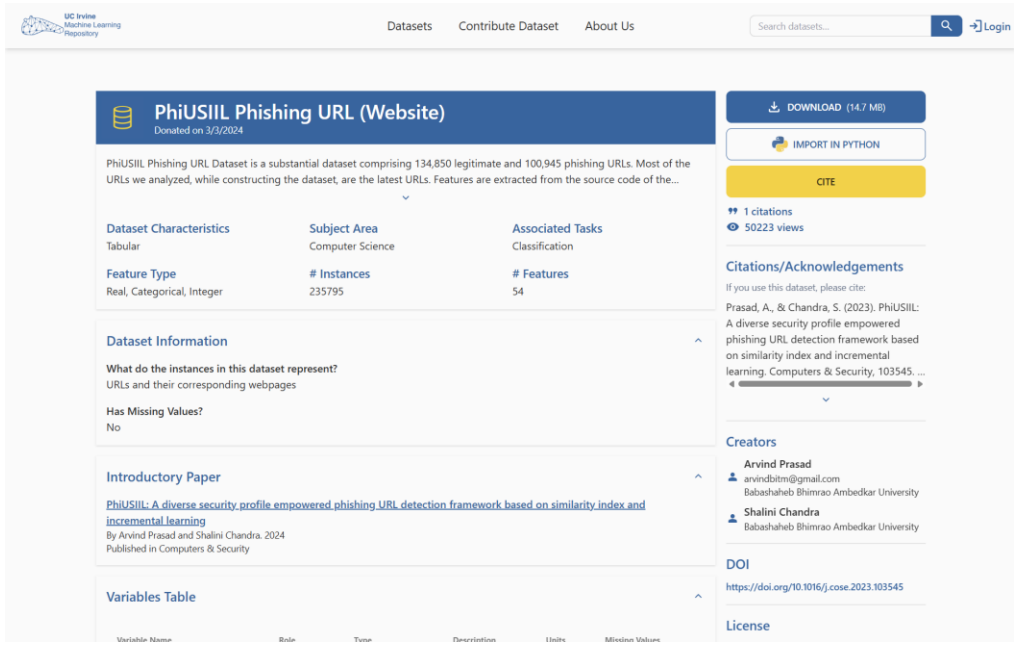
Data Science project about phishing websites and predictions

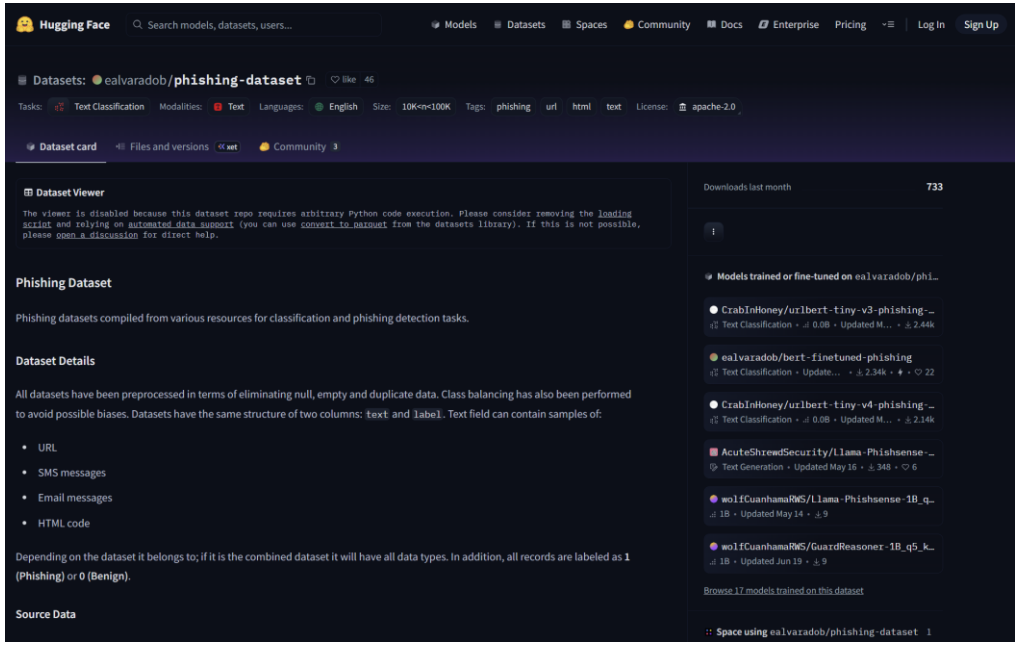
…so why this?

- …as most people know, most phishing sites nowadays uses suspicious TLDs and *typosquatting* techniques to conceal its domain
- Existing solutions (such as *Google Safe Browsing*) has well-maintained, up-to-date data about phishing domains, but it is not suitable for data training since there are no legitimate domain-only databases
  - Most of them requires online access for checking. This one uses a model that was built from multiple datasets containing phishing and legitimate domains

...and how about the data sources?

This project sources from these datasets:





PhiUSIIL

Cutoff: ~2022. Contains websites of organizations as well as personal websites, such as blogs for its legitimate data.

Phishing Dataset (from Hugging Face)

Cutoff: ~2024. Contains commonly used websites and other modern websites as its legitimate data. Also contains unusual TLDs for phishing data.

Data processing

```
['URL', 'URLLength', 'Domain', 'DomainLength', 'IsDomainIP', 'TLD',
       'URLSimilarityIndex', 'CharContinuationRate', 'TLDLegitimateProb',
       'URLCharProb', 'TLDLength', 'NoOfSubDomain', 'HasObfuscation',
       'NoOfObfuscatedChar', 'ObfuscationRatio', 'NoOfLettersInURL',
       'LetterRatioInURL', 'NoOfDegitsInURL', 'DegitRatioInURL',
       'NoOfEqualsInURL', 'NoOfQMarkInURL', 'NoOfAmpersandInURL',
       'NoOfOtherSpecialCharsInURL', 'SpacialCharRatioInURL', 'IsHTTPS',
       'LineOfCode', 'LargestLineLength', 'HasTitle', 'Title',
       'DomainTitleMatchScore', 'URLTitleMatchScore', 'HasFavicon', 'Robots',
       'IsResponsive', 'NoOfURLRedirect', 'NoOfSelfRedirect', 'HasDescription',
       'NoOfPopup', 'NoOfiFrame', 'HasExternalFormSubmit', 'HasSocialNet',
       'HasSubmitButton', 'HasHiddenFields', 'HasPasswordField', 'Bank', 'Pay',
       'Crypto', 'HasCopyrightInfo', 'NoOfImage', 'NoOfCSS', 'NoOfJS',
       'NoOfSelfRef', 'NoOfEmptyRef', 'NoOfExternalRef', 'label']
```

The columns from the original *PhiUSIIL* dataset were shown above. It contains most of the domain details, comparison (ratios), URL details, website characteristics, metadata, website category (bank, payment, cryptocurrency), number of elements, as well as the label.

Data processing

```
['URL', 'URLLength', 'Domain', 'DomainLength', 'IsDomainIP', 'TLD',
       'URLSimilarityIndex', 'CharContinuationRate', 'TLDLegitimateProb',
       'URLCharProb', 'TLDLength', 'NoOfSubDomain', 'HasObfuscation',
       'NoOfObfuscatedChar', 'ObfuscationRatio', 'NoOfLettersInURL',
      'LetterRatioInURL', 'NoOfDegitsInURL', 'DegitRatioInURL',
       'NoOfEqualsInURL', 'NoOfQMarkInURL', 'NoOfAmpersandInURL',
       'NoOfOtherSpecialCharsInURL', 'SpacialCharRatioInURL', 'IsHTTPS',
       'LineOfCode', 'LargestLineLength', 'HasTitle', 'Title',
       'DomainTitleMatchScore', 'URLTitleMatchScore', 'HasFavicon', 'Robots',
       'IsResponsive', 'NoOfURLRedirect', 'NoOfSelfRedirect', 'HasDescription',
       'NoOfPopup', 'NoOfiFrame', 'HasExternalFormSubmit', 'HasSocialNet',
       'HasSubmitButton', 'HasHiddenFields', 'HasPasswordField', 'Bank', 'Pay',
       'Crypto', 'HasCopyrightInfo', 'NoOfImage', 'NoOfCSS', 'NoOfJS',
       'NoOfSelfRef', 'NoOfEmptyRef', 'NoOfExternalRef', 'label']
```

Most features are removed since it is not important for parsing website (domain).

Data processing

`['URLLength',` URL length of the domain, which includes its protocol (http / https).

`'DomainLength',` Length of the domain (example: *google.com* => 10).

`'TLD',` Top level domain of the domain (example: *.com*, *.org*, *.net*, *.io*, etc.).

`'TLDLength',` Length of the top level domain (example: .com => 3).

`'NoOfSubDomain',` Number of subdomains in the URL (example: colab.research.*google.com* – 2 (separated by ".", length minus 2, which excludes *google.com*)).

`'LetterRatioInURL',` The percentage of how many letters are there in the URL, calculated with *SequenceMatcher*.

`'label']` Expected result (1: legitimate, 0: phishing).

Although most of them are selected by Claude due to me being too confused in the project *(and the only part from the project that utilizes AI for thinking)*, I understand why those are important – **they are mostly numerical values that can parse if a domain is phishing or not**.

Data processing

Example domain: `www.winchester.gov.uk`

Ratio (from PhiUSIIL dataset): `0.500`

Letter-only ratio without dots (expected result from SequenceMatcher): `0.923`

Ratio with protocol included, filtering only letters (expected result from SequenceMatcher): `0.885`

...but some features in this dataset are incorrect. There is no clue on how `LetterRatioInURL`'s ratio is calculated, so recalculation from the domain is required before selecting these features. Due to this, the model calculates data incorrectly – while See above for example. The recalculation process is done by replacing the ratio from a copy of the data frame.

Data processing

Domain (not included in dataset): `www.winchester.gov.uk`

`URLLength`	`30`	
`DomainLength`	`21`	
`TLD`	`1772`	`(uk)` TLD encoded with `LabelEncoder`
`TLDLength`	`2`	`(uk)`
`NoOfSubDomain`	`1`	`(www)`
`LetterRatioInURL`	`0.885`	
`label`	`1`	(Legitimate)

Data is combined and used for training. The processed data, including the reparsed letter ratio is then passed and filtered to a new dataset.

Data processing

Dataset Accuracy (PhiUSIIL only)

`0.845`

URL provided `www.winchester.gov.uk`

Result Legitimate

Accurate due to the website features being available in the dataset.

URL provided `accounts.google.com`

Result Phishing

Inaccurate due to non-similar website features from the dataset. This is a known legitimate website.

URL provided `www.suspicious-site.one`

Result Phishing

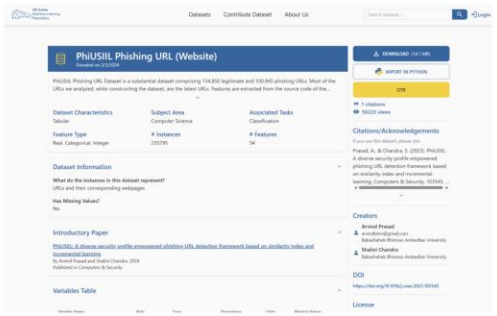
Accurate due to TLD only appearing on most phishing sites.

...but after training with RandomForest with only the PhiUSIIL dataset, the accuracy is alright, but with other foreign data, including known URLs, it is flagged as phishing because the features is inconsistent.

Data processing

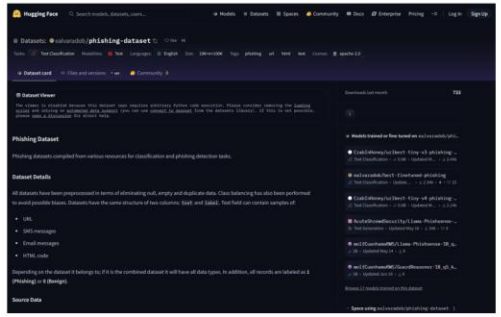
...and how about the data sources?

This project sources from these datasets:



PhiUSIIL

Cutoff: ~2022. Contains websites of organizations as well as personal websites, such as blogs for its legitimate data.

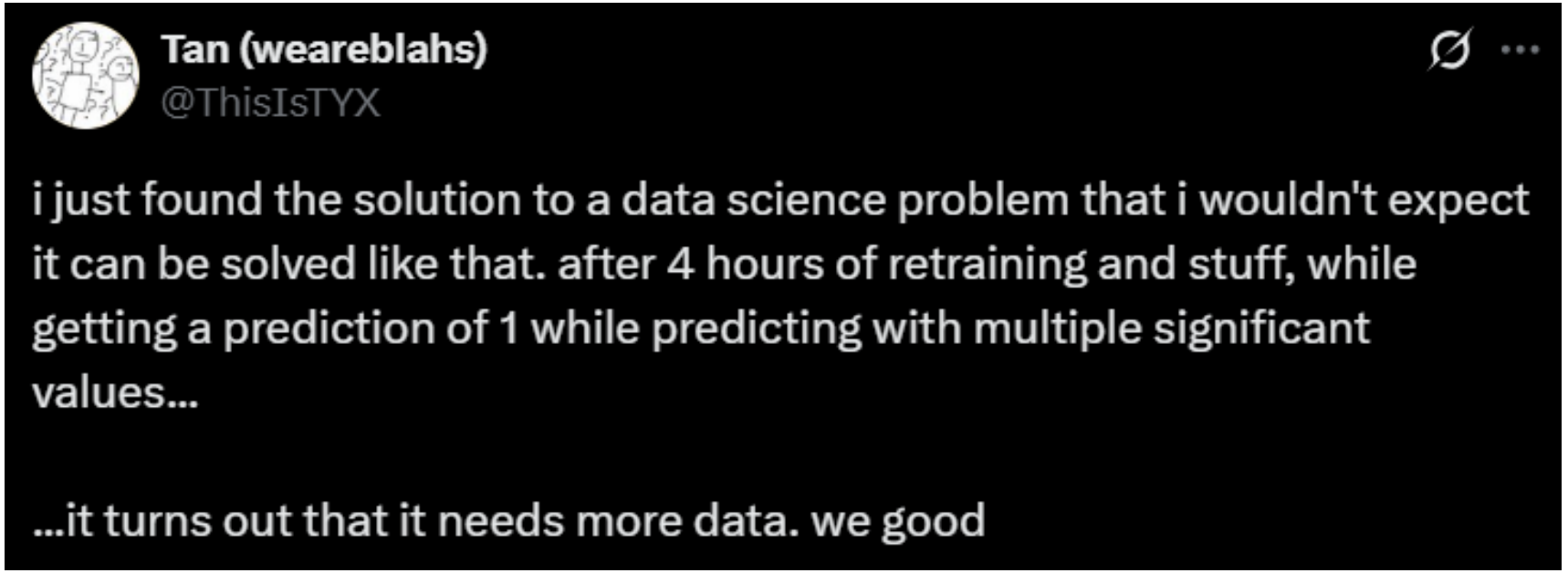


Phishing Dataset (from Hugging Face)

Cutoff: ~2024. Contains commonly used websites and other modern websites as its legitimate data. Also contains unusual TLDs for phishing data.

Remember this slide? This is how the issue is solved – combining two datasets into one model for training purposes.

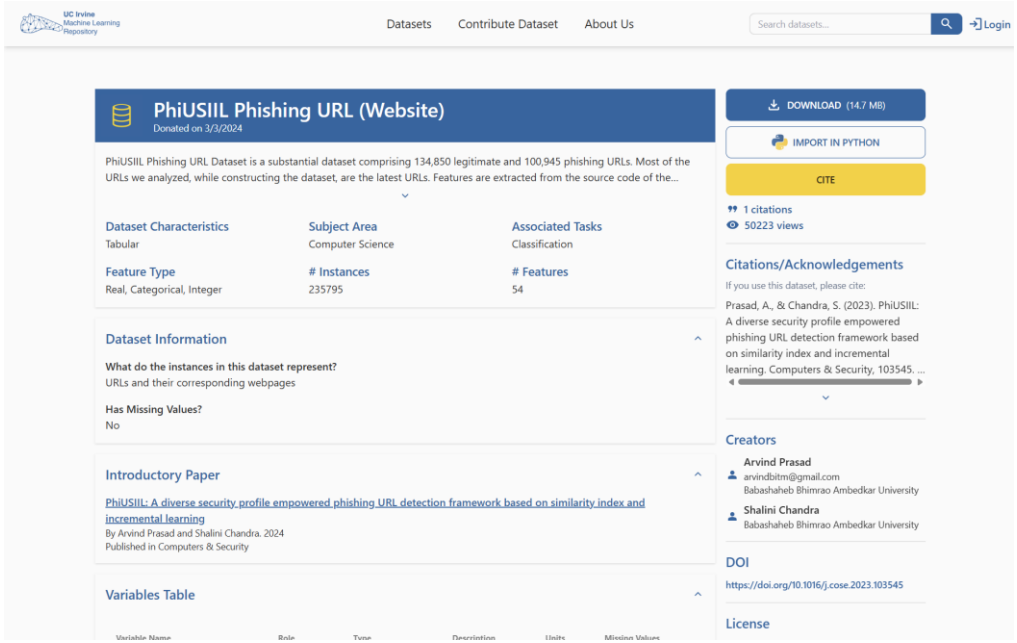
Data processing

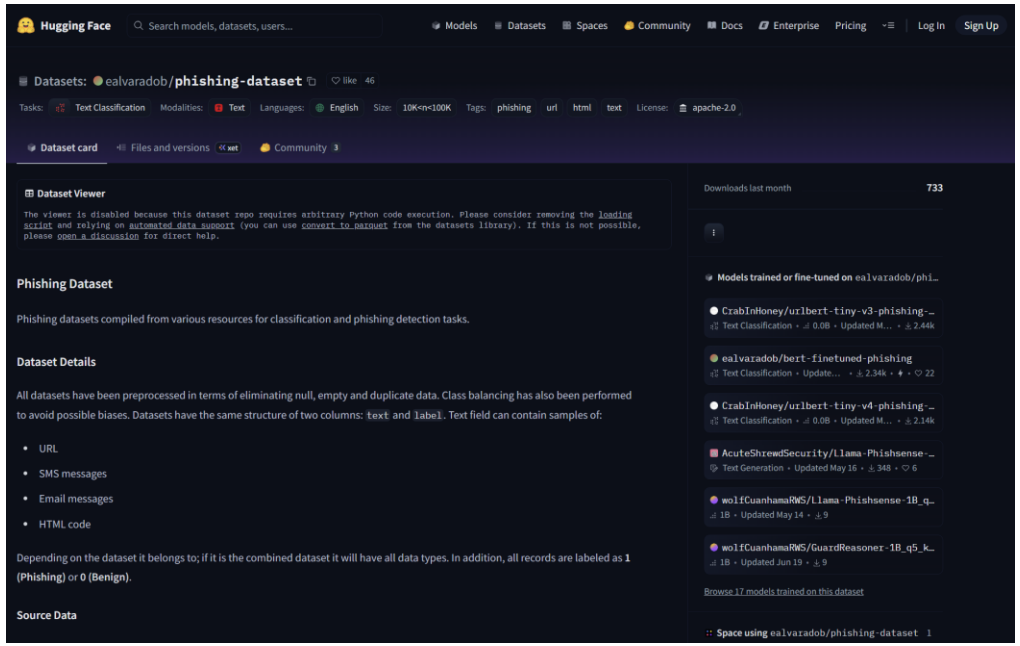


...well, for some reason, there is a tweet from me for (almost) every project I've done *just to remind me of what happened before and how did I solve it*.


Tan (weareblahs) @ThisIsTYX · Jun 6
me just now
Enable Sign in with Google
Enables Sign in with Google on the web using
Android apps or Chrome extensions.
Cyber Tea
电 脑 茶
./gradlew signingReport
CAN'T SIGN IN WITH GOOGLE THROUGH AN ANDROID APP?
GOOD
Use this command to get the fingerprint.
keytool -keystore path-to-debug-or-production-keystore
ALT

Data processing





PhiUSIIL

winchester.gov.uk, u.com.my, docs.google.com, aap.org, web.app, etc

Phishing Dataset (from Hugging Face)

youtube.com, google.com, github.com, linktr.ee, pages.dev, accounts.spotify.com, etc

Both datasets has unique URL values, as well as letter ratios. For the Hugging Face dataset, all the data is re-extracted from the domain. Above shows examples of websites from the respective datasets.

Data processing

Dataset Accuracy (PhiUSIIL and Hugging Face)
`0.858`

URL provided `www.winchester.gov.uk`

Result Legitimate

Accurate due to the website features being available in the dataset.

URL provided `accounts.google.com`

Result Legitimate

Accurate due to similar website features being available in the dataset.

URL provided `www.suspicious-site.one`

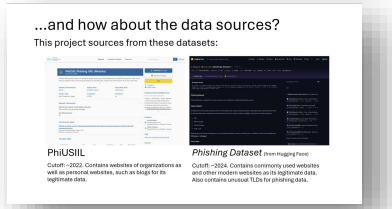
Result Phishing

Accurate due to this TLD being on most phishing sites. The prediction includes the letter ratio.

After training with the combined dataset with 651,971 rows of available data, the prediction is accurate. Additional domains for recent phishing sites, such as .cyou, .space, .tk are now available in the dataset thanks to additional data from the Hugging Face dataset.

Challenges

- Details extraction – the letter ratio of the project needs to be repaired since an unknown parsing method is used for the dataset, hence inaccurate results. Solved by recalculating letter ratio for the dataset.
- What significant data to extract? – Since the dataset has too much columns, it is confusing on what to extract. For the first few tries, with significant features, such as number of elements (divs, iframes, HTML, JavaScript, etc), the training resulted in overfitting (1.0 accuracy and 1.0 prediction no matter what).
  - There was a plan to use Selenium for parsing elements, but due to this issue, it is not used.
- The use of 2 datasets



Challenges

- Some websites has legitimate / illegitimate uses, but the dataset only contains phishing use. Examples:
  - **linktr.ee** – Common link shortener used in social media. Some bad actors uses Linktree links to “bypass” restrictions set in social media platforms. In recent times, the links are suspended by Linktree due to community guideline violations.
  - **docs.google.com** – Google Docs, mostly related to bad actors’ use of Google Forms for phishing-related forms.
  - **bit.ly** – common link shortener (Bitly). Some bad actors use it to shorten phishing links. Bitly does provide security (*Are you sure?* popups) for suspicious links.
  - **vercel.app**, **netlify.app** and **pages.dev** – all are default domains usually used for deployment services (Vercel, Netlify, Cloudflare Pages), but bad actors used it for phishing websites.

Demonstration

Thank You

Terima Kasih

謝謝

நன்றி