Text Classification via Supervised Machine Learning for an Issue Tracking System

Marty Kandes, Ph.D.

HPC User Services Group
San Diego Supercomputer Center
University of California, San Diego

HEPiX Spring 2019
Monday, March 25th, 2019
2:00 - 2:25PM PT

About Me

- Applied math, computational physics, HPC
- Distributed High-Throughput Computing Group @ SDSC

  OSG / CMS GlideinWMS Factory Operations
  OSG Software Team - Software Dev & Testing

- HPC User Services Group @ SDSC

A not-so long time ago in a data center not that far, far away ...

- In 2012, 99% of all computational jobs run on NSF-funded HPC resources utilized fewer than 2048 CPU-cores, while accounting for approximately 50% of the total core-hours consumed across these resources.
- Nearly 70% of all jobs actually ran on only a single compute node (16 CPU-cores) or less.

Comet: A Supercomputer Built to Serve the 99%

Design Goals for Comet

- **Flexability** - Ability to run a wide range of scientific and engineering applications to support complex, dynamic and multidisciplinary computational workflows.
- **Scalability** - Ability to support a large, diverse community of modestly-sized research projects that in aggregate represent a significant amount of research activity.

Comet By the Numbers

- **1944 compute nodes**: Dual-socket; 2.5 GHz Intel Xeon E5-2680v3 processors; 12 cores per processor; 128 GB DDR4 DRAM; 120 GB/s memory bandwidth; 320 GB SSD (210 GB Avail)
- **4 large-shared memory nodes**: Quad-socket; 2.2 GHz Intel Xeon E7-8860v3 processors; 16 cores per processor; 1.5 TB DDR4 DRAM; 400 GB SSD (260 GB Avail)
- **36 k80 gpu nodes**: Same as standard *compute* node, but with 2 PCIe-based NVIDIA Tesla K80 dual-GPU accelerators per node
- **36 p100 gpu nodes**: Dual-socket; 2.4 GHz Intel Xeon E5-2680v4 processors; 14 cores per processor; 128 GB DDR4 DRAM; 150 GB/s memory bandwidth; 400 GB SSD (260 GB Avail); 4 PCIe-based NVIDIA Tesla P100 GPU accelerators per node

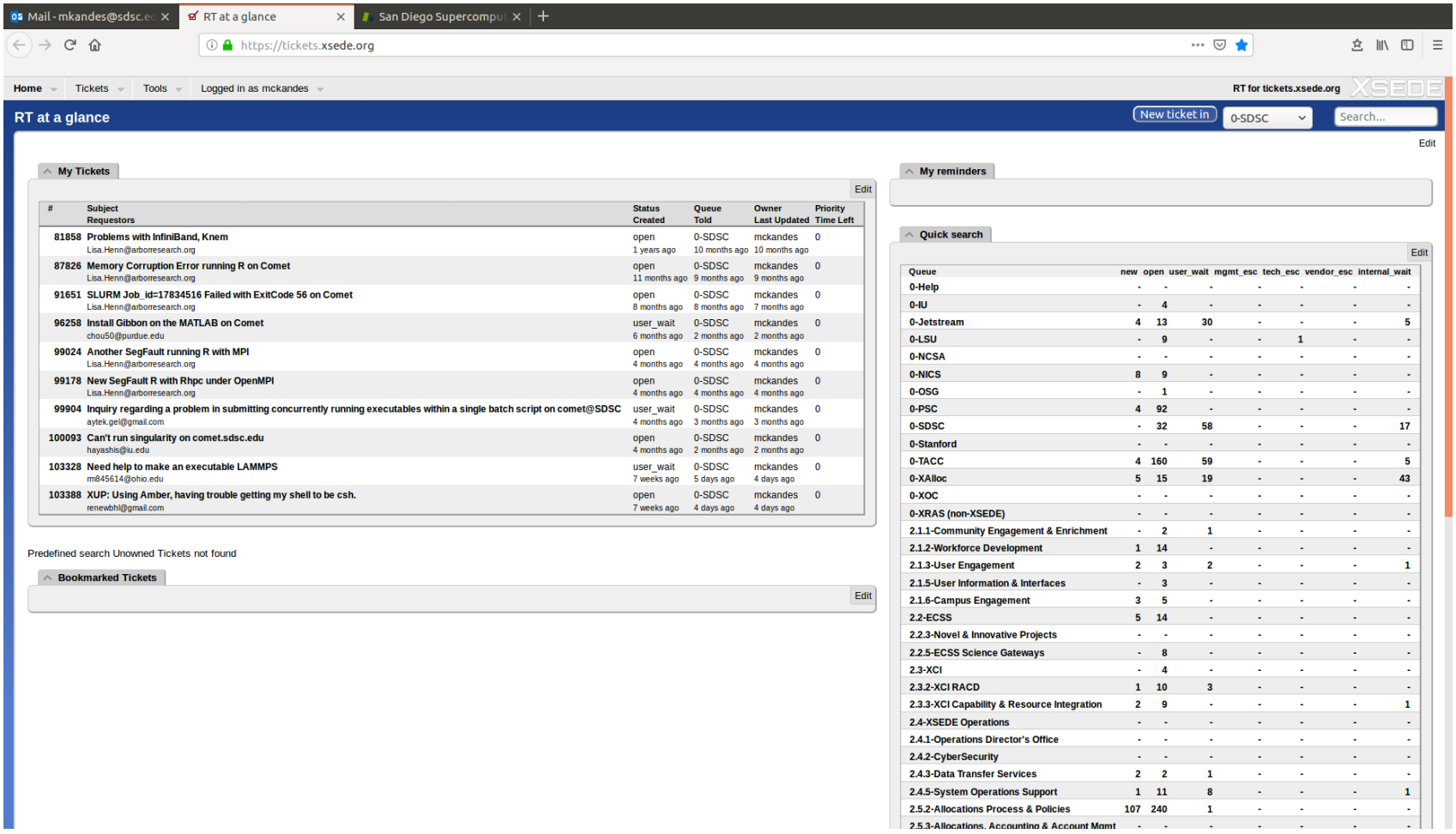
2.76 Pflop/s

Comet By the Numbers

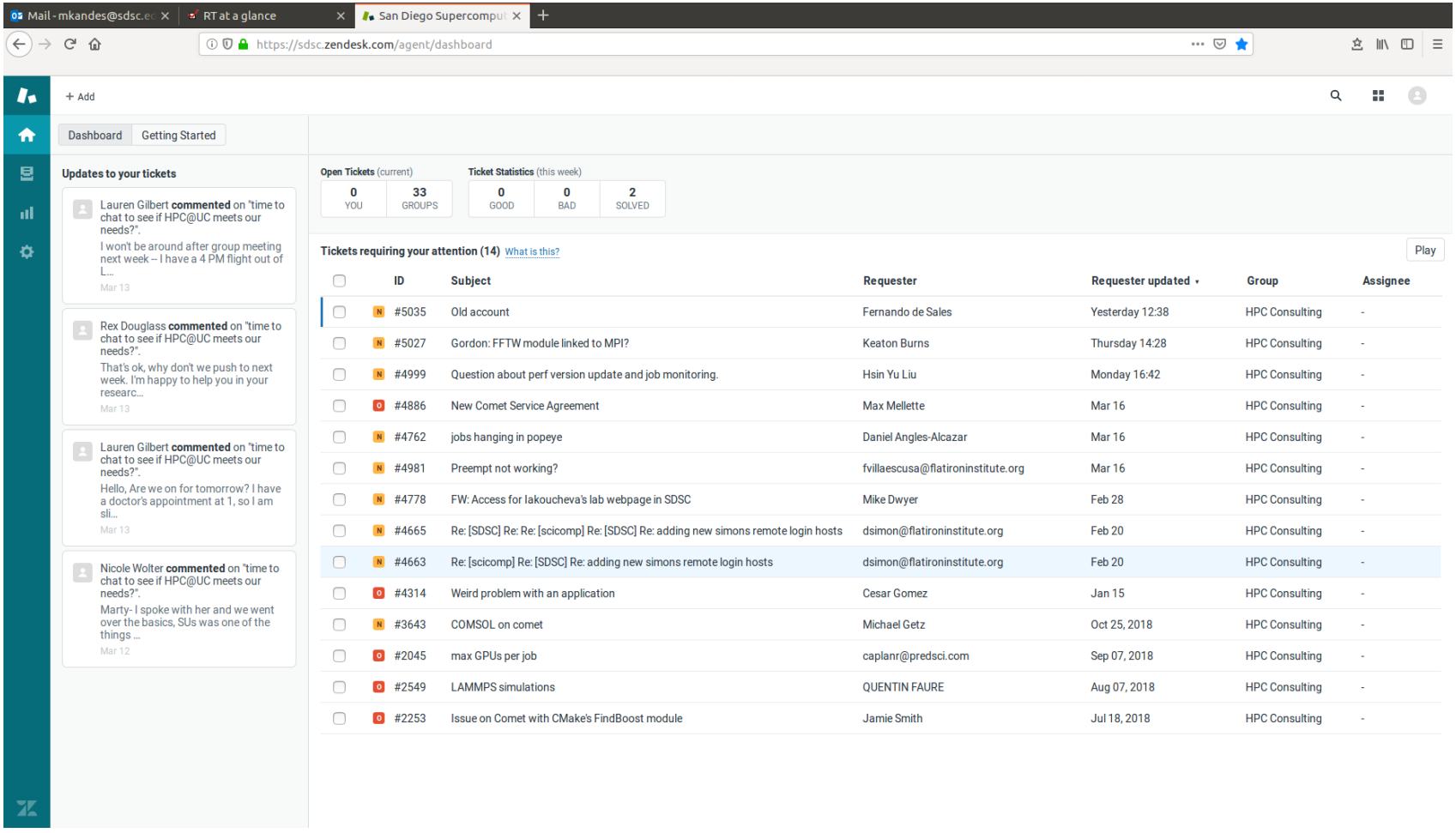
- **Interconnect**: Mellanox FDR (56Gbps) InfiniBand; hybrid fat-tree topology; rack-level (72 node) full bisection bandwidth; 4:1 oversubscription cross-rack bandwidth
- **Storage**: NSF-based $HOME storage (100 GB per user); 6.4 PB 200 GB/s Lustre-based parallel filesystem storage (intermediate-term use: at least 500 GB per group allocation in /oasis/projects short-term use: up to 10 TB per user in /oasis/scratch)
- **Applications**: More than 173 software applications and libraries maintained and deployed via Rocks (Linux) cluster distribution; accessible to users via software modules; span a wide range of scientific diciplines, including, but not limited to, bioinformatics, chemistry, data analytics, engineering, fluid dynamics, mathematics, molecular dynamics, neuroscience, and statistics
- **Scientific Impact**: 1755 PIs; 358 institutions; 1144 research allocations; 4709 direct-access users; 33000+ gateway users; 997 publications

My Daily Grind: User Support Tickets

User Support Tickets via XSEDE RT

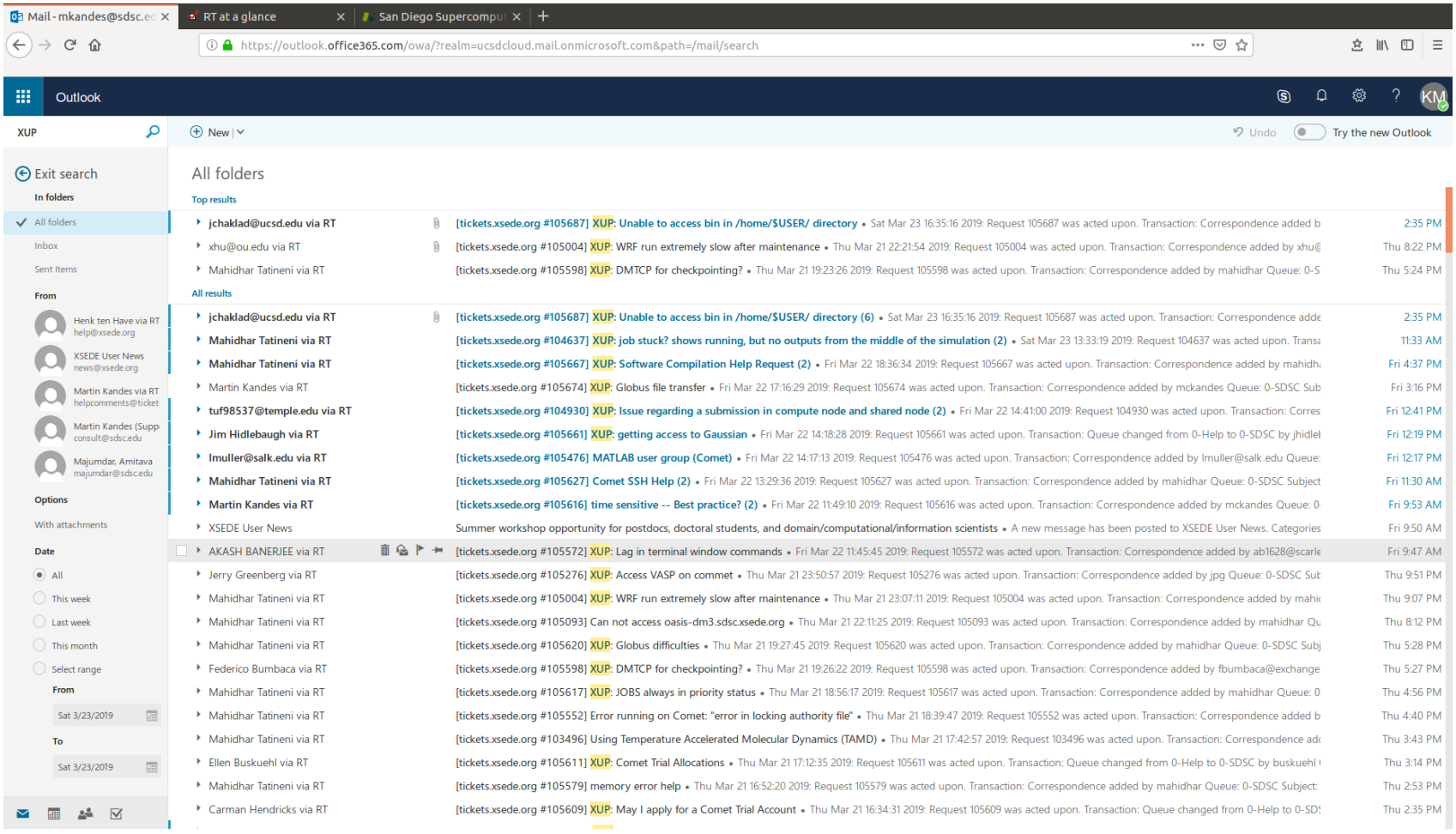


User Support Tickets via Zendesk



ID	Subject	Requester	Requester updated	Group	Assignee
#5035	Old account	Fernando de Sales	Yesterday 12:38	HPC Consulting	-
#5027	Gordon: FFTW module linked to MPI?	Keaton Burns	Thursday 14:28	HPC Consulting	-
#4999	Question about perf version update and job monitoring.	Hsin Yu Liu	Monday 16:42	HPC Consulting	-
#4886	New Comet Service Agreement	Max Mellette	Mar 16	HPC Consulting	-
#4762	jobs hanging in popeye	Daniel Angles-Alcazar	Mar 16	HPC Consulting	-
#4981	Preempt not working?	fvillaescusa@flatironinstitute.org	Mar 16	HPC Consulting	-
#4778	FW: Access for Iakoucheva's lab webpage in SDSC	Mike Dwyer	Feb 28	HPC Consulting	-
#4665	Re: [SDSC] Re: Re: [scicomp] Re: [SDSC] Re: adding new simons remote login hosts	dsimon@flatironinstitute.org	Feb 20	HPC Consulting	-
#4663	Re: [scicomp] Re: [SDSC] Re: adding new simons remote login hosts	dsimon@flatironinstitute.org	Feb 20	HPC Consulting	-
#4314	Weird problem with an application	Cesar Gomez	Jan 15	HPC Consulting	-
#3643	COMSOL on comet	Michael Getz	Oct 25, 2018	HPC Consulting	-
#2045	max GPUs per job	caplanr@predsci.com	Sep 07, 2018	HPC Consulting	-
#2549	LAMMPS simulations	QUENTIN FAURE	Aug 07, 2018	HPC Consulting	-
#2253	Issue on Comet with CMake's FindBoost module	Jamie Smith	Jul 18, 2018	HPC Consulting	-

User Support Tickets via Outlook



User Support Tickets by Example: Matlab Group

" I am receiving and error message trying to run a matlab code. Can you help me find a solution?"

User Support Tickets by Example: Gaussian Group

"I want to use gaussian09 on comet. I include the following in my batchscript:

module load gaussian/09.D.01
/opt/gaussian/09.D.01/g09 < scoE_small_opt.com >
scoE_small_opt.log

But i get an error saying permission denied. Could you please guide me on how to use gaussian09 for my calculation?"

User Support Tickets by Example: VASP Licencing

"Our group has a VASP licence with number: 5-321. Could you please give me access to VASP on Comet?"

User Support Tickets by Example: No $HOME Directory

"When I gsissh to comet using SSO Hub I get an error "Could not chdir to home directory /home/user008: No such file or directory". I'm unable to work because of this. Please let me know how this issue can be resolved."

User Support Tickets by Example: `git` old SSL libs

"I am writing to ask about an ssl error that occurs when I try to clone git repositories from github into my comet account. The error and context are below:

git clone https://github.com/words-sdsc/modular_deep_learning.git
Cloning into 'modular_deep_learning'...
fatal: unable to access
'https://github.com/words-sdsc/modular_deep_learning.git': SSL connect error

Do I need to load some specific module to fix the error?"

User Support Tickets by Example: Slurm Down

"I am trying to submit a job using sbatch and it does not work apparently. It gives me the following:

sbatch batch_job.sb
sbatch: error: slurm_receive_msg: Socket timed out on send/recv operation
sbatch: error: Batch job submission failed: Socket timed out on send/recv operation

squeue is not working either. Do you know if anything is wrong over there?"

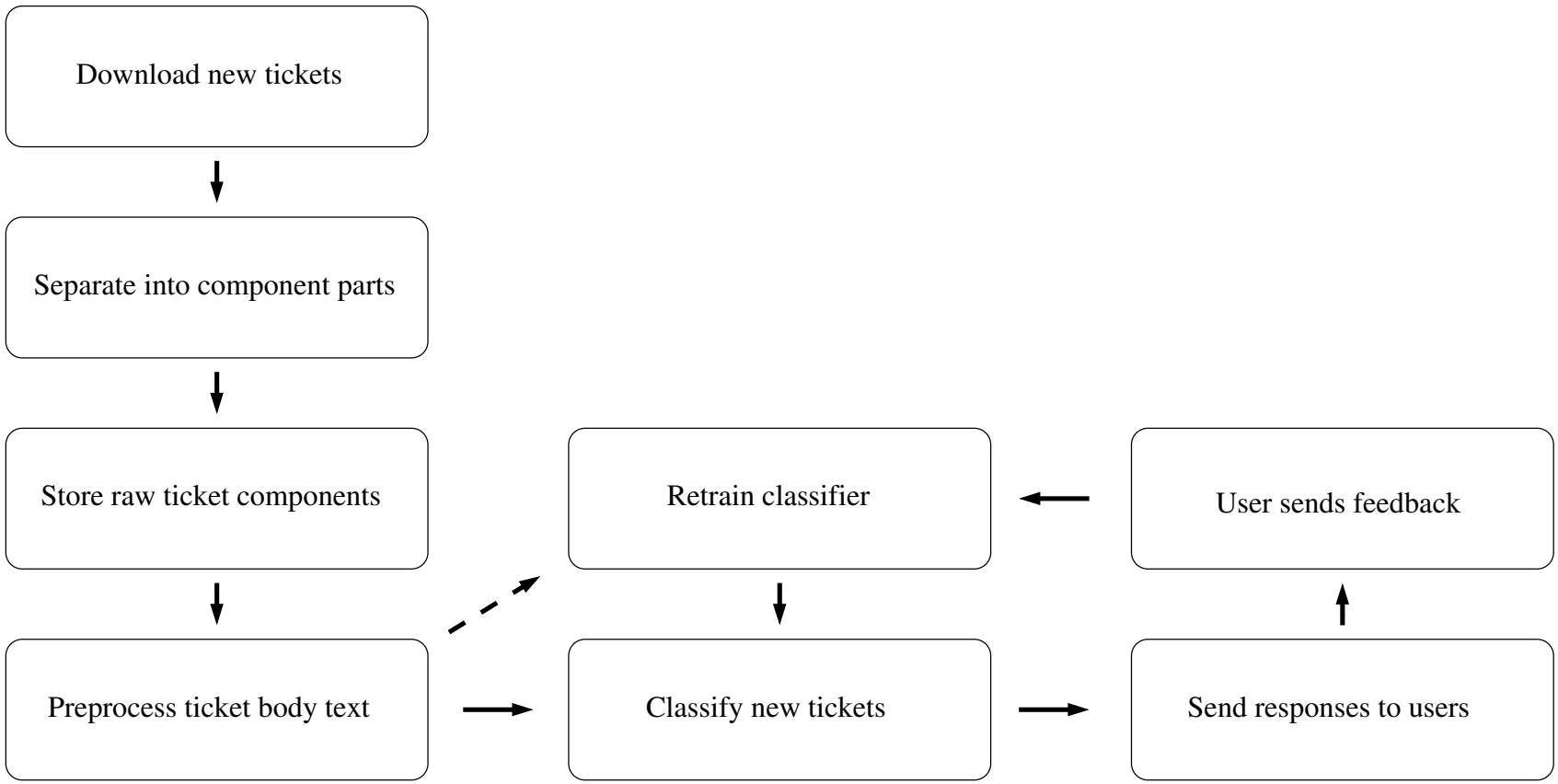
The Fundamental Question(s)

- What if we could utilize machine learning algorithms to classify and answer common user questions as they arrive?
- In an efficient classification scheme can be developed, can we provide useful, generic automated email responses to users to help them resolve their issues more quickly?
- How much time would such a system save the us?

Project Objective(s)

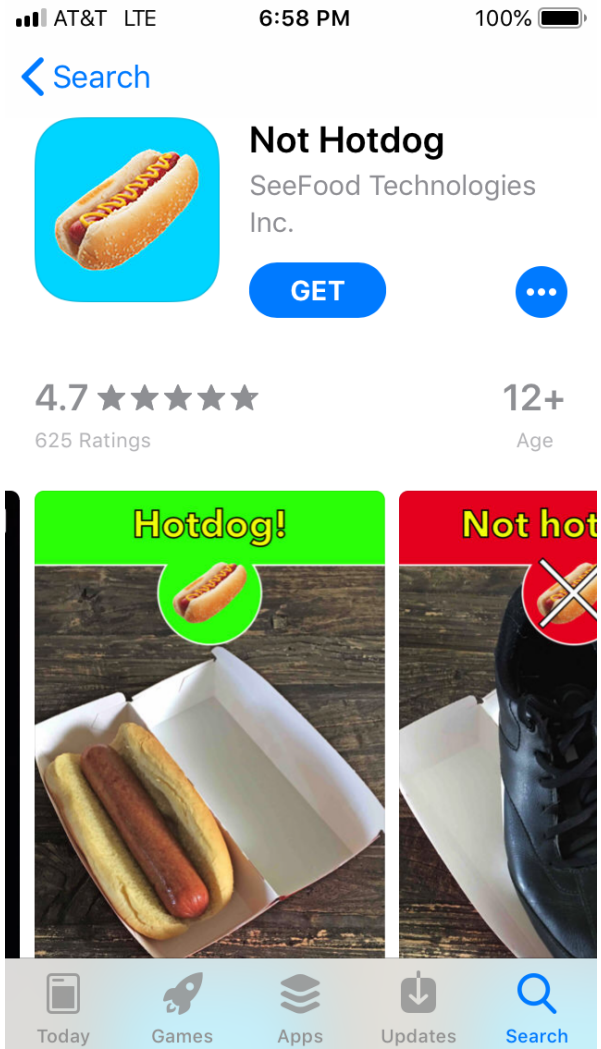
- Utilize supervised machine learning algorithms to classify common user support tickets with greater than 90% accuracy.
- For classifiable tickets, provide automated email responses to users (during non-business hours) within minutes of arrival.
- Focus on only initial ticket responses.
- Incorporate user feedback mechanism to help improve training dataset labels over time.

Classification and Response Workflow



Naive Bayes Classifer

- Conditional probability model for text categorization
- First introduced and studied in early 1960s
- Classifies documents as belonging to one category or another based on word frequencies
- Assumes feature independence for each given class
- Advantage: Only requires small amount of training data



Naive Bayes Classifer

For each word frequency vector $\mathbf{x} = \{x_1, x_2, \ldots, x_N\}$ representing a document to be classified, compute the conditional probability that the document belongs to category c_k

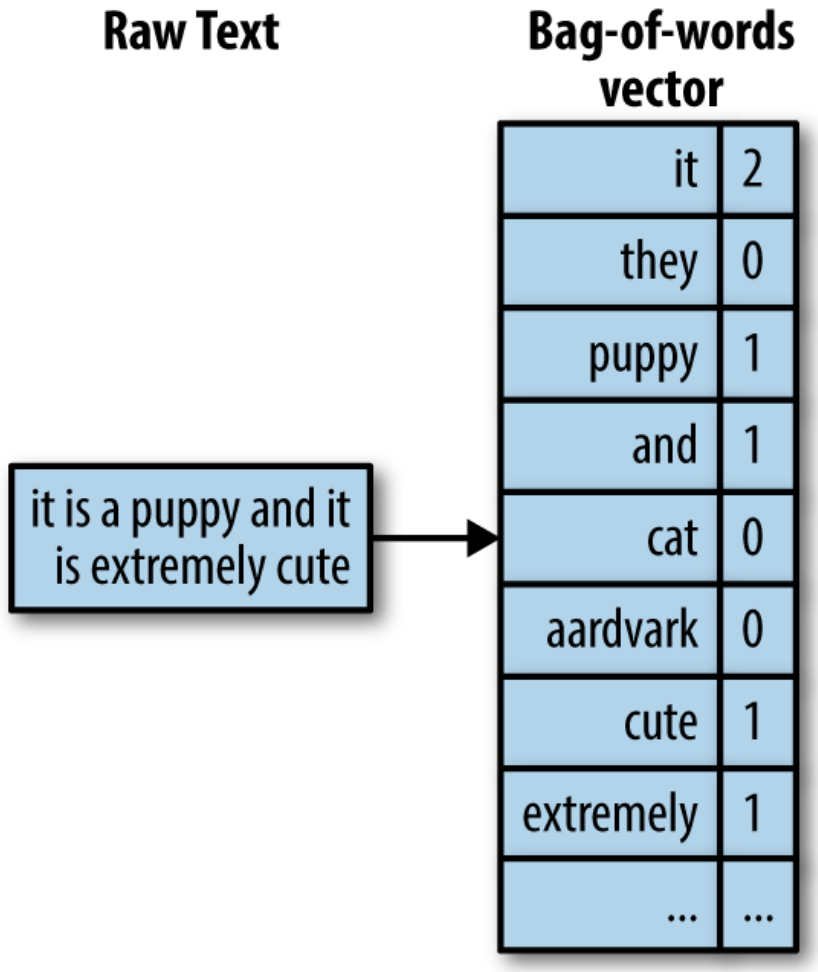
$$p(c_k|\mathbf{x}) = \frac{p(c_k)p(\mathbf{x}|c_k)}{p(\mathbf{x})}$$

for each of the K possible outcomes or classes $\mathbf{c} = \{c_1, c_2, \ldots, c_K\}$.

Assign document to category c_k when $p(c_k|\mathbf{x})$ is the highest probability in the set and above some minimum acceptable cutoff.

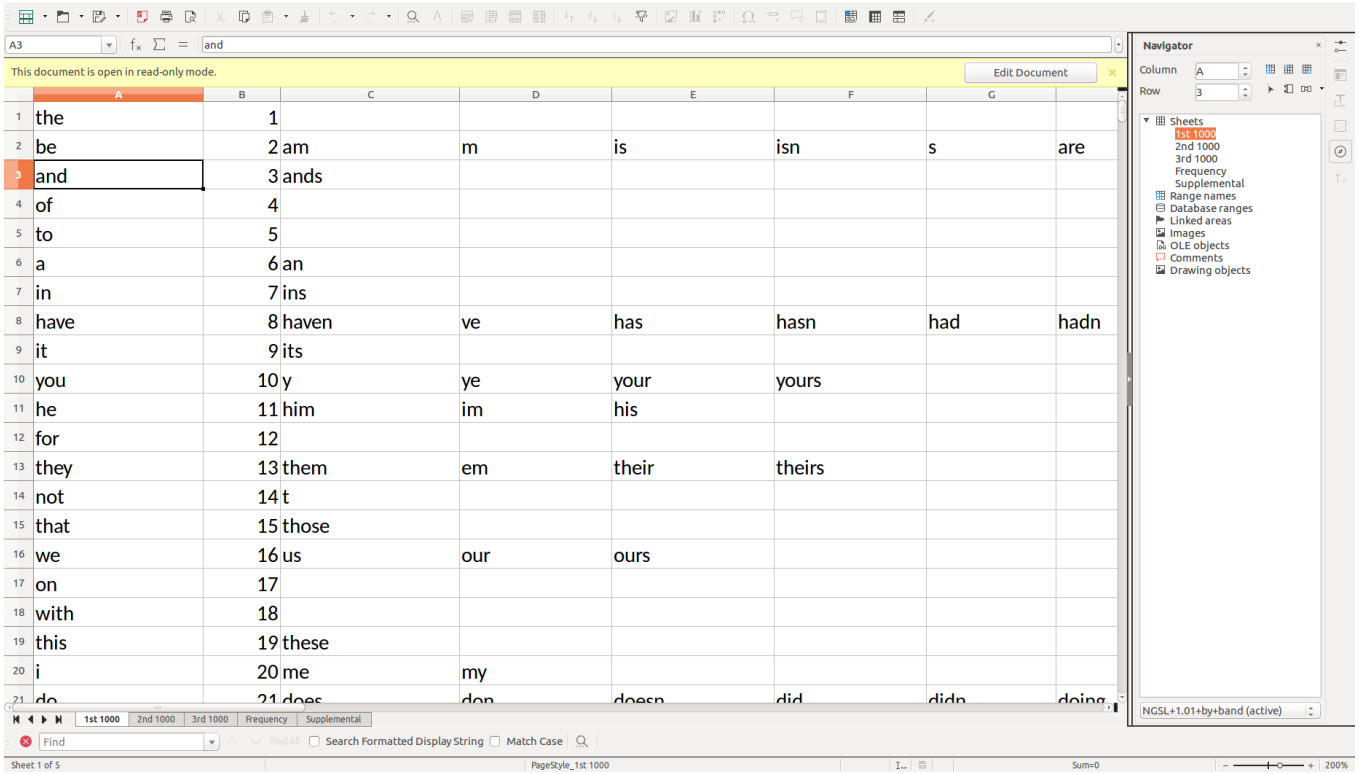
Bag-of-Words Model

- Simple representation for natural languange processing
- Any text is represented as a multiset of its words
- Count frequency of occurence of each word in a text
- Ignore grammar and word order



Blacklisting

Remove most commonly used words

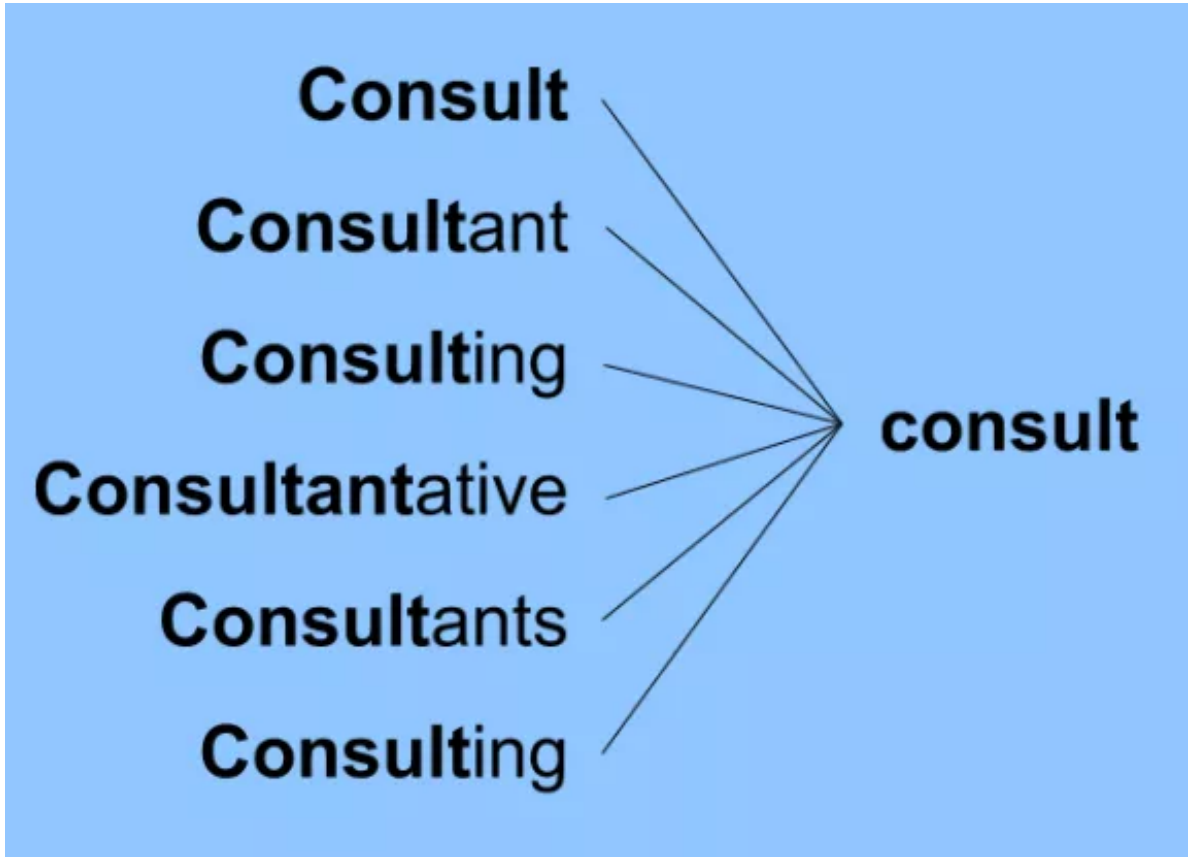


1	the	1					
2	be	2	am	m	is	isn	s	are
3	and	3	ands					
4	of	4					
5	to	5					
6	a	6	an					
7	in	7	ins					
8	have	8	haven	ve	has	hasn	had	hadn
9	it	9	its					
10	you	10	y	ye	your	yours		
11	he	11	him	im	his			
12	for	12						
13	they	13	them	em	their	theirs		
14	not	14	t					
15	that	15	those					
16	we	16	us	our	ours			
17	on	17						
18	with	18						
19	this	19	these					
20	i	20	me	my				
21	do	21	does	don	doesn	did	didn	doing

New General Service List is a list of approximately 2,800 of the most important high frequency words in the English language

Stemming

Process of removing all modifiers of a keyword including prefix, suffix, and pluralization until only the root of the word remains.



Porter Stemming Algorithm: Removes suffixes

Prototyping: Summer 2018

1 of 1 150%

Text Classification via Supervised Machine Learning for an Issue Tracking System

SDSC SAN DIEGO SUPERCOMPUTER CENTER

N. Clark[1], R. A. Martin[2], D. H. Wu[3], and M. C. Kandes[4]

UC San Diego

[1] *La Jolla Country Day School* [2] *Canyon Crest Academy* [3] *Pacific Ridge High School*
[4] *San Diego Supercomputer Center*

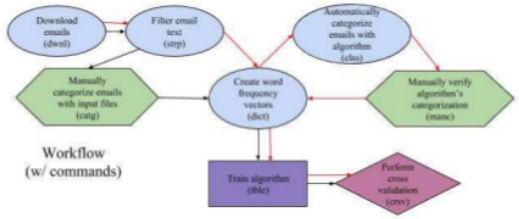
Abstract

The objective of our project was to use a text classification algorithm to categorize support ticket emails into common issues. We implemented our project in python, employing regular expressions and the imap library to download emails. After building a classified database of emails, we researched the process of building and training a text classification algorithm, focusing on the Naive Bayes text classification algorithm, a simple but powerful probabilistic algorithm that makes the naive assumption that words in the sentence are independent from one another. Gradually, we developed our own version of Naive Bayes and a scheme for collecting and processing email data and finally compared other implementations of text classification algorithms to our own. Here, we discuss our process and initial results.

Introduction

For each scientist that innovates in his or her respective field with the help of SDSCs supercomputers, someone has to help navigate him/her through the often confusing and error-prone world of software development. Each day, users encounter technical issues and send support ticket emails, and its up to a few help desk employees to point them in the right direction. Their work is often repetitive as users frequently run into common issues. We studied different text classification and python techniques involving regular expressions, stemming, and file organization, among other things. Our goal was to conduct research on text classification, so that we could design a machine learning algorithm to categorize support tickets pertaining to the 20 most common issues encountered by SDSC users and relieve user services of handling simple errors.

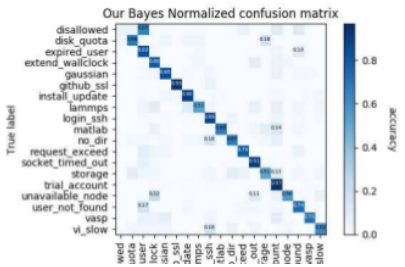
Methods



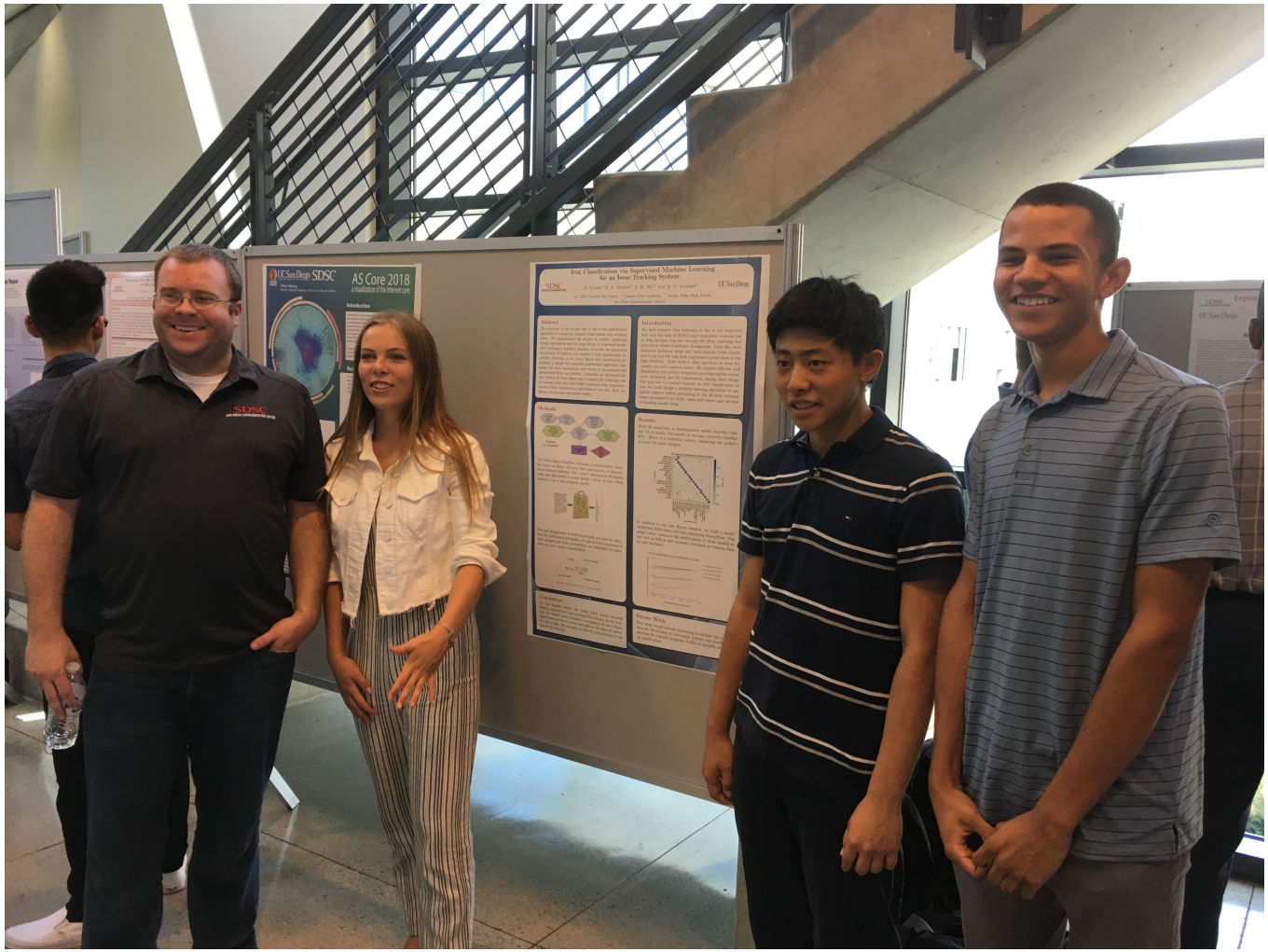
The Naive Bayes Classifier technique is probablilistic classifier based on Bayes' theorem with assumptions of independence between features. The "naive" assumption disregards order and placement of words within a body of text—often referred to as a 'bag of words' model.

Results

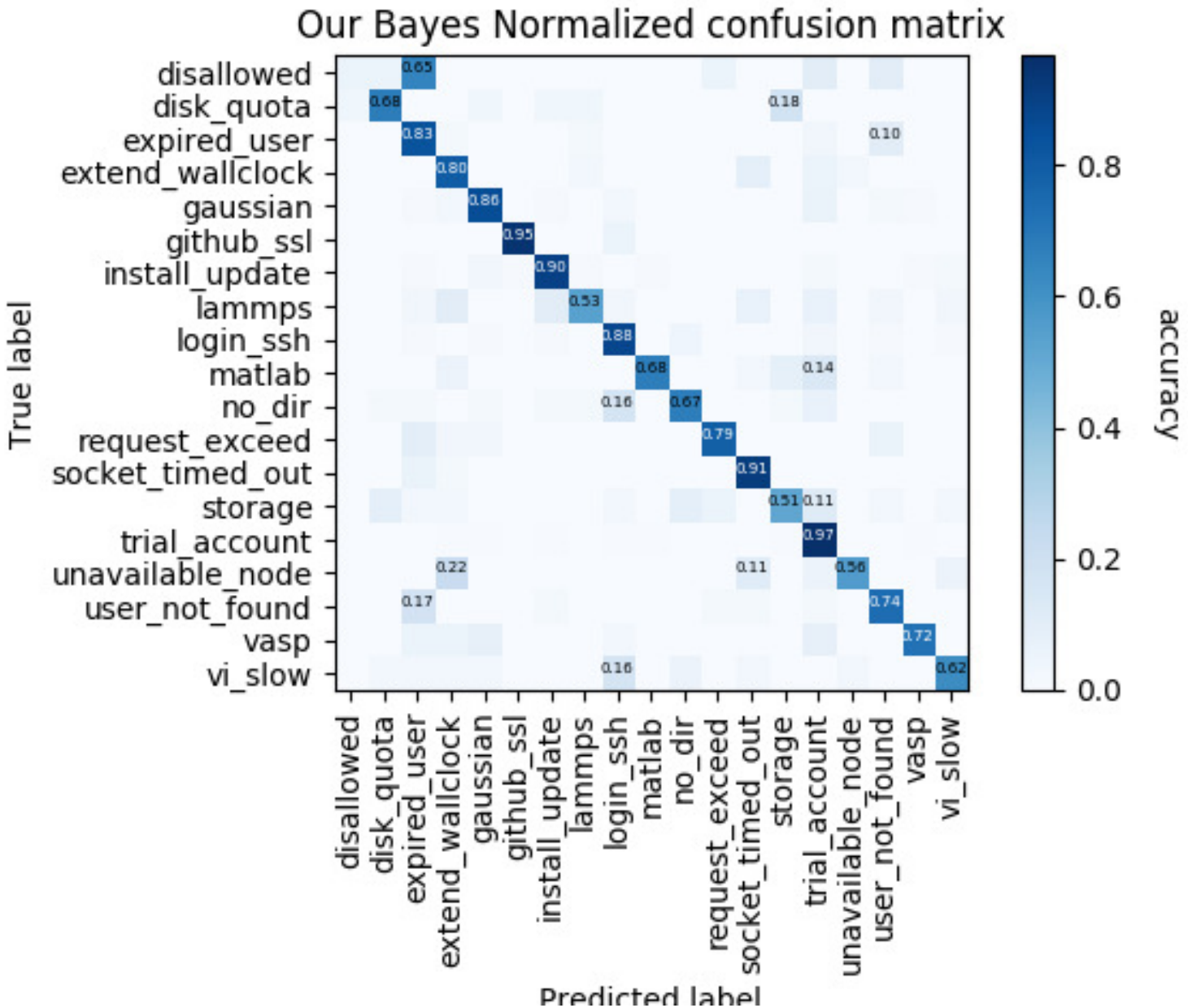
With 20 categories, a randomization would correctly classify 5% of emails. Our model, on average, correctly classifies 80%. Below is a confusion matrix, displaying the model's accuracy for each category.



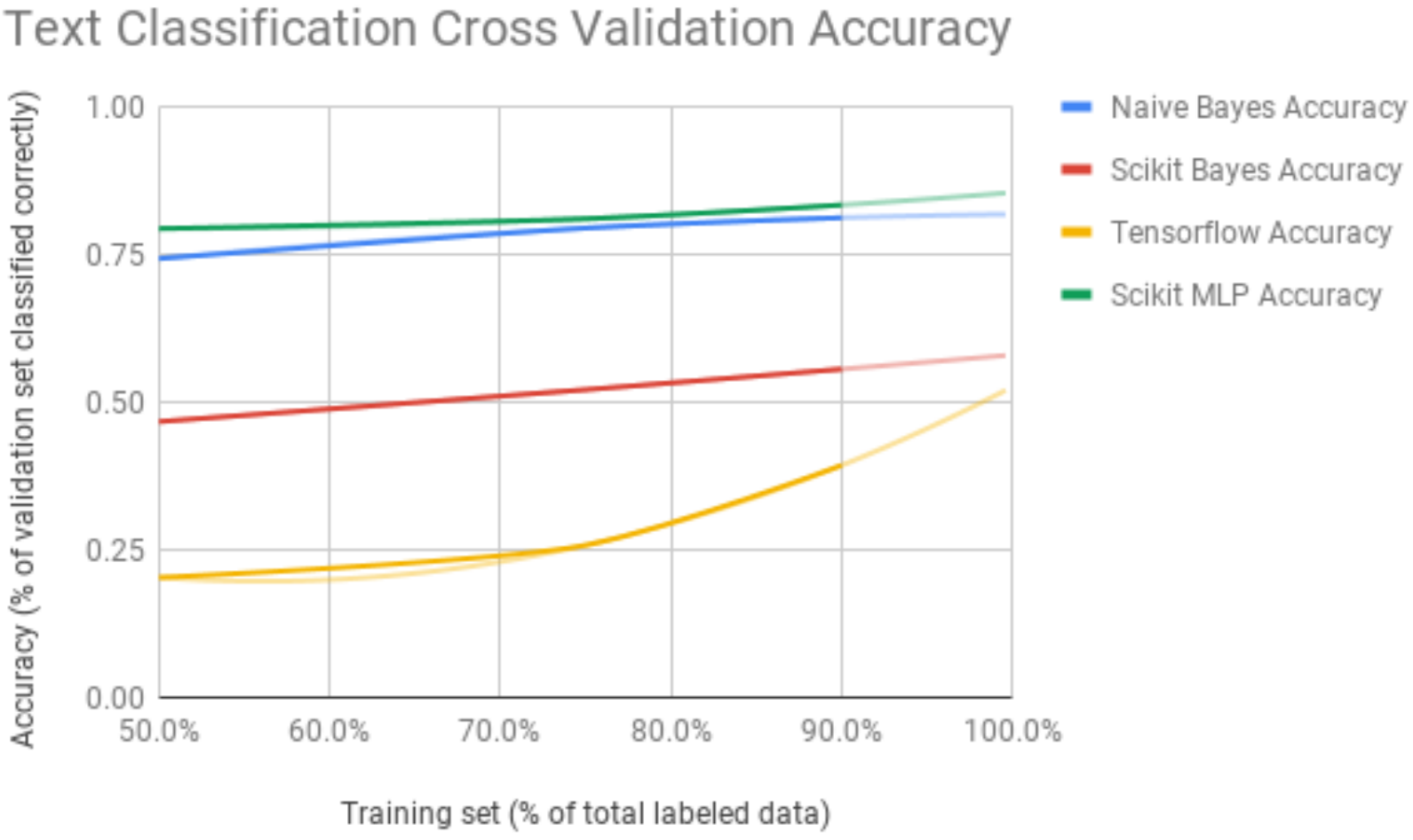
Research Experience for High School Students



Preliminary Results



Preliminary Results



Current Status and Future Work

- Still in prototyping phase; working towards minimum viable prototype for production test
- Developing sqlite3 database schema; plan to complete and implement prior to REHS 2019
- Developing generic python class to help simplify sqlite3 database management
- Continue to label and increase raw training dataset size
- Integrate XOAuth2 support with Outlook Mail API
- Explore multilayer perceptron (MLP) and support vector machine (SVM) approaches more completely
- Explore stemming algorithms; e.g., algorithms for prefix and pluralization removal
- Explore text segmentation algorithms; e.g., in-body metadata removal and other personally identifiable information

Questions?