Tale of training a *Web Terminator!*

Bharadwaj Machiraju
@tunnelshade_

Aka ML - What works in offensive appsec and what doesn't!

ME?

I am Bharadwaj Machiraju

Appsec Engineer at Yodlee Infotech.

Project Leader for owasp owtf.

~~Loves machine learning~~.

Are you still reading this?

It seems so! @tunnelshade_

Please keep in mind!

- All the work was done in my free time over a course of time!
- If you think a problem can be approached in a better manner, let us share the knowledge over ~~coffee~~ beer.
- Have some more ideas, please write it down so we can discuss.
- There are multiple ways of approaching a problem.
- I also share my failures so that you know what not to do.

*All Images to original owners

Our way to cool stuff!!

1

Objectives

Convince you that ML is not rocket science.

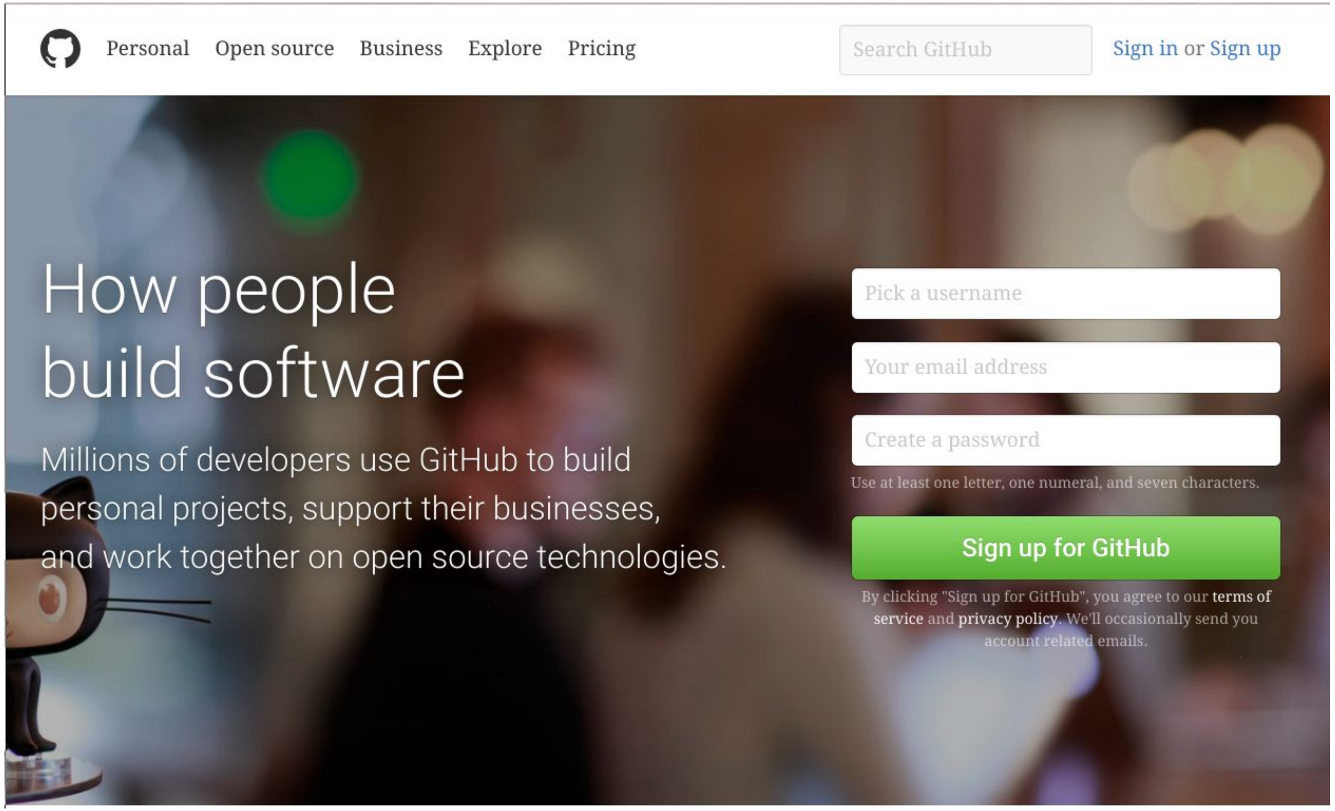
ML in appsec is interesting.

What real ML looks like!

Maybe convince you to work in this direction!

Once upon a time, imagine you were hunting bugs in this cool new site called github!

Imagine you are visiting github for the first time!



Personal Open source Business Explore Pricing

Search GitHub

Sign in or Sign up

How people build software

Millions of developers use GitHub to build personal projects, support their businesses, and work together on open source technologies.

Pick a username

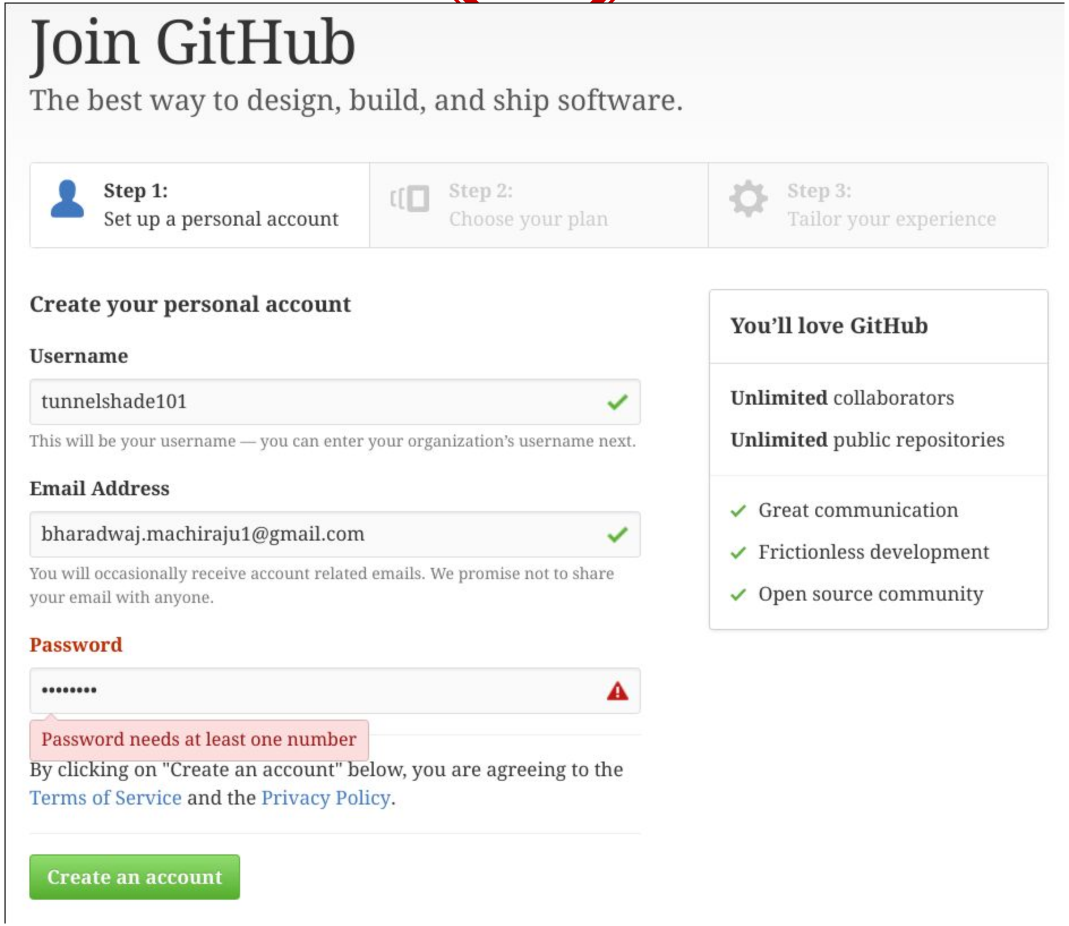
Your email address

Create a password

Use at least one letter, one numeral, and seven characters.

Sign up for GitHub

By clicking "Sign up for GitHub", you agree to our terms of service and privacy policy. We'll occasionally send you account related emails.



Join GitHub

The best way to design, build, and ship software.

Step 1: Set up a personal account

Step 2: Choose your plan

Step 3: Tailor your experience

Create your personal account

Username

tunnelshade101 ✓

This will be your username — you can enter your organization's username next.

Email Address

bharadwaj.machiraju1@gmail.com ✓

You will occasionally receive account related emails. We promise not to share your email with anyone.

Password

••••••••• ⚠

Password needs at least one number

By clicking on "Create an account" below, you are agreeing to the Terms of Service and the Privacy Policy.

Create an account

You'll love GitHub

Unlimited collaborators

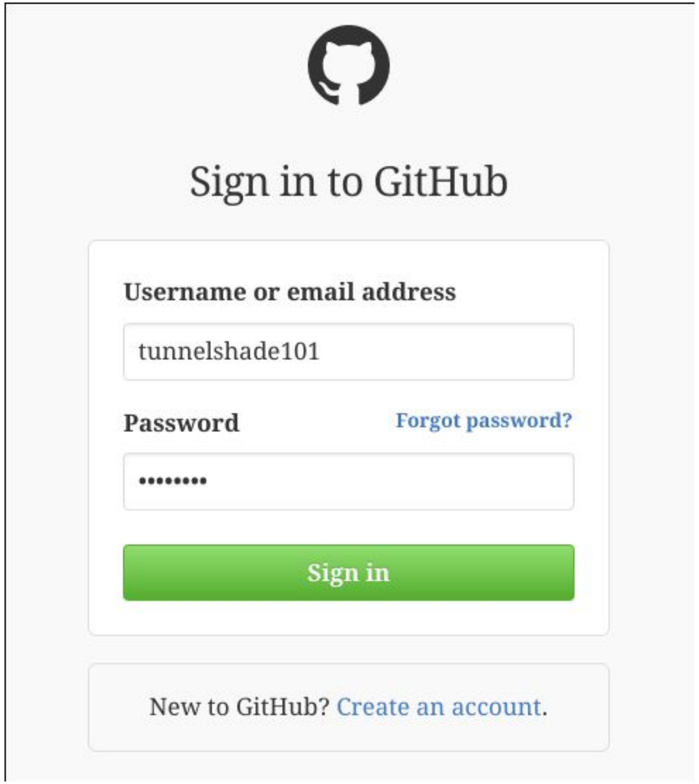
Unlimited public repositories

✓ Great communication

✓ Frictionless development

✓ Open source community

Now, you are open source rockstar! (i.e second visit onwards)



Sign in to GitHub

Username or email address

tunnelshade101

Password

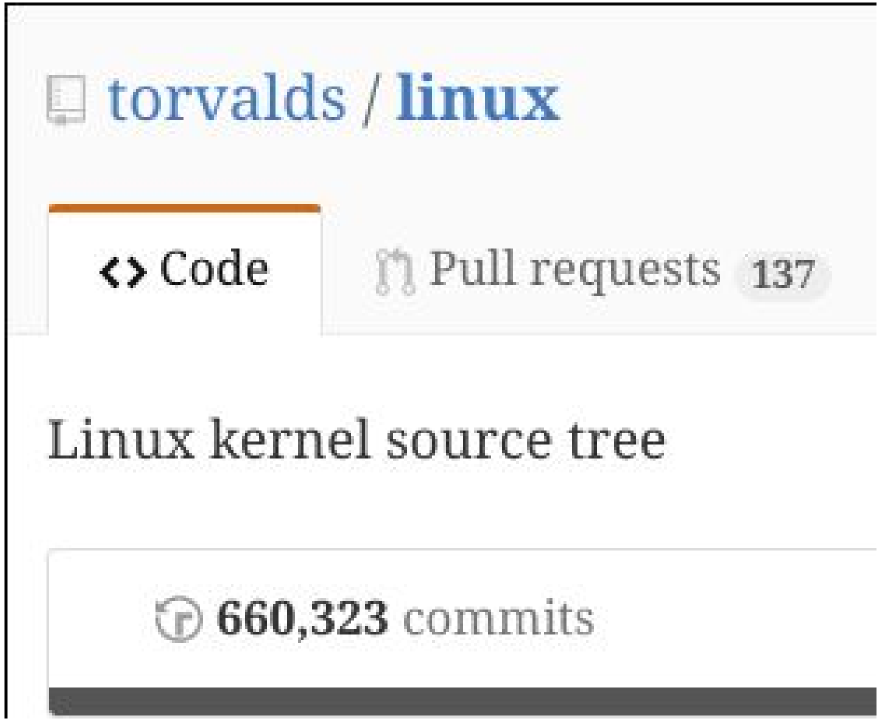
Forgot password?

••••••••

Sign in

New to GitHub? Create an account.

Browsing the functionality you end up on this page!


torvalds / linux

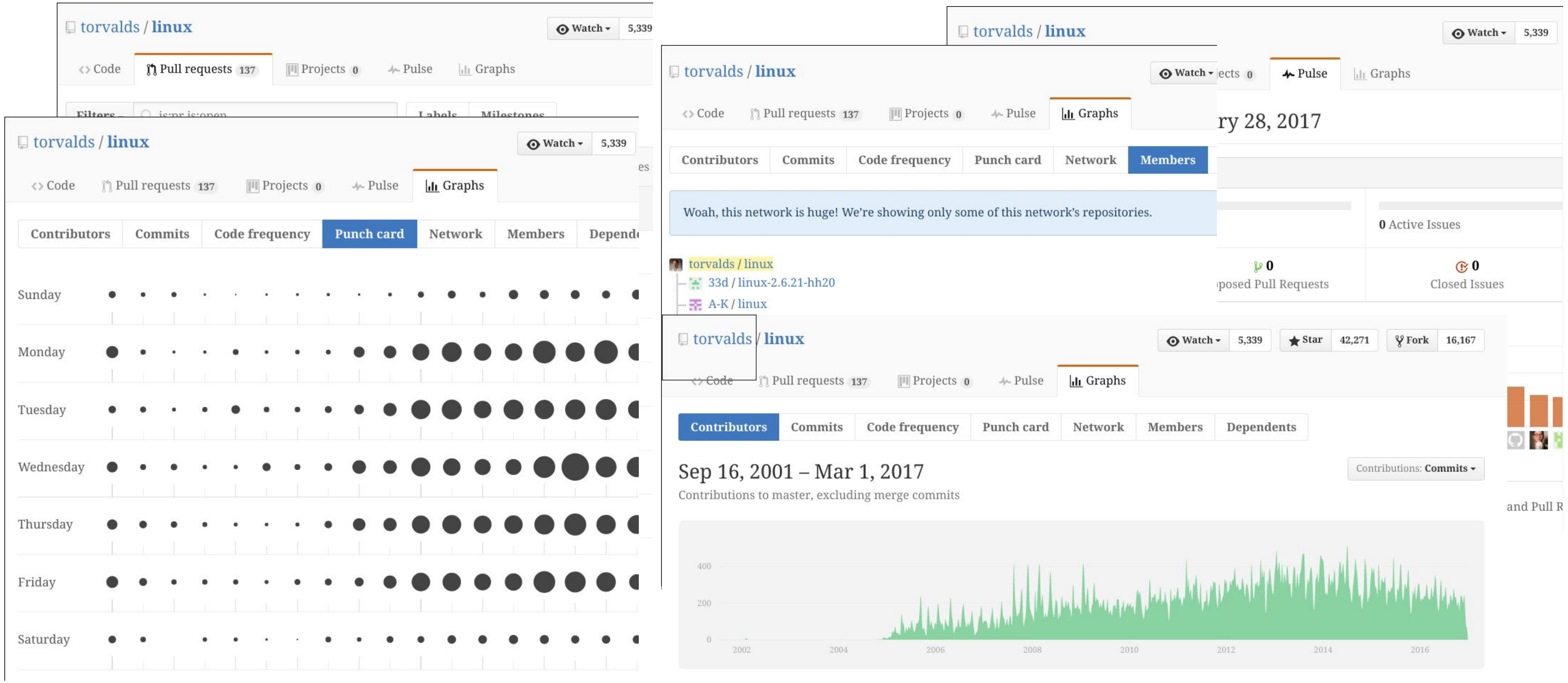
Code

Pull requests 137

Linux kernel source tree

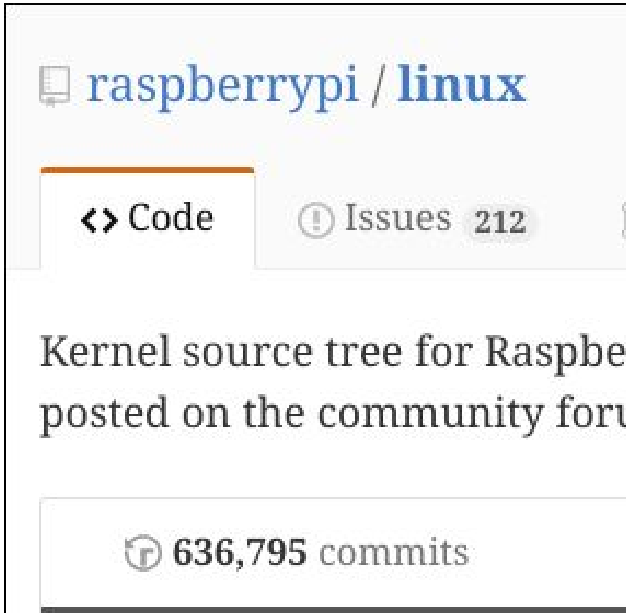
660,323 commits

You check different features like issues, pull requests etc….



And you click on a link and arrive here!

Will you repeat all the tests that you did for the previous repository? **NO!**



raspberrypi / linux

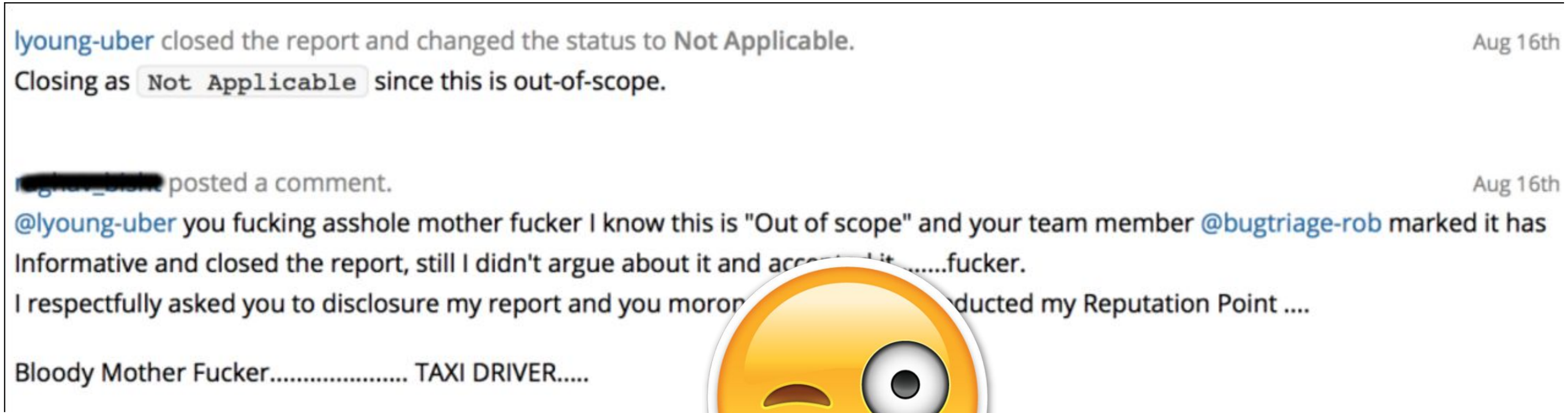
Code

Issues 212

Kernel source tree for Raspbe
posted on the community foru

636,795 commits

Apart from testing for vulnerabilities, this story outlines the approach of any application pentester!



lyoung-uber closed the report and changed the status to **Not Applicable.** Aug 16th

Closing as `Not Applicable` since this is out-of-scope.

[redacted] posted a comment. Aug 16th

@lyoung-uber you fucking asshole mother fucker I know this is "Out of scope" and your team member @bugtriage-rob marked it has Informative and closed the report, still I didn't argue about it and acc[illegible] ....fucker.
I respectfully asked you to disclosure my report and you moron [illegible]ducted my Reputation Point ....

Bloody Mother Fucker.................... TAXI DRIVER.....

Let me help you visualize the beauty of what you did in the story.

Recognize Sequences
From your previous experiences you recognize that you need to login but signup before that to access more functionality.

Identify Inputs
You understood that an email input field expects an email address while a credit card number expects a well, credit card number.

Understand Feedback
Feedback of application in this case incorrect format of password was understood by you and correspondingly the input was changed.

Recall Input Values
For logging in you recollected that these values are same as the ones provided during the sign-up hence used them.

Categorize Pages
While browsing the pages found were partially remembered for their functionality to be recalled later.

Avoid Redundancy
If a page similar to a previously browsed one turned up, it was ignored to prevent redundancy and save time.

Identify Inputs

Ability to correctly identify the inputs along with the format that is expected plays a vital role in understanding or navigating an application

Understand Feedback

Understanding the feedback given by the application makes the system self improving hence enhancing it's performance overtime.

Perform Sequences

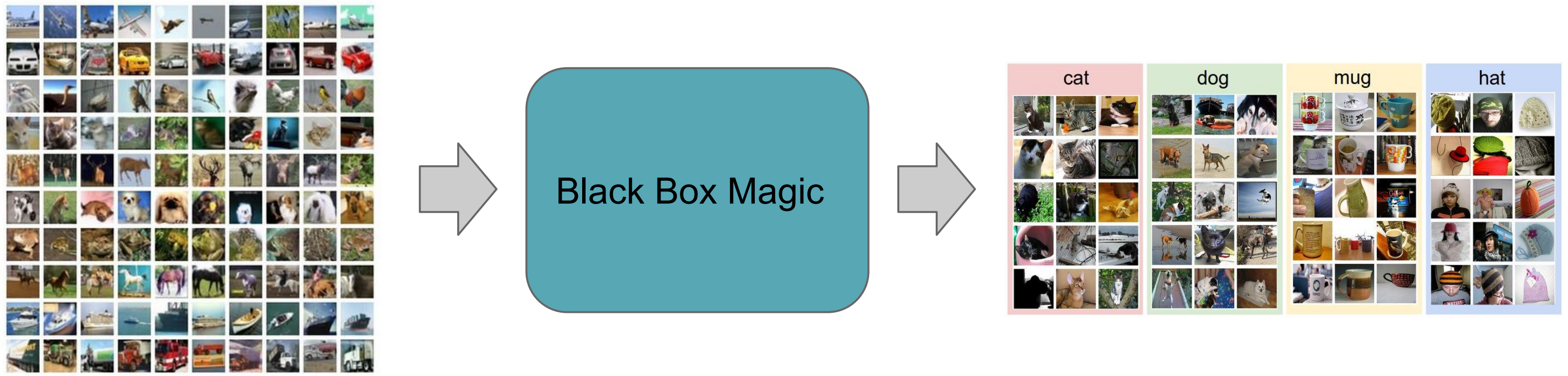
Being able to learn and perform simple sequence of steps like registering and logging in or adding a new address and editing it allows for better targeted navigation.

2

ML 101

Introduction to Machine Learning

- Sample aim of machine learning
- Technically, ml boils down to curve fitting, the black box magic can be defined as **output = a_function(input).**

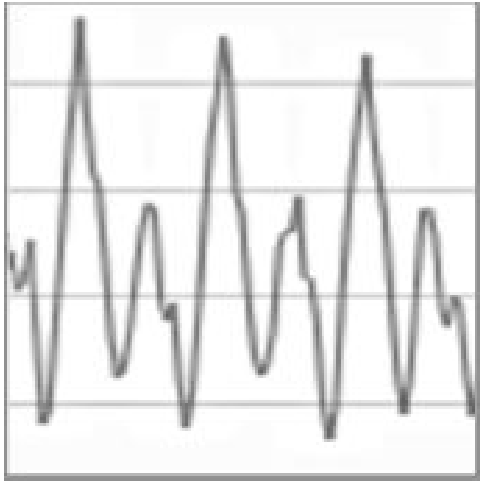
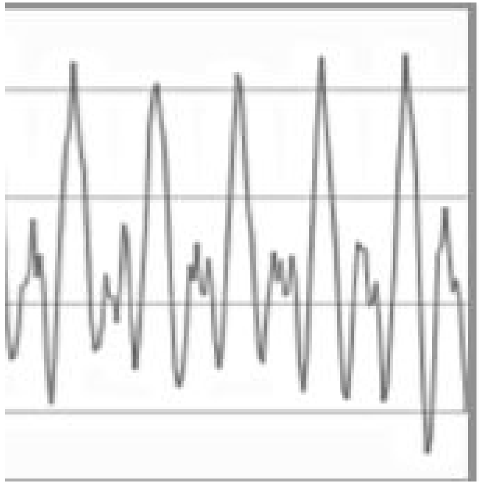
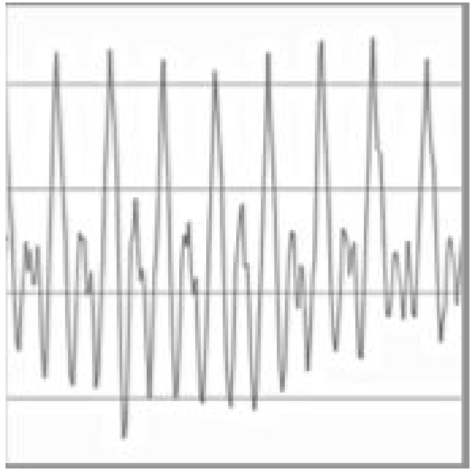


Let us think it through

- You are a **fitbit like manufacturer**, trying to find step count using your gadget, the data that you have is pulse rate.
- Now, you need to tell if user took 1,2 or 3 steps depending on his pulse rate.
- You need a function that is like **steps = f(pulse_data)**

Sample Pulse Data

- What do you think is the most important aspect of pulse to determine the step count? Guess for the following samples.

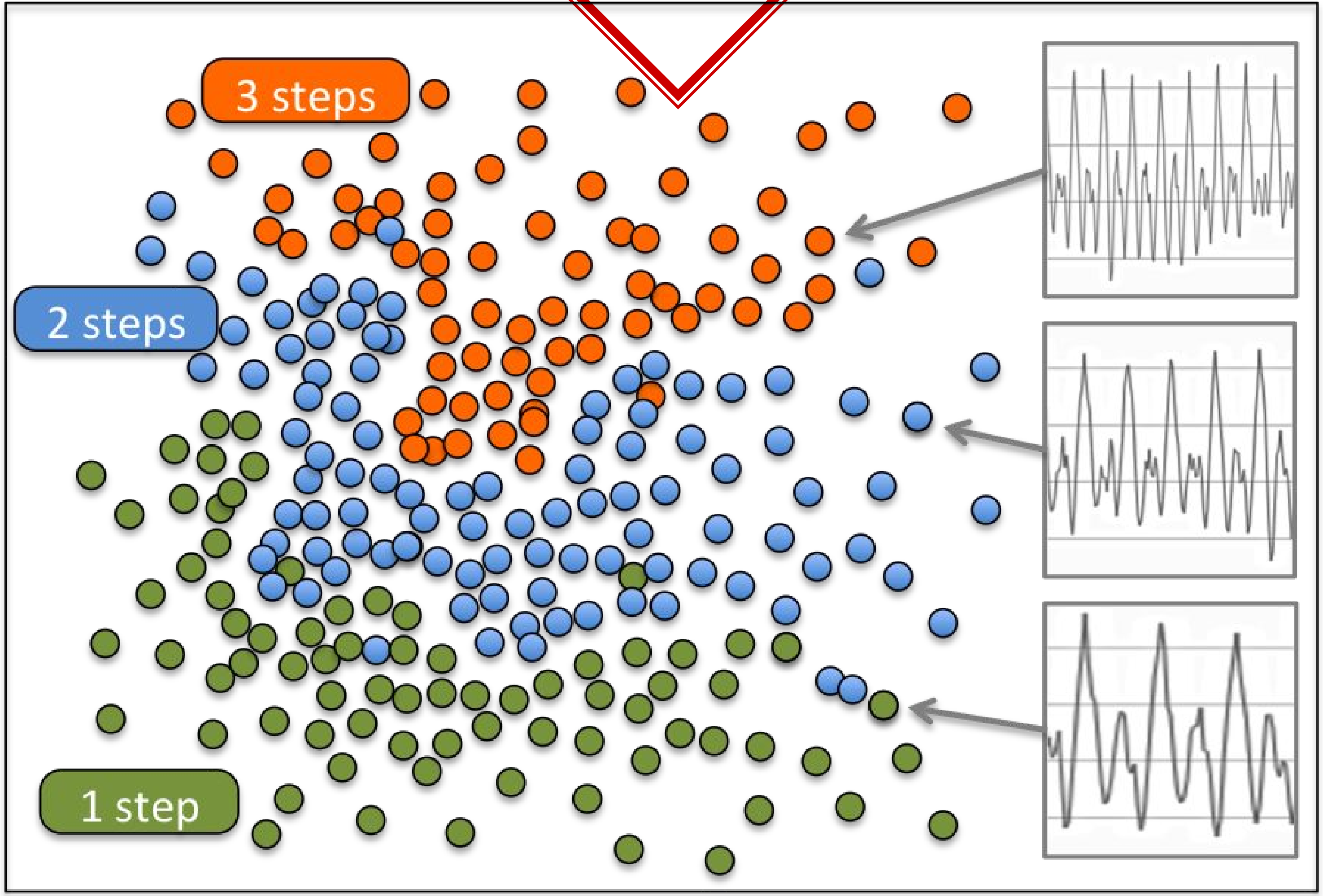




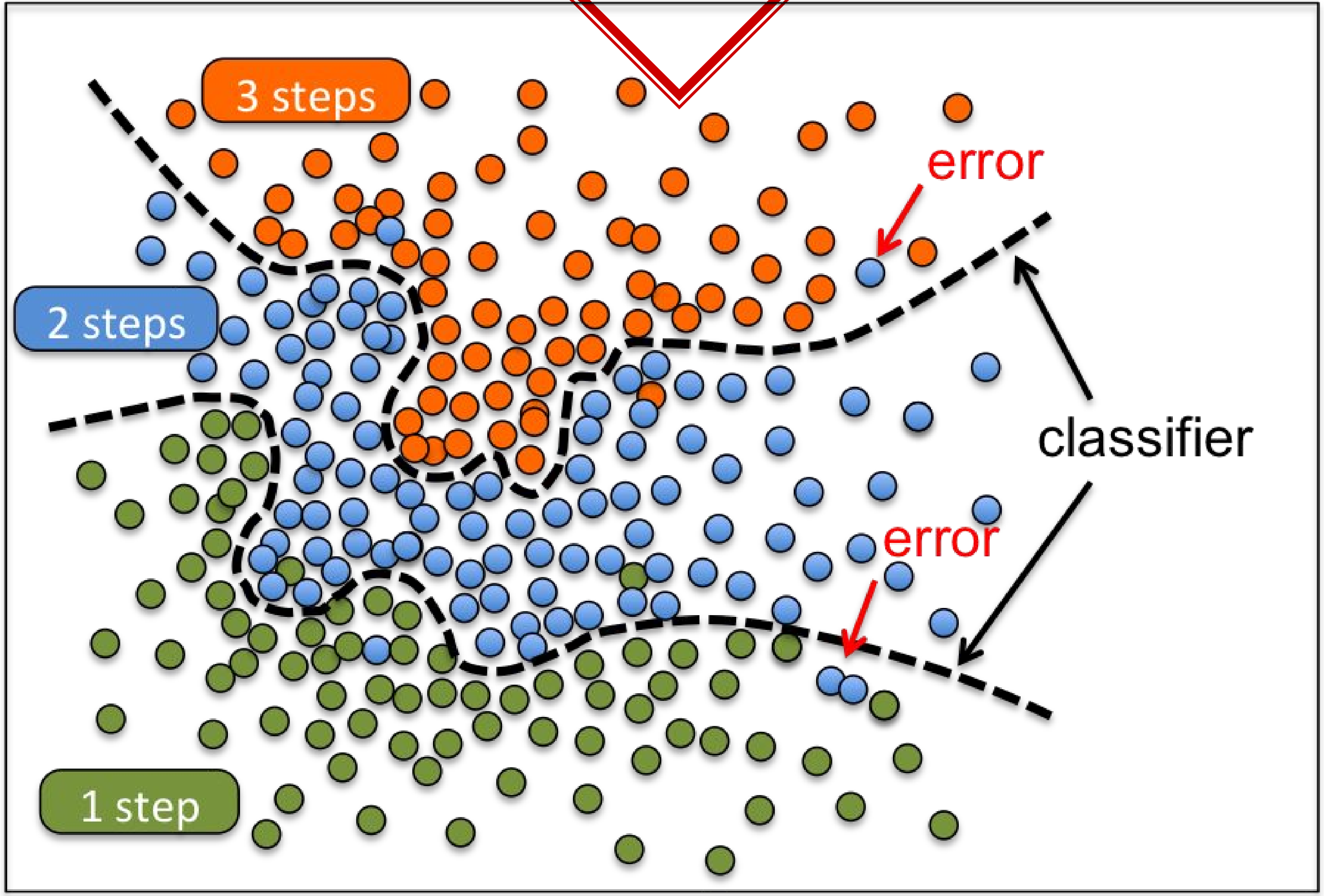
How did you guess?

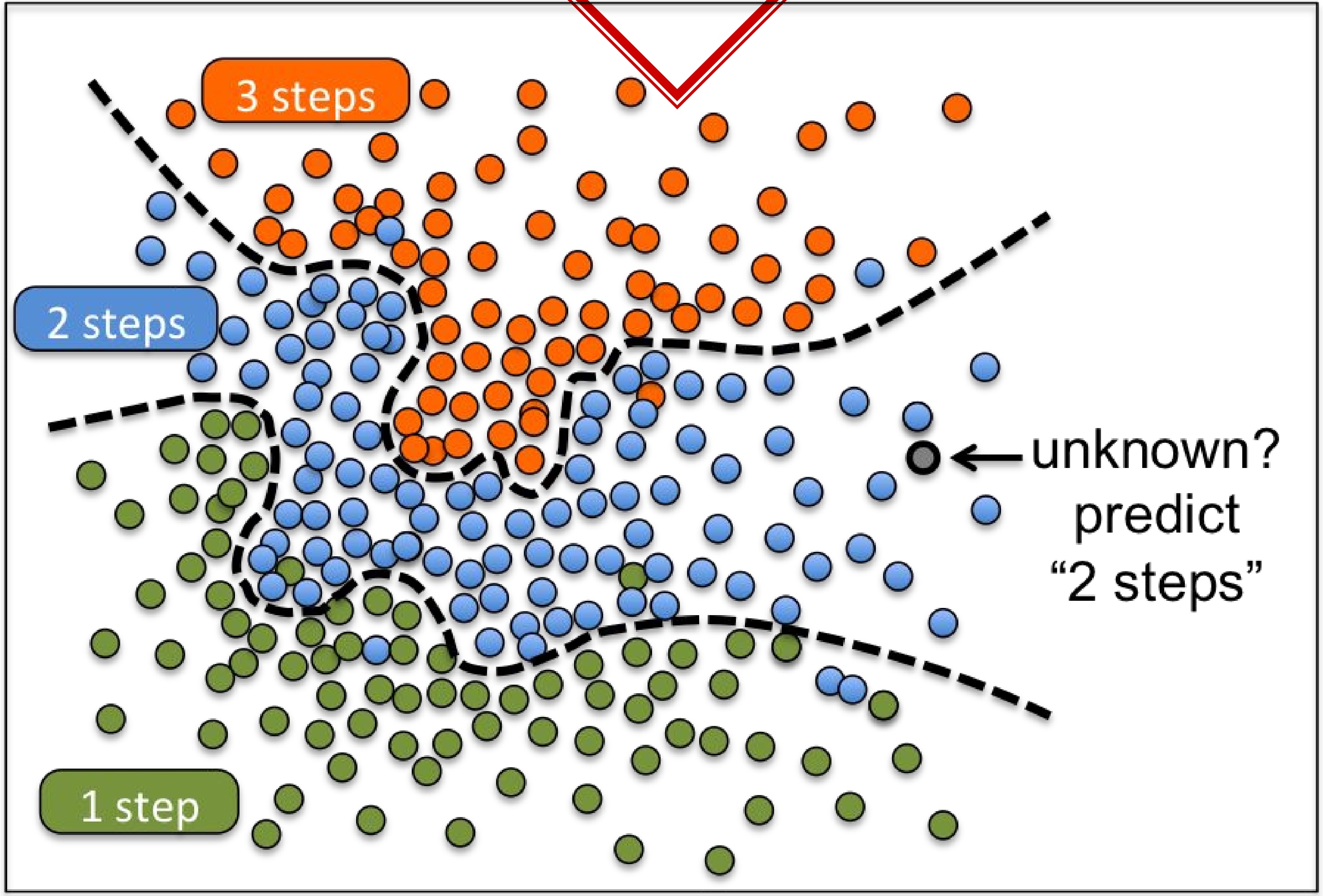
- **Pulse height** and **time between pulses.**
- Simplified function

steps = f(average_pulse_height, average_time_between_pulses)

- What we did above is called **feature engineering.**
- If you plot the pulse data samples and known step counts on a 2-d plot


3 steps
2 steps
1 step


3 steps
error
2 steps
classifier
error
1 step


3 steps
2 steps
1 step
unknown?
predict
“2 steps”

Technical Recap!

- We extracted useful features from input data.
- Collected lots of samples.
- Train the samples.
- What we just did is called **Supervised Learning.**
- Obviously because we had to label the training data ourselves.

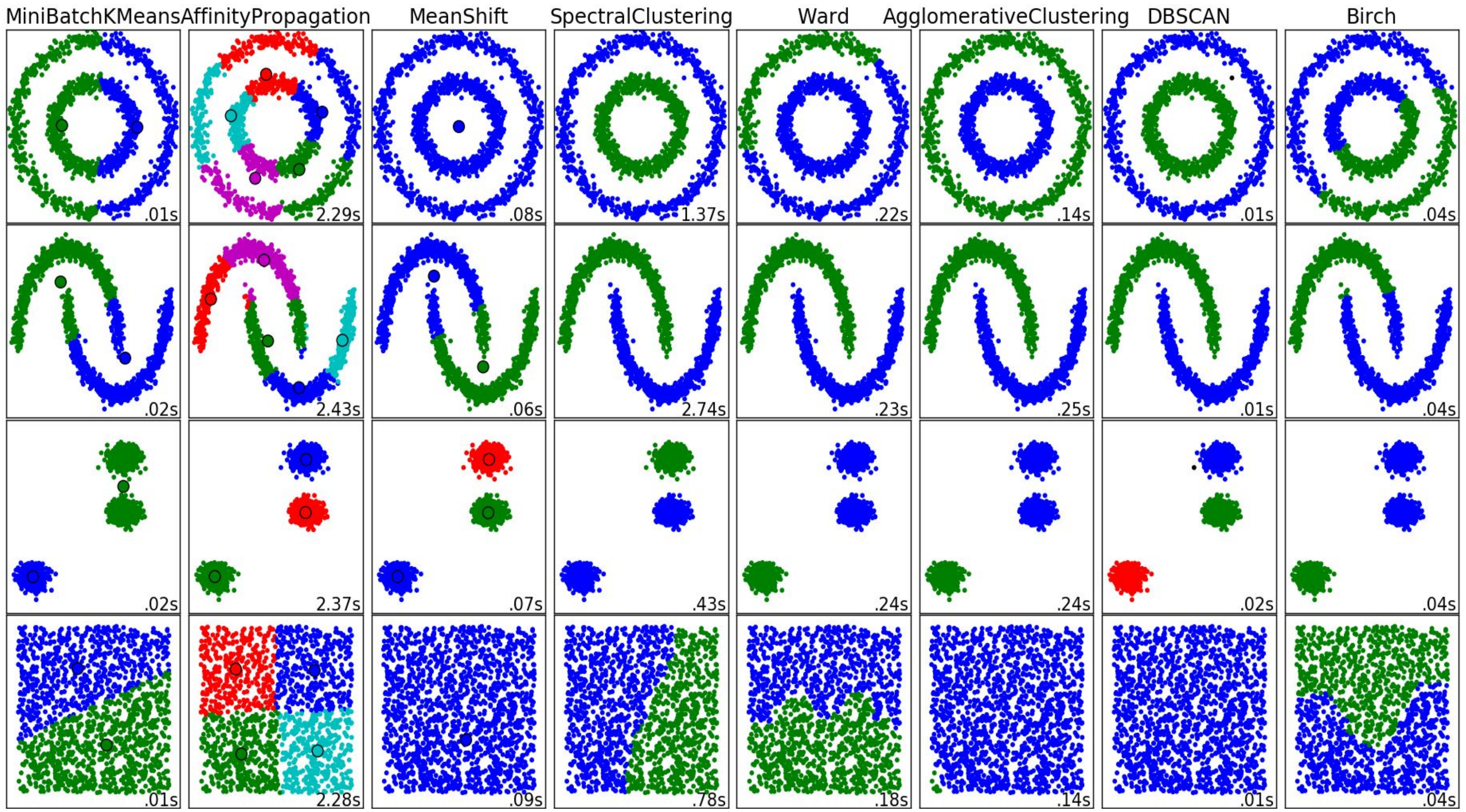
Variants of Machine Learning

- **Supervised Learning** ← We just saw this!
- **Unsupervised Learning**

- **Semi-Supervised learning** is a class of supervised where small amount of labelled data is used along with large amount of unlabelled data.

Unsupervised Learning

- Consider same problem statement.
- Instead of hand labelling all the training data, you just tell how many **clusters** need to be formed in the data.
- Depending on the algorithm and hyperparameters you get your function.
- Downside of this is the **difficulty in visualizing the clusters** especially in **high dimensional data**.

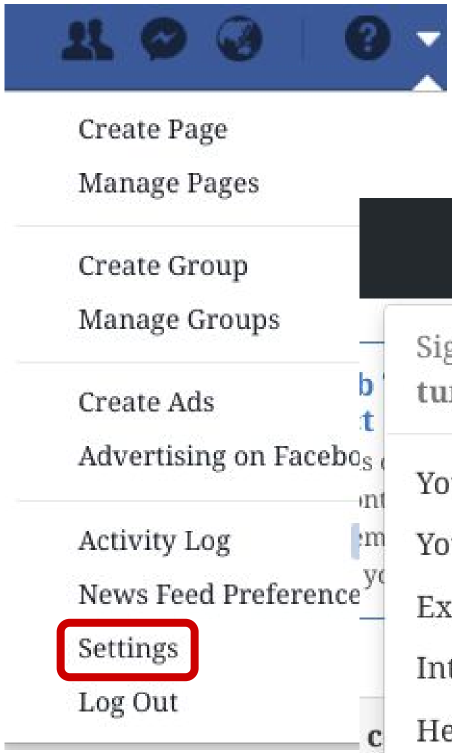

MiniBatchKMeans
AffinityPropagation
MeanShift
SpectralClustering
Ward
AgglomerativeClustering
DBSCAN
Birch
.01s
2.29s
.08s
1.37s
.22s
.14s
.01s
.04s
.02s
2.43s
.06s
2.74s
.23s
.25s
.01s
.04s
.02s
2.37s
.07s
.43s
.24s
.24s
.02s
.04s
.01s
2.28s
.09s
.78s
.18s
.14s
.01s
.04s

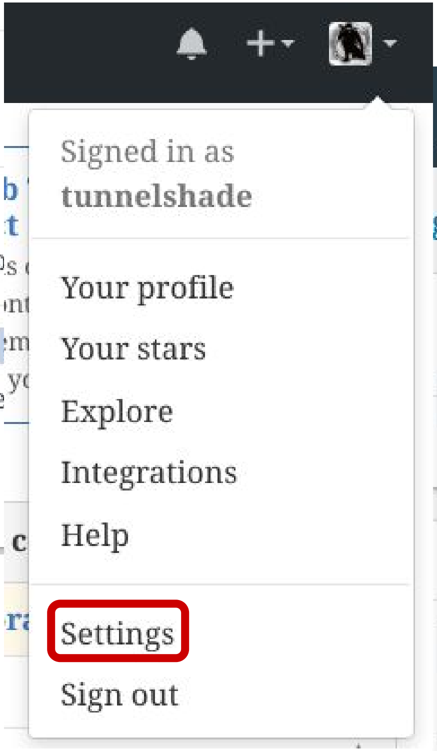
3

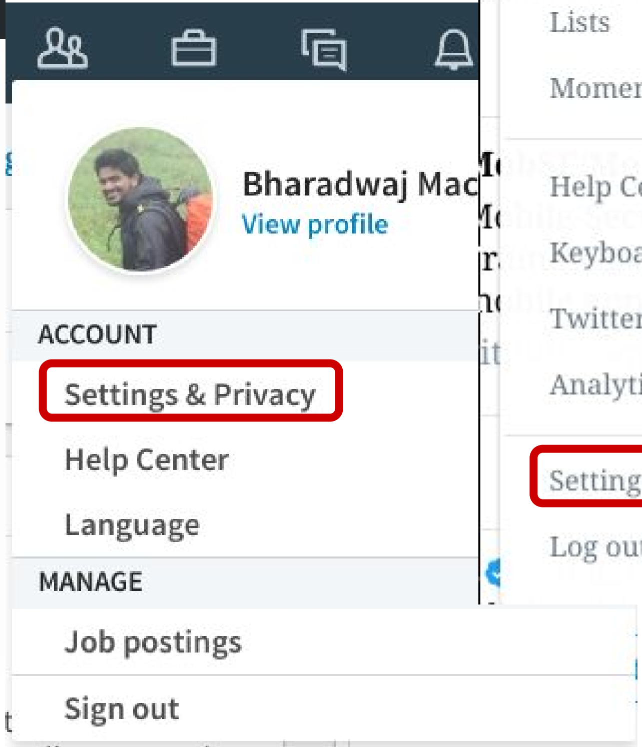
Problems of Text Representation

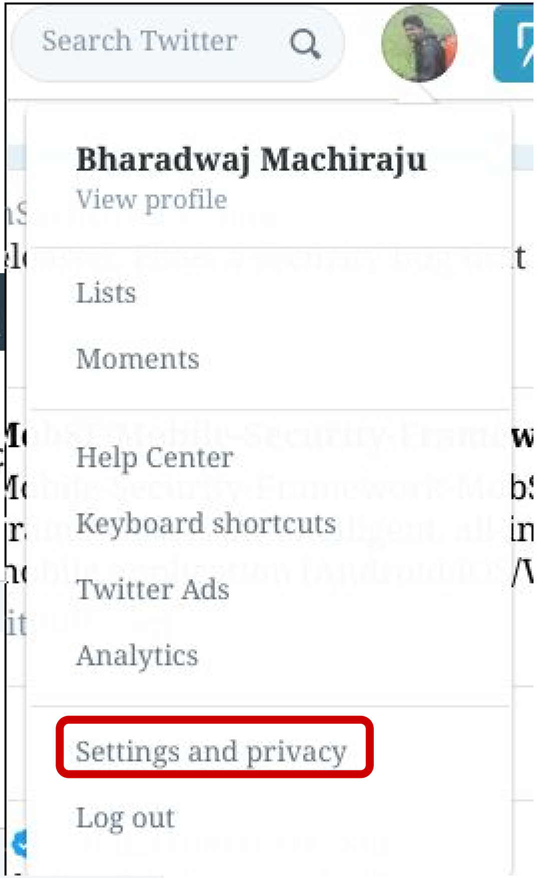
Time to get to the core issue.

Is text our only option?









Pretty much, YES!!

- Our understanding of any web application is heavily dependent on the textual content.
- Does that mean we should used it as our features? **Not necessarily**.
- But it is consistent across applications and general feedback from apps is textual again!

Why is Text processing hard?

- One major difficulty is that we don't **consciously understand language** ourselves. The second major difficulty is **ambiguity**.
- Representation of text in numbers is another story. Imagine the following line
  - What do you understand by **“Pressing a suit”**?
  - Now think like a **Lawyer** and then like a **dry cleaner**.

4

Identifying Inputs Failures & Successes!

Phew!!!

Identify Inputs

Ability to correctly identify the inputs along with the format that is expected plays a vital role in understanding or navigating an application

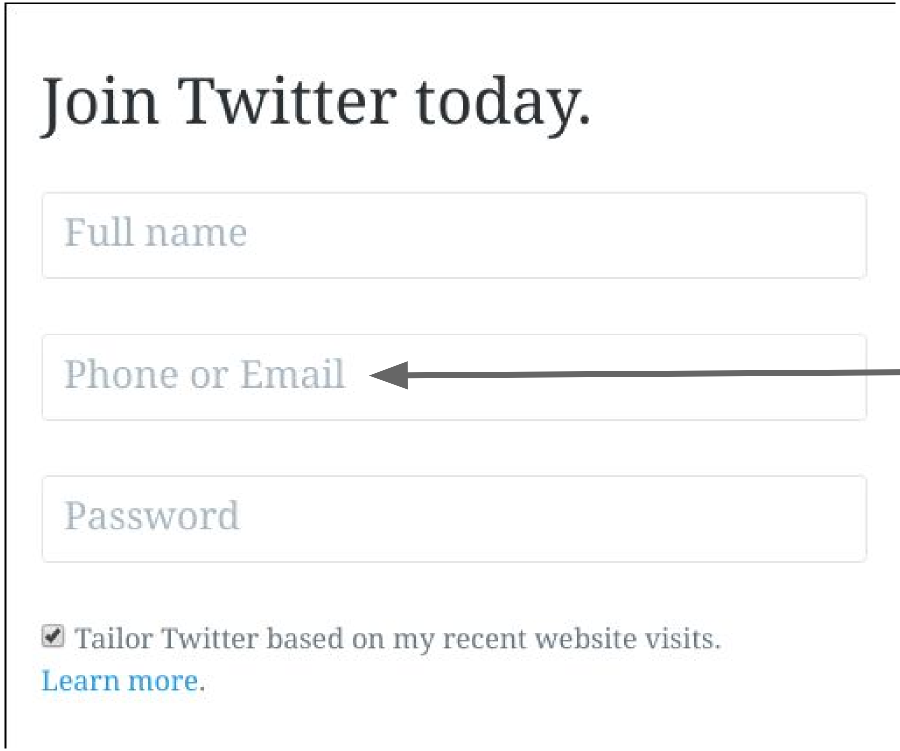
Understand Feedback

Understanding the feedback given by the application makes the system self improving hence enhancing it's performance overtime.

Perform Sequences

Being able to learn and perform simple sequence of steps like registering and logging in or adding a new address and editing it allows for better targeted navigation.

Lingo



Join Twitter today.

Full name

Phone or Email

Password

☑ Tailor Twitter based on my recent website visits. Learn more.

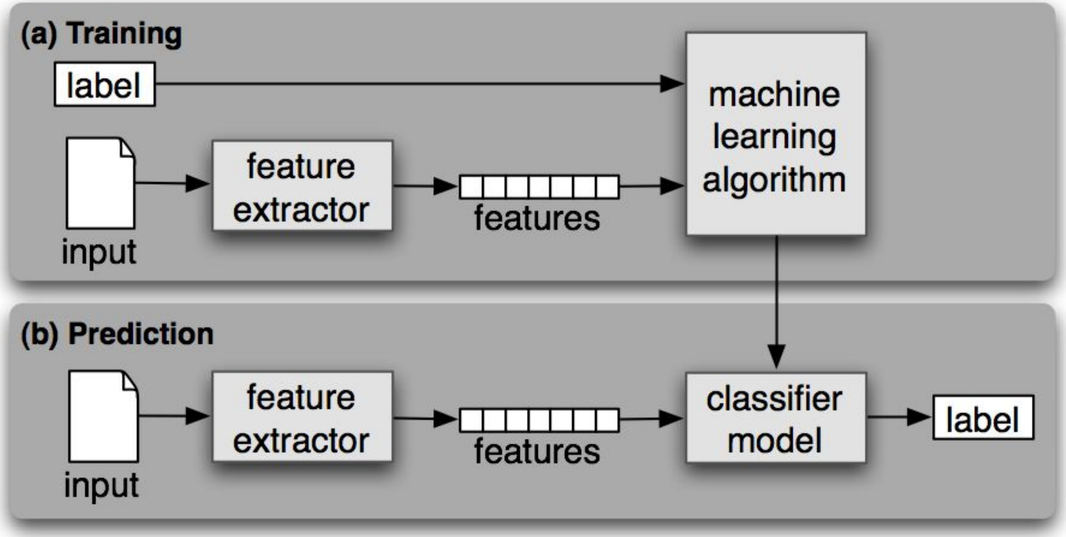
Placeholder

Labels

- E-Mail
- Mobile
- Dates
- Name
- ….

Attempt 1 - Hand-labelling Input placeholders

- Use simple NaiveBayes classifier with hand labelled training data gives good accuracy.



Attempt 1 - Training Data

```
{
    "text": "username",
    "label": "username"
},
{
    "text": "your username"
    "label": "username"
},
{
    "text": "password",
    "label": "password"
},
```

```
{
    "text": "mobile",
    "label": "mobile"
},
{
    "text": "address",
    "label": "address"
},
{
    "text": "zip/pin code",
    "label": "zipcode"
},
```

Attempt 1 - Hand-labelling Input placeholders

- Pros
  - Simple solution
- Cons
  - **Hand labelling takes manual effort**. This is a big NO NO for me.
  - Unknown placeholders or a different language support is a whole new story.

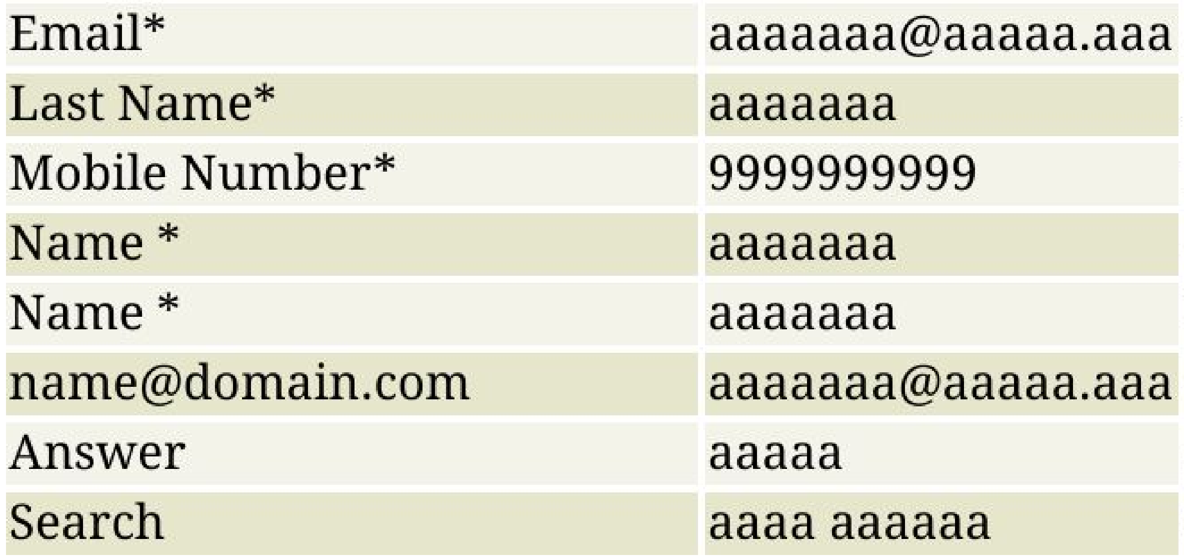
After few days of trying to gather and label placeholders.
I thought the whole point was to simplify human efforts :(

Data Gathering

- You know how **Tesla's AutoPilot learns**? **From humans**!
- :D
  - Build a browser extension and share it with people.
  - Collect input data formats they are filling.
  - Use it as training data!! **Yuhoo**!

Data Gathering

- Instead of a full extension, wrote a userscript for TamperMonkey addon and let users install that script.



Email*	aaaaaaa@aaaaa.aaa
Last Name*	aaaaaaa
Mobile Number*	9999999999
Name *	aaaaaaa
Name *	aaaaaaa
name@domain.com	aaaaaaa@aaaaa.aaa
Answer	aaaaa
Search	aaaa aaaaaa

placeholder	suggestion
Company	AaaaAaaaaaaa
Credit card	aaaaaa_aaaa
Credit card number	9999 9999 9999 9999
Update README.md	Aaaaaa AAAAAA.aa
Credit / Debit Card Number	9999 9999 9999 9999
CVV	999
Credit / Debit Card Number	9999 9999 9999 9999
CVV	999
Receive occasional updates and special offers for...	AA
Filter by date	9
Enter CVV	999
Check-in date	Aaa, 99 Aaa 9999
Check-out date	Aaa, 99 Aaa 9999
Credit/Debit Card Number *	9999999999999999
CVV Code *	999
Company Name	Aaaaaaaaaa Aaaaaaaaaaaa

Attempt 1 - Fails!

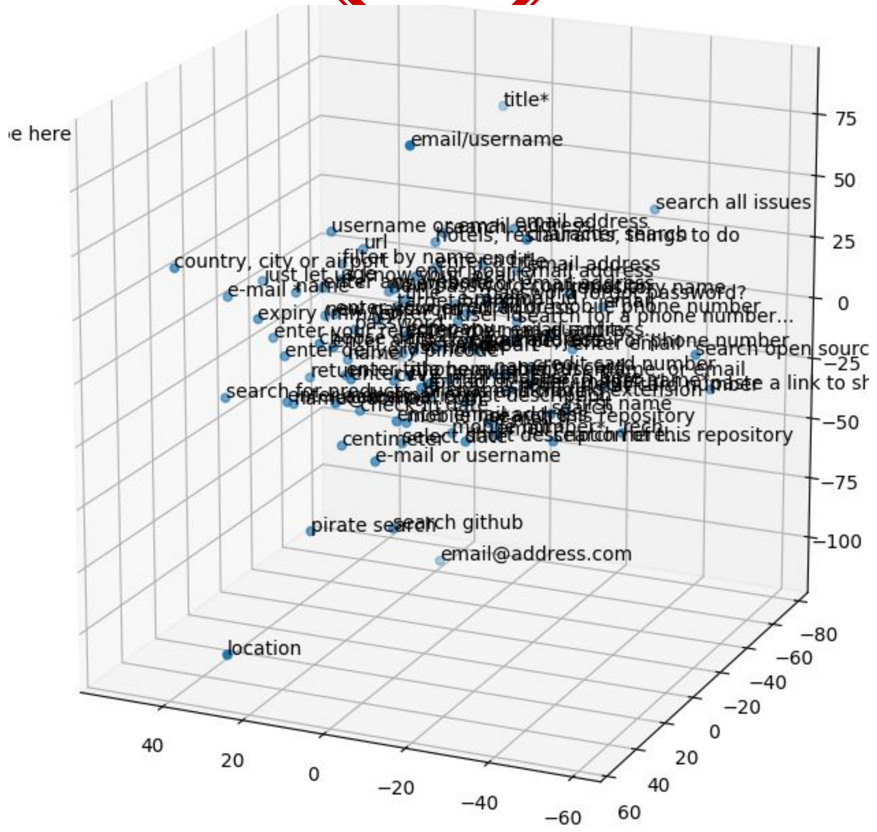
- Even though I have lots of data, I **cannot label them manually or automatically**.
- Manually is too much effort.
- Automatically not possible because input formats cannot be categorized without complex rules, so drop classification.
- Clustering is a daunting task as the data is high dimensional.

Attempt 2 - Placeholder vectorization

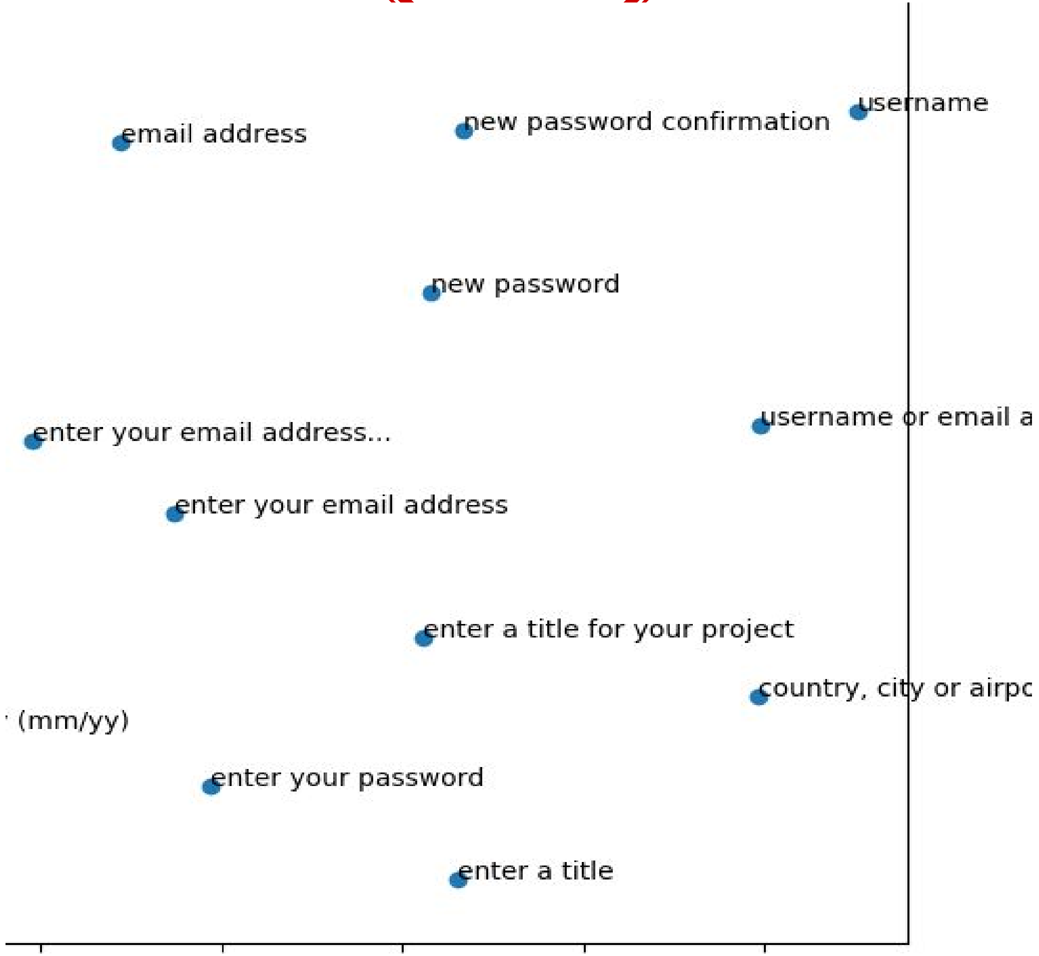
- Idea was to **convert placeholders into a multi-dimensional vector space** so that **similar placeholders are nearby**. Used Term Frequency-Inverse Document Frequency.
- Imagine a 3d space, our vectorization should be such that

  f(“Enter your email address”) ≅ f(“E-mail address..”)

- So, if you get a new placeholder, you can check the spatial cosine distance between the placeholder with all known data to get most similar matches. Let us visualize



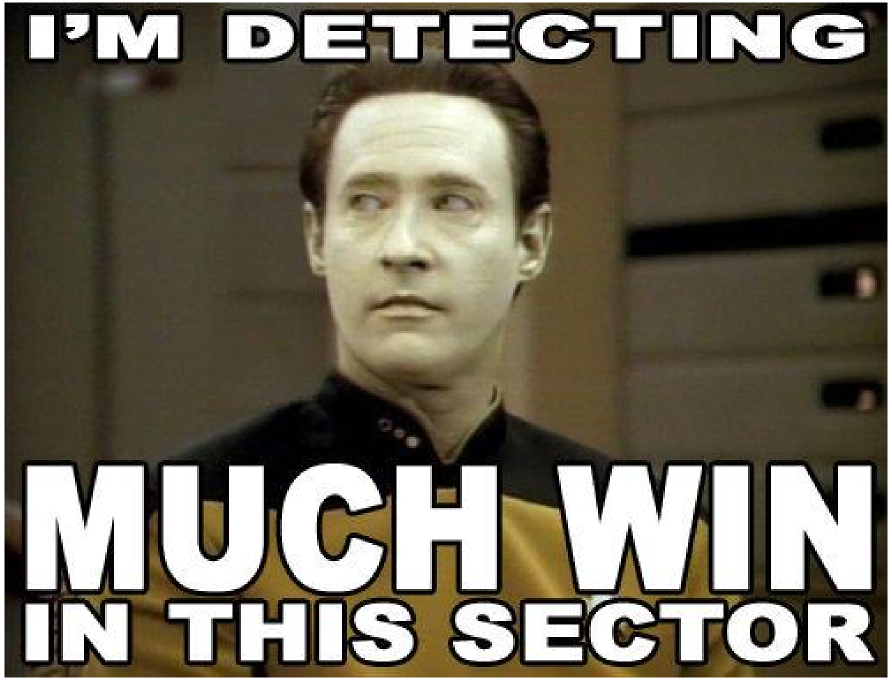
Attempt 2 - 3D Visualization :P



Attempt 2 - 2D Visualization

Attempt 2 - Placeholder vectorization (DEMO!)

- So now, whenever a new placeholder is encountered similarity is checked with other placeholders to determine the input and if input is successful, the new placeholder is added into training data….



5

Understanding Application Feedback

Vroooooom!!!

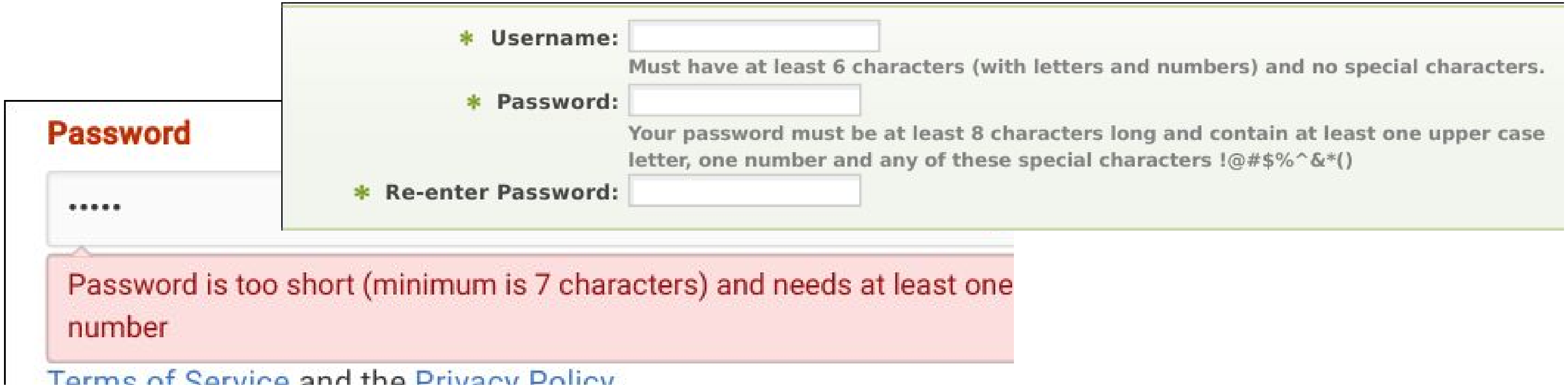
Identify Inputs
Ability to correctly identify the inputs along with the format that is expected plays a vital role in understanding or navigating an application

Understand Feedback
Understanding the feedback given by the application makes the system self improving hence enhancing it’s performance overtime.

Perform Sequences
Being able to learn and perform simple sequence of steps like registering and logging in or adding a new address and editing it allows for better targeted navigation.

By Murphy's Law, you will encounter

- Unknown placeholders or previously unknown input formats.
- Way to deal with them is to understand the feedback given by the application



* Username:

Must have at least 6 characters (with letters and numbers) and no special characters.

* Password:

Your password must be at least 8 characters long and contain at least one upper case letter, one number and any of these special characters !@#$%^&*()

* Re-enter Password:

Password

••••••

Password is too short (minimum is 7 characters) and needs at least one number

Terms of Service and the Privacy Policy

Attempt 1 - Use PoS tagging!

- PoS = **Parts of Speech**
- One way to understand feedback like these is to breakdown the **sentence into phrases and extract information**.
- Pros
  - It works!! ← Yes, it is a pro
- Cons
  - **Complex logic to get it right.**
  - Different languages might need different logic.
  - Illformed english is a NO NO.

Attempt 1 - PoS tagging!

Must/MD/B-VP/O **have**/VB/I-VP/O **at**/IN/B-PP/O **least**/JJS/B-ADJP/O **6**/CD/B-NP/O **characters**/NNS/I-NP/O **(**/(/O/O **with**/IN/B-PP/B-PNP **letters**/NNS/B-NP/I-PNP **and**/CC/I-NP/I-PNP **numbers**/NNS/I-NP/I-PNP **)**/)/O/O **and**/CC/O/O **no**/DT/B-NP/O **special**/JJ/I-NP/O **characters**/NNS/I-NP/O ./././O/O

Code: https://goo.gl/cXEeiC (DEMO?)
Video: https://www.youtube.com/watch?v=UBcTwMShzsQ

Attempt 1 -Input Help Relation!

- How to relate a help text to a particular input field?
  - **Noun reference** - When the help text directly refers to the input placeholder like “Your password should be …”
  - **Visual Correlation** - When the help texts are placed in such a way that they are visually related to the input.

Attempt 2 - LSTM (In Progress)!

- **LSTM = Long Short Term Memory** network (a class of Recurrent Neural Networks)
- Using **seq2seq** to translate these help texts into suggestions like previously seen.
- Seq2seq is a popular model which are **extremely successful in translation tasks**.

Attempt 2 - LSTM (In Progress)!

- **Cons**
  - **Lots of training data is necessary**
  - Manually identification of these help texts will be initially required.

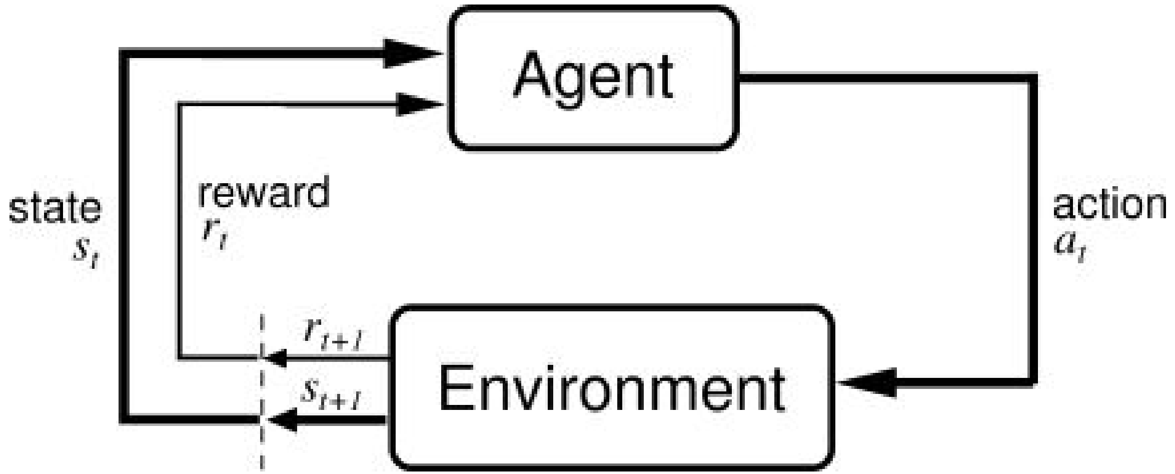
6

Reinforcement Learning 101

AI is here!

RL

- Reinforcement learning is the most intuitive of all as it is inspired from behaviorist psychology.



State = outdoors, action = play, positive reward

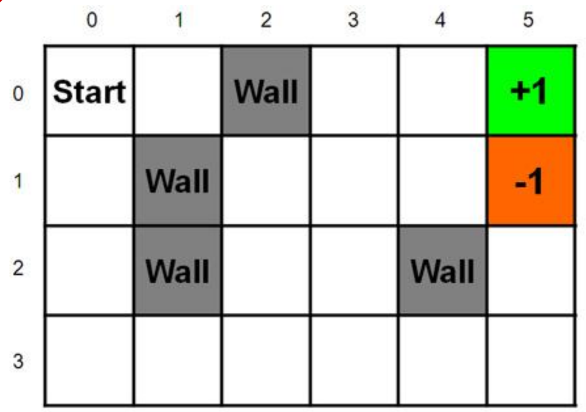
State = kitchen, action = play, reward = flying chappal!

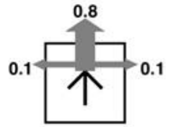
State = kitchen, action = play, reward = well!

RL

- We humans **tend to repeat actions** in states which lead to **cumulative positive reward** while avoiding actions that lead to a cumulative negative reward.
- So, generally an agent can be **rewarded** based on the states it arrives in which in long term make the agent biased towards the actions that lead to these states.
- Any problem that can be expressed as a **Markov Decision Process** can be an application for RL.

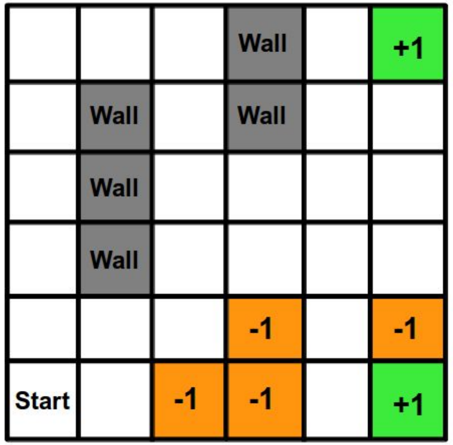
Maze Solver

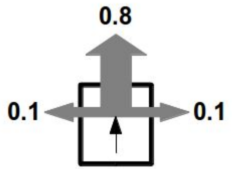




- Actions: Left, Right, Up, Down
- State: [left_box_type, right_box_type, up_box_type, down_box_type]
- Reward: +10 for “+1” box, -1 for every step and -10 for “-1” box

Can your previous agent solve this maze?





- Actions: Left, Right, Up, Down
- State: ???
- Reward: +10 for "+1" box, -1 for every step and -10 for "-1" box

[left_box_type, right_box_type, up_box_type, down_box_type, **direction_of_nearest_+1**]

Value Function

- A function which states how good it is to perform certain action in certain state **Q(state_vector, action_num).**
- **Higher the Q value more profitable the action will be**.
- Value functions are generally stored in different ways of function approximators like **RBF, Neural Networks etc..**.
- So, when you say **experience is stored**, it means that this **value function was updated according to latest occurances**.

7

Perform Sequences

We just saw how important state representation is.

Identify Inputs
Ability to correctly identify the inputs along with the format that is expected plays a vital role in understanding or navigating an application

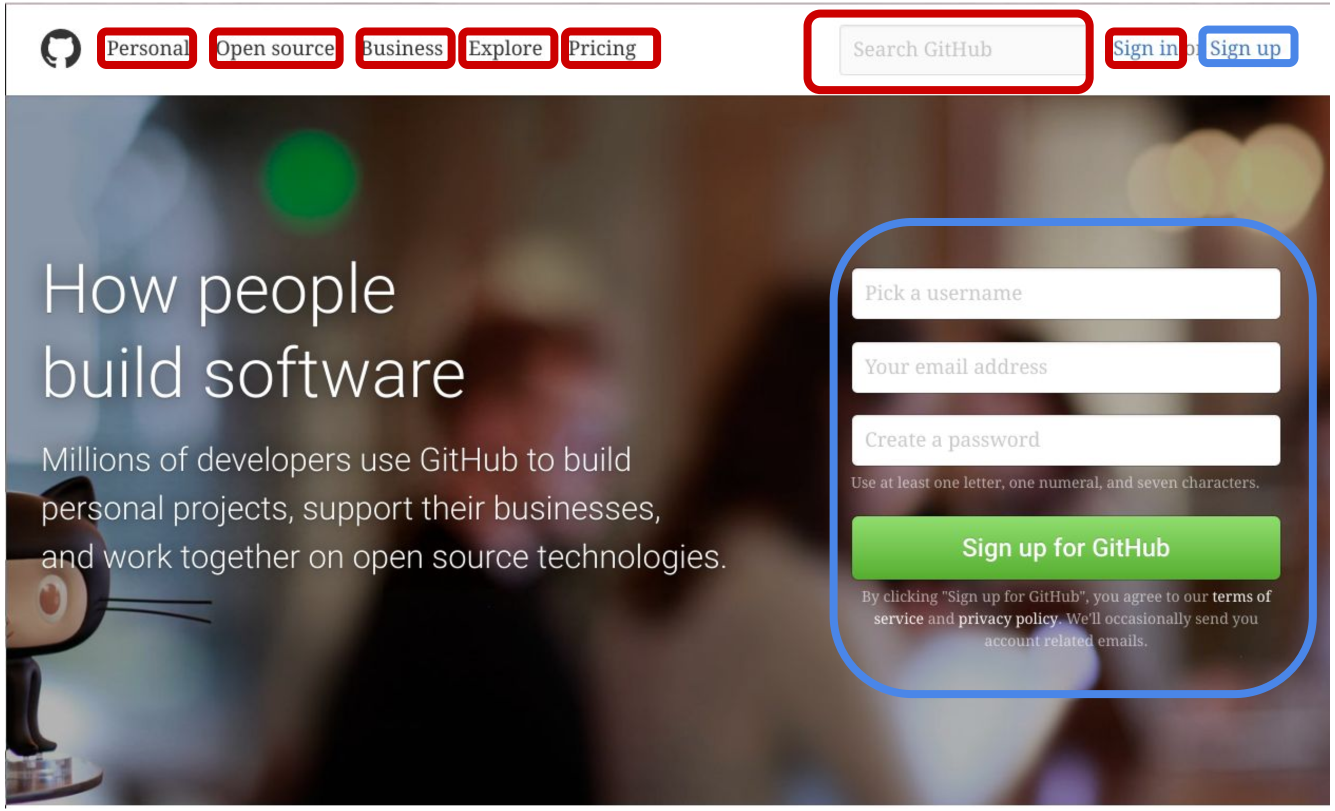
Understand Feedback
Understanding the feedback given by the application makes the system self improving hence enhancing it's performance overtime.

Perform Sequences
Being able to learn and perform simple sequence of steps like registering and logging in or adding a new address and editing it allows for better targeted navigation.

In a regular web scenario, learning sequences

- Learning the sequence of link clicks and form fills! So, these two types generally constitute our **actions**.
- **Reward** based on the end result requirements.
- What is **state** made up of then?

Let us roll back to what we did the beginning? Why you chose Sign Up?



How people build software

Millions of developers use GitHub to build personal projects, support their businesses, and work together on open source technologies.

Pick a username

Your email address

Create a password

Use at least one letter, one numeral, and seven characters.

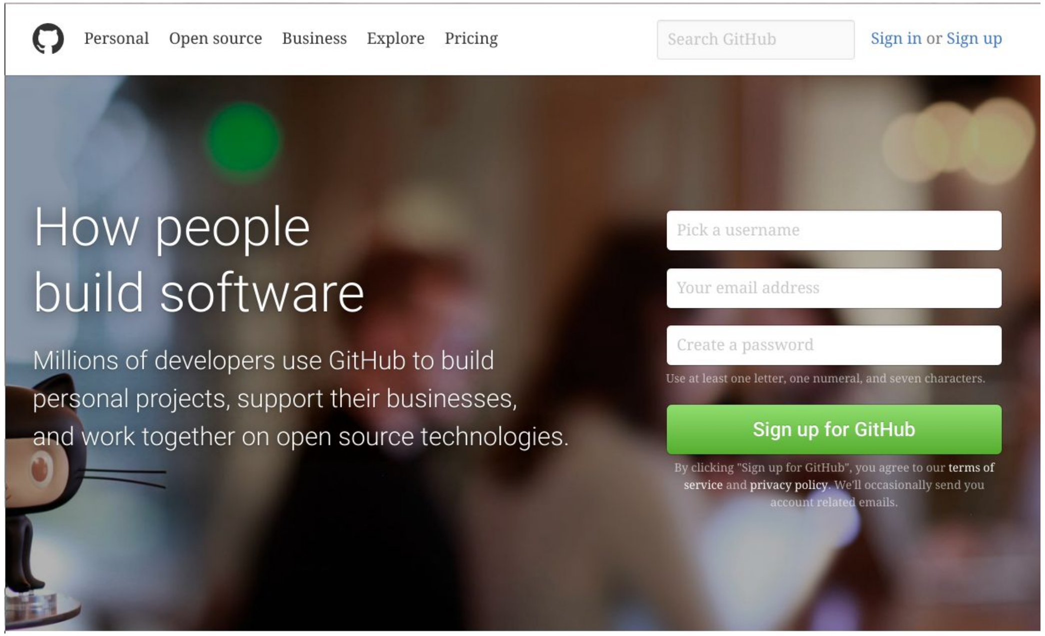
Sign up for GitHub

By clicking "Sign up for GitHub", you agree to our terms of service and privacy policy. We'll occasionally send you account related emails.

Why SignUp?

- You have an idea that clicking on a text “SignUp” generally leads to a form where you can register.
- You didn’t click on other links on the page like Open Source etc.. because the **reward** that you were pursuing is to login.

Attempt 1 - State Representation



How people build software

Millions of developers use GitHub to build personal projects, support their businesses, and work together on open source technologies.

Use at least one letter, one numeral, and seven characters.

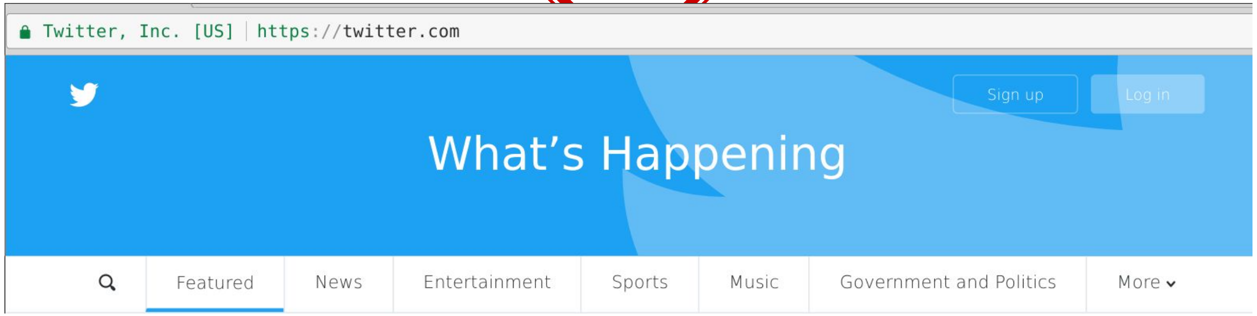
By clicking "Sign up for GitHub", you agree to our terms of service and privacy policy. We'll occasionally send you account related emails.

Attempt 1 - State Representation

- For a machine, state need to be **numbers**. So again we have to represent the **text in a numerical form** so that **similar states are closer in their vector form**.
- What might be a good way?

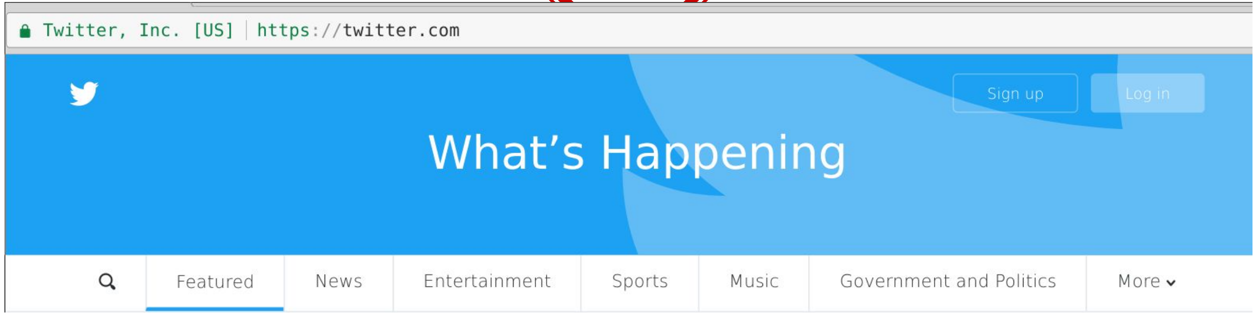
Attempt 1 - State Representation

- Why not just **add all the link texts, form placeholders and label input data as a string** and **vectorize** it?
- **Doc2Vec/Paragraph2Vec** was used for this.
- The idea still remains same, you have a bunch of state strings which when mapped have the similarities between them.
- For example



Assuming this is first visit, the state at this point of time is made from all the links (**“Featured”**, **“News”**, **“Entertainment”**, **“Sports” …..**) + Forms (**There aren’t any**) + Label Values (**No label values**)

= **[0.23423…, 0.3242…, 0.23423…]**

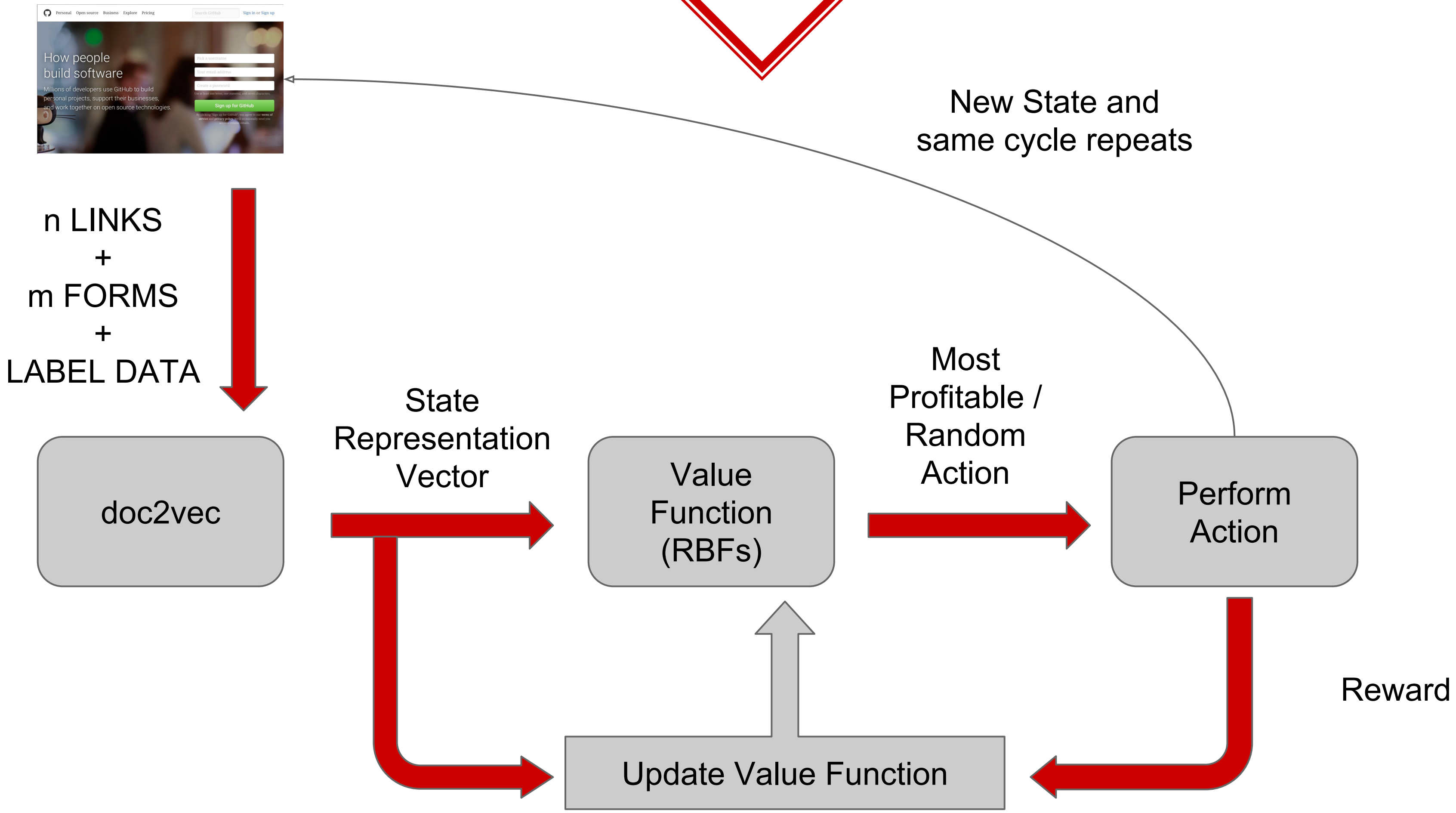


When we have an account in twitter i.e have label values for username and password, the state for the same page is different, which is all the links (**"Featured", "News", "Entertainment", "Sports" ....**) + Forms (**There aren't any**) + Label Values (**username, password, email**)

= **[0.25192…, 0.40123…, 0.29188...]**

Attempt 1 - Actions & Reward

- **Possible actions** in each state were
  - **Clicking a link** (which was part of state info)
  - **Filling a form** (which was part of state info)
- **Reward**
  - Each step: -1
  - **Successfully Logged In: +20**
- **LSPI SARSA agent** was used with **RBF** for storing the value function.
- Let us quickly look at it step by step



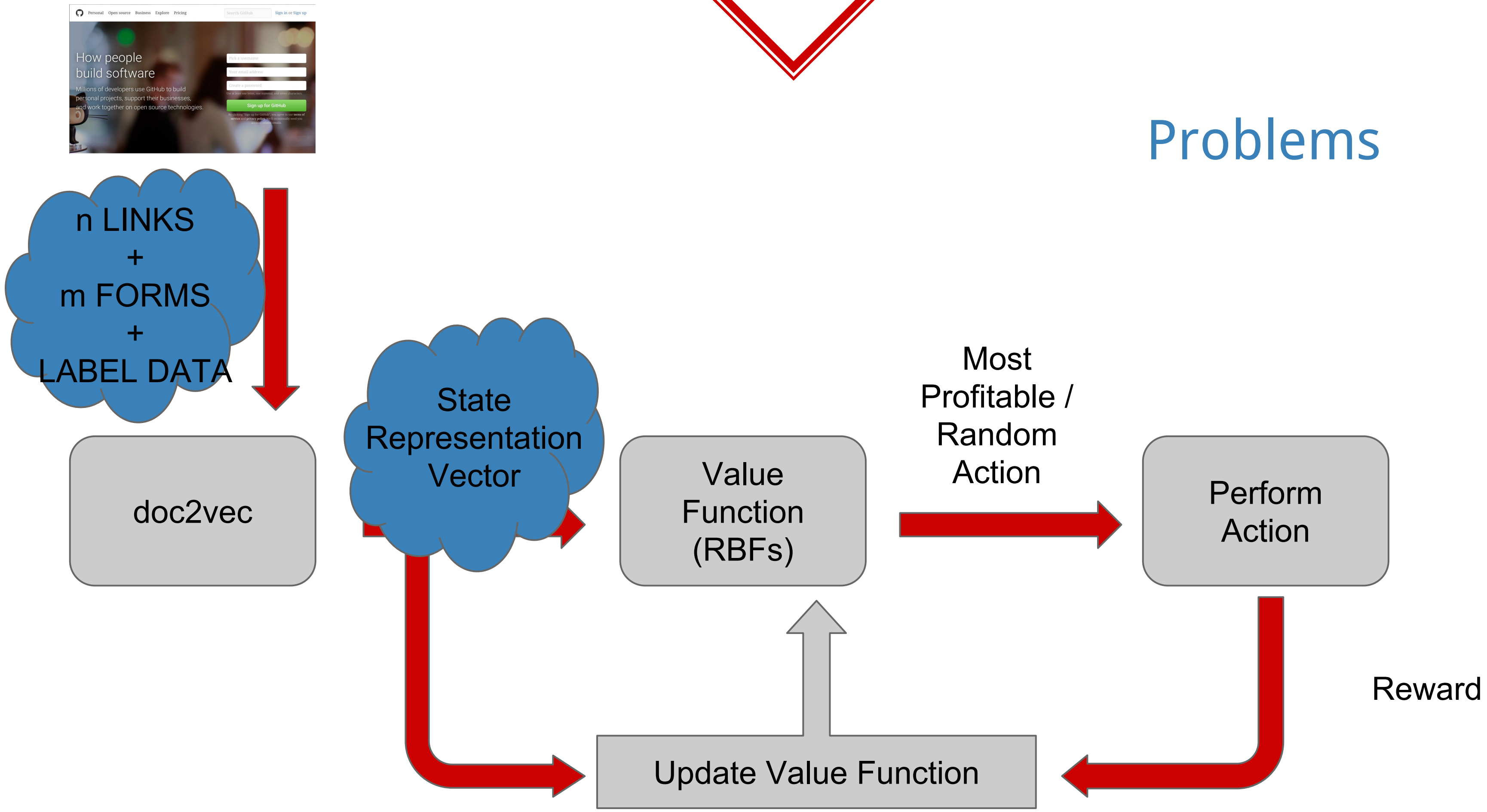
How people build software

Millions of developers use GitHub to build personal projects, support their businesses, and work together on open source technologies.

Attempt 1 - Did it work?

- Oh boy it did, but only for **simple applications**.
- Huge applications like E-Commerce sites have lots of links of which most of them lead to a product page.
- The state representation being **continuous** poses **convergence challenges to the policy**. I.e even though the state vector is only 3 dimensional, those three dimensions can take any value between -1 to +1 which makes the **state space extremely complex** for quickly learning a policy.

Problems


How people build software
Millions of developers use GitHub to build personal projects, support their businesses, and work together on open source technologies.
Sign up for GitHub
n LINKS + m FORMS + LABEL DATA
doc2vec
State Representation Vector
Value Function (RBFs)
Most Profitable / Random Action
Perform Action
Reward
Update Value Function

Attempt 1 - Arrghhhh!

Attempt 2 - State Representation

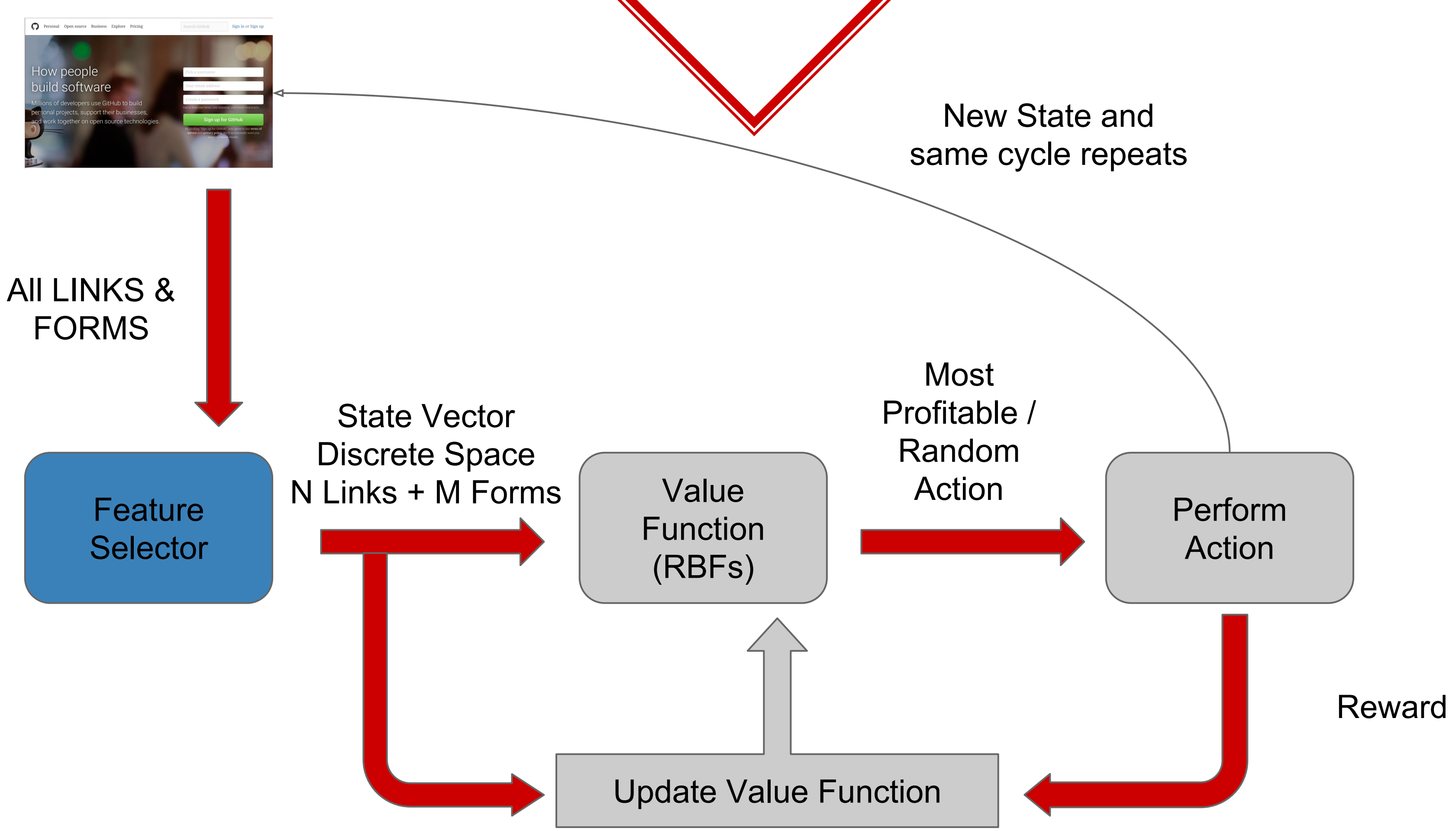
- Let us try to solve the problems in the previous model
  - **Limited set of links & forms for consideration**.
  - **Huge state space**.
- One good way is to use **an unsupervised feature selector** and then converting those selected features into a state vector.

Attempt 2 - State Representation

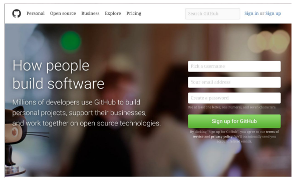
- The feature selector will allow us to consider **huge number of links and forms at the same time** giving a **discrete state space**.
- The added **complexity** is to **train the feature selector** along with the value function.

Attempt 2 - Updated Training Algo

- **Randomly pick a subset of features** and try episodic learning with the RL agent.
- If an episode ends with a **positive reward**, try to store the elements (i.e links and forms) if they are not already present.
- For the next episode, **label the required number of forms or links** as per the stored elements.
- These **labels constitute the state vector**.
- Damn!! This is difficult to explain, let us look at an example


Personal
Open source
Business
Explore
Pricing
Search GitHub
Sign in or Sign up
How people build software
Millions of developers use GitHub to build personal projects, support their businesses, and work together on open source technologies.
Sign up for GitHub
New State and same cycle repeats
All LINKS & FORMS
State Vector Discrete Space N Links + M Forms
Most Profitable / Random Action
Feature Selector
Value Function (RBFs)
Perform Action
Reward
Update Value Function

Login Module



How people build software

Millions of developers use GitHub to build personal projects, support their businesses, and work together on open source technologies.

All LINKS & FORMS

Feature Selector

Trained to pick only login & signup links and forms.
2 links and 2 forms are labelled

State Vector
[1.0, 2.0, 1.0, 0.0]

[1, 2, 4, 0] = [link1, link2, form1, form2]

Link Labels

1. SignUp type link
2. Login type link

Form Labels (0 indicates no match)

1. Login Type Form
2. Signup Type Form

Attempt 3 - Semi-Supervised Approach

- Instead of **randomly picking a subset of features**, manually **perform new sequences for once**.
- This greatly reduces the training time as you **assist the feature selector** in picking the right features.
- So **feature selector** is trained in a **semi-supervised approach.**
- Using only one module per sequence makes it more efficient and simple state and action space.

8

Vulnerability Detection

Reflected XSS PoC.

Idea

- Consider the markup **<div class="INJECTION_POINT">**.
- In order to execute js you come out of the class attribute context and put a payload.
- A simple vector in the above scenario is **"><img src=x onerror=alert()>**.
- This is very trivial because of your exposure to **HTML markup**.
- If we can somehow impart the knowledge of html to an RL agent, it should be able to provide **some simple XSS payloads**.

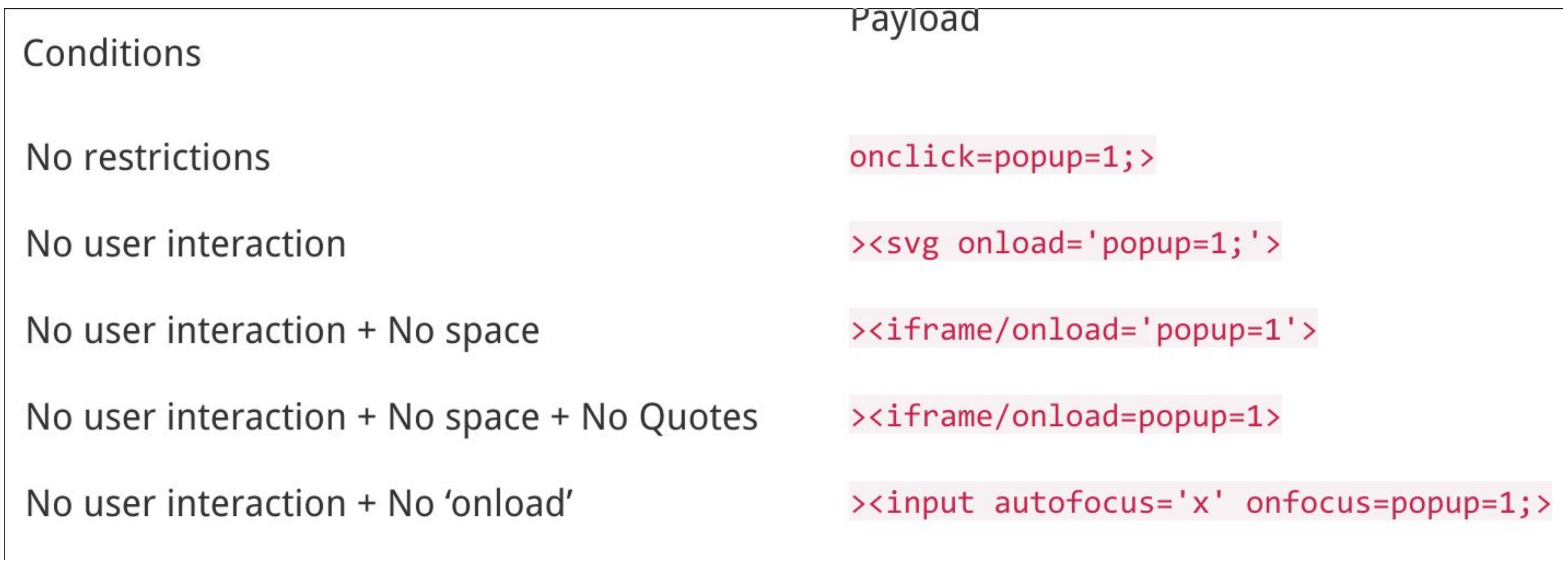
State Representation

- The only useful factors are the tag and the attribute context you are in
  - 1_tag: div
  - 1_tag_ap: class
  - context: attr_value
  - context: class

Enhancements

- To speedup the training time, instead of giving just alphabets I gave all the **html tags, attributes as actions**.
- Based on the html parsing, html tags are made available only when the context is a tag name etc..
- Similarly based on the html parsing attributes and their values are made available only when the context needs one of those.

Results - For <button INJECTION_POINT>



Conditions	Payload
No restrictions	`onclick=popup=1;>`
No user interaction	`><svg onload='popup=1;'>`
No user interaction + No space	`><iframe/onload='popup=1'>`
No user interaction + No space + No Quotes	`><iframe/onload=popup=1>`
No user interaction + No ‘onload’	`><input autofocus='x' onfocus=popup=1;>`

https://www.youtube.com/watch?v=7YUDa4nxX2I

9

Conclusion

Giving back is necessary!

What we looked at?

- Basics of machine learning and its types.
- Ways to make machine **identify inputs**, **understand application feedback**.
- Reinforcement learning and its use cases.
- Ways to **perform sequences** using rl.
- Vulnerability detection with the help of reinforcement learning.

Where to look for more?

Two nice approaches that I am aware of

- https://www.cloudsek.com/announcements/blog/cloud-ai-an-artificial-intelligence-on-the-cloud/
- https://www.slideshare.net/babaroa/code-blue-2016-method-of-detecting-vulnerability-in-web-apps

How to get started?

- https://bigml.com/ml101/
- https://kofzor.github.io/Reinforcement_Learning_101/
- http://karpathy.github.io/2015/05/21/rnn-effectiveness/
- https://github.com/cazala/synaptic/wiki/Neural-Networks-101
- https://www.coursera.org/learn/machine-learning
- http://scikit-learn.org/stable/tutorial/machine_learning_map/index.html
- https://docs.google.com/document/d/1custsRlKzNHRiY7P9S9khYqHc0PQSXEDTAXoJlbSl4U/edit?usp=sharing

Credits

- Special thanks to **all the people who trusted me and used the addon**.
- Kamaiah Nadavala
- Ajin Abraham
- Francis Alexander
- Yodlee Appsec Team!
- Others who are pursuing similar goals.

Thanks! And Questions?

Email: bharadwaj.machiraju@gmail.com

Twitter: @tunnelshade_

Blog: blog.tunneshade.in

Github: github.com/tunnelshade