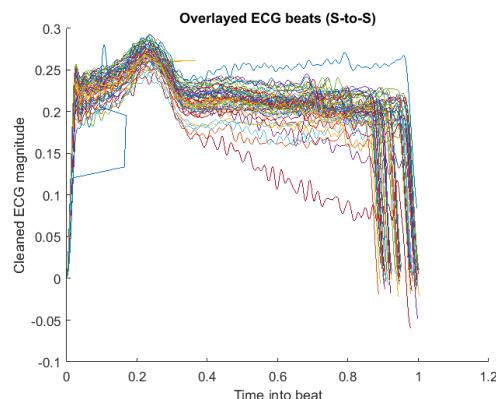
- MATLAB + DSP Toolbox
- Python hmmlearn



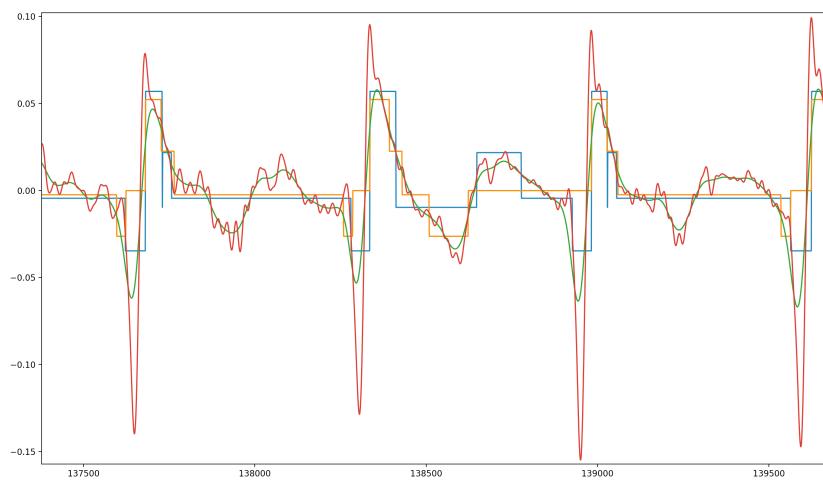
Visualizations:



De-trended signals and labelled R-wave and S-wave



Overlaying S-to-S ECG complexes



HMM applied to processed ECG signals

Algorithmic Implementations - Constraint Satisfaction

// Follow up from Georgia Tech's A.I. course for undergraduates

Efficient C# implementation of CSP available publicly at <https://github.com/p13i/CSP>.

Features extremely-readable, well-documented, and professional C# code.

Includes comprehensive unit test suite and continuous integration with Travis CI: [build passing](#)

Features sample usage of library solving Sudoku puzzles, including this "expert" puzzle:

.	4		
.	5	2	.		
.	.	.	.	6	.	8	1	3		
----- ----- -----			----- ----- -----			----- ----- -----				
.	7	.	.	2	6	.	.	.		
.	.	4	.	.	5	3	.	7	8	.
3	2	.	.	.	8	
----- ----- -----			----- ----- -----			----- ----- -----				
6	.	5	.	.	7	.	.	.	1	
.	2	3	.		
4	

Before ...

2	5	3	.	8	1	7	.	9	6	4
8	1	6	.	4	3	9	.	5	2	7
9	4	7	.	2	6	5	.	8	1	3
----- ----- -----			----- ----- -----			----- ----- -----				
5	7	8	.	1	2	6	.	3	4	9
1	6	4	.	9	5	3	.	7	8	2
3	2	9	.	7	4	8	.	1	5	6
----- ----- -----			----- ----- -----			----- ----- -----				
6	8	5	.	3	7	2	.	4	9	1
7	9	1	.	6	8	4	.	2	3	5
4	3	2	.	5	9	1	.	6	7	8

... after

```
- Running Expert 1...
|-- Finished test in expert-1 after 249433 steps in 2646 milli-seconds.
|-- PASS Expert 1
```

Algorithmic Implementations - Traveling Salesperson

// Python Package created to support Contextual Computing Group

Created globally-available gt-tsp package: <https://pypi.org/project/gt-tsp/>

Features extremely-readable, well-documented, and professional Python code.

Includes comprehensive unit test suite and continuous integration with Travis CI: [build passing](#)

Employed TSP solver to [generate pick paths](#) for novel sparse order-picking study, visualizing the shortest pick paths inside a model warehouse (dense graph) environment.

