# Design Review

Proofpoint, Inc.

Aidan Cheng
Kevin Herrera
Carli Lessard
Vidushi Ojha

Advisor: Elizabeth Sweedyk
Liaisons: Thomas Lynam, Mike Morris

# What is the Proofpoint team doing in their corner?

- Tokenization
- Naive Bayes Classifier
- Support Vector Machine Classifier
- Deep Learning

Using tools such as:

- Python, JSON, scikit-learn, open-source code, TensorFlow

# Example of Data Entry

```
{ u'queue_ts': datetime.datetime(2016, 8, 1, 0, 56, 37, 485000),
  u'url': u'http://www.korcham.net/image/templet/images/mail_2015_06_bg.gif)',
  u'recv_ts': datetime.datetime(2016, 8, 1, 0, 41, 4),
  u'misc': { u'content': None,
             u'ip': u'218.55.99.171',
             u'num_messages': 2,
             u'uuids': [ u'dceefc3a7c101be7cf30dc1dd9ac4346',
                         u'd4f7ebdc6fad85bcbb7678a3dc15bbb9' ],
             u'subject':
u'=?euc-kr?B?KLGksO0pIFu068fRu/Ow+Mi4wMe80l0gJ7nMsbnAxyC89sDUIA==?=\n
=?euc-kr?B?sdTBpiCwrcitILW/x+Kw+iC068DAIMD8t6sgvLy5zLOqJyC+yLO7?=' },
  u'results': { u'normalized_forensics': {},
                u'details': None,
                u'scanid': u'3096224878164377',
                u'forensics_score': 0,
                u'result': u'clean' },
  u'source': u'spam',
  u'state': u'scanned',
  u'scan_ts': datetime.datetime(2016, 8, 1, 0, 57, 19, 458000),
  u'_id': ObjectId('579e9e45d46e833243494a94'),
  u'type': u'unknown' }
```

# Example of Data Entry

```
{ u'queue_ts': datetime.datetime(2016, 8, 1, 0, 56, 37, 485000),
  u'url': u'http://www.korcham.net/image/templet/images/mail_2015_06_bg.gif)',
  u'recv_ts': datetime.datetime(2016, 8, 1, 0, 41, 4),
  u'misc': { u'content': None,
             u'ip': u'218.55.99.171',
             u'num_messages': 2,
             u'uuids': [ u'dceefc3a7c101be7cf30dc1dd9ac4346',
                         u'd4f7ebdc6fad85bcbb7678a3dc15bbb9' ],
             u'subject':
u'=?euc-kr?B?KLGksO0pIFu068fRu/Ow+Mi4wMe80l0gJ7nMsbnAxyC89sDUIA==?=\n
=?euc-kr?B?sdTBpiCwrcitILW/x+Kw+iC068DAIMD8t6sgvLy5zLOqJyC+yLO7?=' },
  u'results': { u'normalized_forensics': {},
                u'details': None,
                u'scanid': u'3096224878164377',
                u'forensics_score': 0,
                u'result': u'clean' },
  u'source': u'spam',
  u'state': u'scanned',
  u'scan_ts': datetime.datetime(2016, 8, 1, 0, 57, 19, 458000),
  u'_id': ObjectId('579e9e45d46e833243494a94'),
  u'type': u'unknown' }
```

# Metadata

- URL
- IP address
- Subject
- Date/Time
- Number messages

# Feature Generation

- Tokenization
  - Parse data entries to only obtain ‘url’ and ‘result’
  - Parse (tokenize) url by ‘/’ or ‘.’ to create a collection of tokens (words or character strings)

    “http://www.korcham.net/image/templet/images/mail_2015_06_bg.gif” will be tokenized as

    “http:” “www” “korcham” “net” “image” “templet” “images” “mail_2015_06_bg” “gif”
  - Every single unique token gets its own feature, many of our vectors are 10,000+ in length
- Automatic or manual?
  - Could select some seemingly relevant features (e.g. tokenization)
  - Or let an algorithm generate them for us
- Dimension reduction
  - If automatic generation, how to keep within reasonable limits?

# Types of Classifiers?

- Currently using
  - Naive Bayes
  - Deep Learning via TensorFlow
- Tried SVM, but too slow
- Any other machine learning methods that could be useful?