Email ID: 5700 (UNCLASSIFIED)
From: melissa.becker@enron.com
To: kent.castleman@enron.com, sally.beck@enron.com, howard.selzer@enron.com,
Date: Tue, 16 Jan 2001 07:34:00 -0800 (PST)
Subject: Merit - Action requested

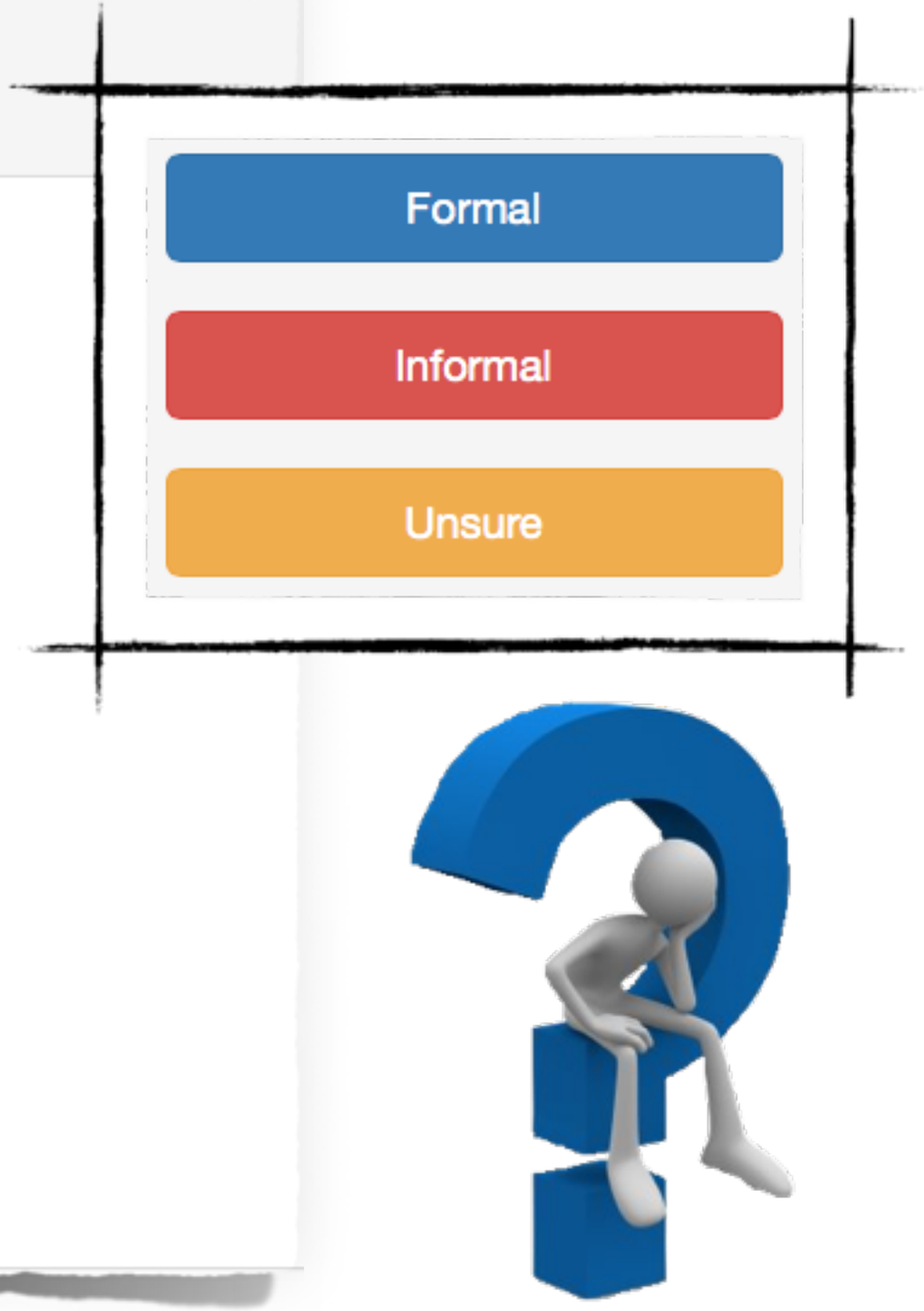
Rick has asked that I help him ensure that we all finish up our merit increases in time for the new deadline of this Friday, January 19.

1. Could you please "submit" by the end of the day tomorrow, Wednesday, January 17? Then Rick and I can review and get back with you with any changes.

2. If you could leave me a voice mail or eMail when you are finished, I would appreciate it. I am moving back to the Enron building and will get back my old number (X36641) effective tomorrow.

3. In addition, we are interested in your thoughts on the merit process overall, both for yourself and for your groups. In general, it seems like a lot of "administration" for some pretty small numbers. What are your thoughts - do you like the current process or would less frequent, larger increases be preferable (assuming we could make that change in the larger overall Enron framework)?

Please call me or Andrea Yowman (x31477) with any questions. Thanks for your help!



Email Formality Classification

CNIT49900NLT

Group Members

- Dan O’Day
- Robert Hinh
- Sangmi Shin
- Upasita Jain
- Penghao Wang

Introduction

- Classify emails as either formal or informal
- Framework developed for feature extraction
- Web application developed for classification and framework validation

Enron Email Corpus

cs.cmu.edu

Enron Email Dataset

This dataset was collected and prepared by the CALO Project (A Cognitive Assistant that Learns and Organizes). It contains data from about 150 users, mostly senior management of Enron, organized into folders. The corpus contains a total of about 0.5M messages. This data was originally made public, and posted to the web, by the Federal Energy Regulatory Commission during its investigation.

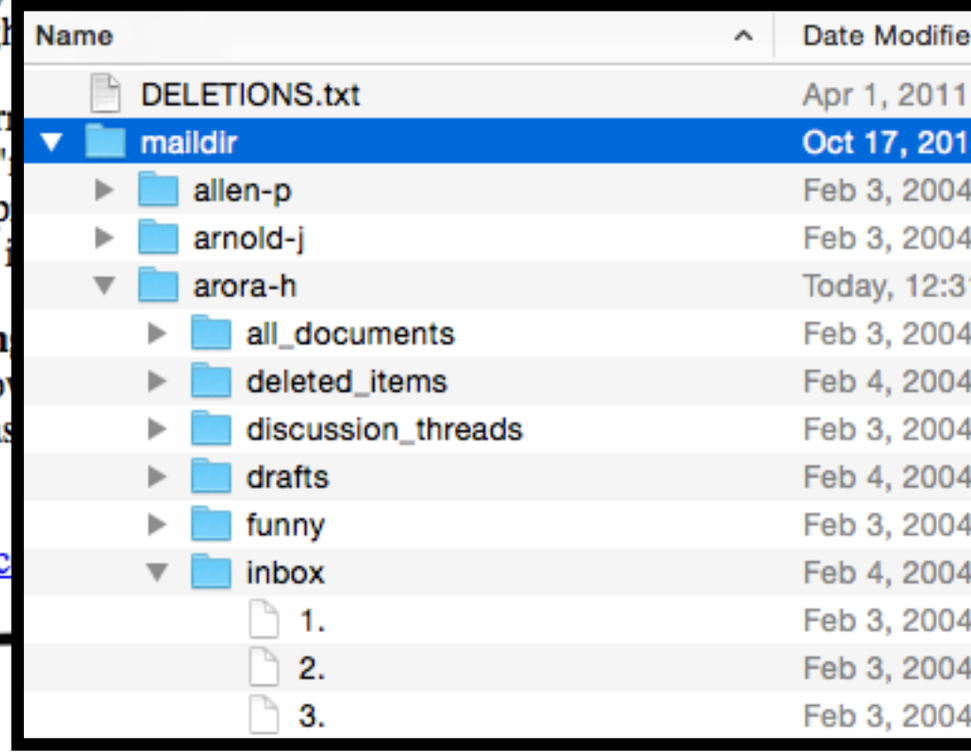
The email dataset was later purchased by Leslie Kaelbling at MIT, and turned out to have a number of integrity problems. A number of folks at SRI, notably Melinda Gervasio, worked hard to correct these problems, and it is thanks to them (not me) that the dataset is available. The dataset here does not include attachments, and some messages have been deleted "as part of a redaction effort due to requests from affected employees". Invalid email addresses were converted to something of the form user@enron.com whenever possible (i.e., recipient is specified in some parse-able format like "Doe, John" or "Mary K. Smith") and to no_address@enron.com when no recipient was specified.

I get a number of questions about this corpus each week, which I am unable to answer, mostly because they deal with preparation issues and such that I just don't know about. If you ask me a question and I don't answer, please don't feel sligh

I am distributing this dataset as a resource for researchers who are interested in improving curr currently used. This data is valuable; to my knowledge it is the only substantial collection of " are not public is because of privacy concerns. In using this dataset, please be sensitive to the p many of these people were certainly not involved in any of the actions which precipitated the i

- ~~March 2, 2004 Version of dataset~~ and the ~~August 21, 2009 Version of dataset~~ are **no lon** for your work, you are requested to replace it with the newer version of the dataset belo copy. A total of four messages have been removed since the original version of the datas
- August 21, 2009 Version of dataset (about 423Mb, tarred and gzipped).

There are also at least two on-line databases that allow you to search the data, at Enronemail.c



CNIT 499NLT Web App

Web application for email classification and framework verification.

This is a group project for CNIT499NLT Natural Language Technologies at Purdue University. The course instructor is Dr. Julia Taylor.

This project uses the Enron email corpus, retrieved from https://www.cs.cmu.edu/~./enron/.

The complete project and corresponding code are available on Github.

Group Members

Dan O'Day
Robert Hinh
Upasita Jain
Sangmi Shin
Penghao Wang

danzek / **email-formality-detection**

Unwatch 3 | Star 0 | Fork 0

Determine whether an email is formally or informally written — Edit

52 commits | **3** branches | **0** releases | **1** contributor

branch: **mysql** | **email-formality-detection** / +

removed partial unnecessary function

danzek authored 14 minutes ago — latest commit `343732fd2a`

classifier_app	add features	a day ago
data	add features	a day ago
features	removed partial unnecessary function	14 minutes ago
.gitignore	add unigram and bigram features	3 hours ago
LICENSE	Added MIT License text	24 days ago
README.md	Update README.md	16 minutes ago
get_email.py	add features	a day ago
process_corpus.py	removed/commented out bigram extraction	20 minutes ago

HTTPS clone URL

https://github.com/

You can clone with HTTPS, SSH, or Subversion.

```
@data
{0 640,1 71,2 40.11,3 177,5 0.56,7 5,8 2,10 1,14 0.03,17 1,18 1}
{0 423,1 41,2 36.61,3 112,7 5,8 2,10 1,14 0.02,17 1,18 1}
{0 534,1 65,2 38.92,3 167,7 3,8 2,10 1,17 1,18 1}
{0 3841,1 142,2 18.66,3 761,5 0.26,7 2,8 2,10 1,20 1}
{0 486,1 20,2 15.27,3 131,4 1,5 1.53,7 2,8 1,10 1,13 1,17 1,18 1,20 1}
{0 142,1 12,2 29.27,3 41,7 5,8 3,10 1,13 1}
{0 614,1 21,2 14,3 150,5 0.67,7 3,8 4,10 3,14 0.01}
{0 734,1 45,2 33.09,3 136,7 2,8 2,10 1,13 1,17 1,18 1,20 1}
{0 56,1 2,2 14.29,3 14,7 2,8 1,10 1}
{0 1141,1 46,2 17.42,3 264,5 0.76,7 3,8 1,10 3,20 1}
{0 961,1 39,2 19.6,3 199,7 5,8 2,10 3,13 1,14 0.01,20 1}
{0 111,1 7,2 26.92,3 26,7 2,10 1}
{0 390,1 16,2 17.39,3 92,7 4,8 1,10 1,20 1}
{0 439,1 20,2 23.53,3 85,5 1.18,7 3,8 2,10 1,20 1}
{0 1296,1 65,2 23.38,3 278,7 4,10 3,14 0.02,20 1}
{0 74,1 6,2 37.5,3 16,7 2,8 2,10 3,20 1}
{0 132,1 12,2 29.27,3 41,4 1,7 5,8 3,10 1,14 0.09,16 1,19 1}
{0 112,1 10,2 34.48,3 29,7 2,8 2,10 2}
{0 105,1 13,2 50,3 26,7 1,8 1,10 3,20 1}
{0 72,1 9,2 42.86,3 21,4 1,7 5,10 1,14 0.11}
{0 116,1 14,2 51.85,3 27,7 3,8 1,10 3,14 0.09,20 1}
{0 1402,1 102,2 40.96,3 249,5 0.4,7 4,10 1,14 0.02,17 1,18 1}
{0 132,1 6,2 20,3 30,7 2,8 2,10 1}
{0 232,1 11,2 19.3,3 57,7 1,10 1,14 0.01,20 1}
{0 102,1 8,2 42.11,3 19,7 1,8 2,10 1,12 25,14 0.08}
{0 236,1 11,2 20,3 55,4 1,7 2,8 1,10 1,14 0.01}
{0 88,1 8,2 42.11,3 19,4 1,7 4,10 1,14 0.02}
{0 126,1 10,2 34.48,3 29,7 1,8 1,10 1,14 0.03,17 1,18 1}
{0 48,1 7,2 58.33,3 12,7 5,8 2,10 1,14 0.32}
{0 220,1 7,2 15.56,3 45,5 2.22,7 4,10 1,20 1}
{0 447,1 17,2 17.71,3 96,5 1.04,7 2,8 1,10 1,14 0.01,17 1,18 1,20 1}
{0 24,1 3,2 42.86,3 7,7 2,8 2,10 1,20 1}
{0 95,1 8,2 32,3 25,5 8,7 2,8 4,10 2}
{0 755,1 33,2 20.5,3 161,7 3,8 3,10 1,14 0.04,20 1}
{0 261,1 14,2 22.58,3 62,7 5,8 3,10 3,20 1}
{0 114,1 8,2 32,3 25,7 1,8 3,10 3}
{0 205,1 20,2 40.82,3 49,7 2,8 5,10 1}
{0 1476,1 56,2 16.77,3 334,5 0.6,7 2,10 1,11 0.07,12 13.3,13 2,14 0.01,20 1}
{0 363,1 31,2 38.75,3 80,7 4,8 1,10 1,14 0.04,20 1}
{0 60,1 5,2 38.46,3 13,7 3,10 3,14 0.05,20 1}
{0 313,1 13,2 15.85,3 82,7 5,10 1,11 0.14,13 1,14 0.01}
{0 1919,1 725,2 78.63,3 922,4 1,5 0.11,7 2,8 1,10 100,12 6.7,14 0.16,20 1}
{0 538,1 29,2 21.01,3 138,7 3,8 2,10 1,14 0.01,20 1}
{0 1402,1 102,2 40.96,3 249,5 0.4,7 4,10 1,14 0.02,17 1,18 1}
{0 16,1 3,2 60,3 5,4 1,7 1,10 1,14 0.2}
```

```
ite_libsvm_file(wf, all=True):
h open('features.libsvm', 'w') as ff:
  c = Corpus()

  if all:
      email_generator = c.fetch_all_emails()
  else:
      email_generator = c.fetch_all_emails(colu

  for email in email_generator:
      email.get_current_message()  # make sure
      feature_dictionary, classifier_to_write_t

      # write feature set for this sample to fi
      string_builder = ""
      string_builder += classifier_to_write_to_
      for f in email.feature_set.items():
          string_builder += " %s:%s" % f

      # ff.write(" # email id: " + str(email.id
      try:
          ff.write(string_builder + '\n')
      except IOError:
          pass
```

```
class Corpus():
    """Corpus object.

        cnx -- mySQL connection object
        cur -- mySQL cursor object
    """
    def __init__(self):
        self.DB_PASSWORD = 'haha_no_not_on_github_:)'
        self.conn = MySQLdb.connect('mysql.server', 'cn

    def build_sqlite_corpus(self, path_to_corpus):
        """Given the path to the unpacked corpus as ava
        into SQLite.

        Arguments:
            path_to_corpus -- full path to unpacked Enr
        """
        exclusion_set = set(['contacts', 'calendar'])
        for root, subdirs, emails in os.walk(path_to_co
            print 'Parsing ' + root + ' folder.'
            try:
                for email in emails:
                    e = Email(self)
                    e.extract_fields(os.path.join(root,
            except OSError:
                print '\tOSError while processing ' + r

            subdirs[:] = [d for d in subdirs if d not i

        print 'Finished creating SQLite corpus.'
```

Methodology

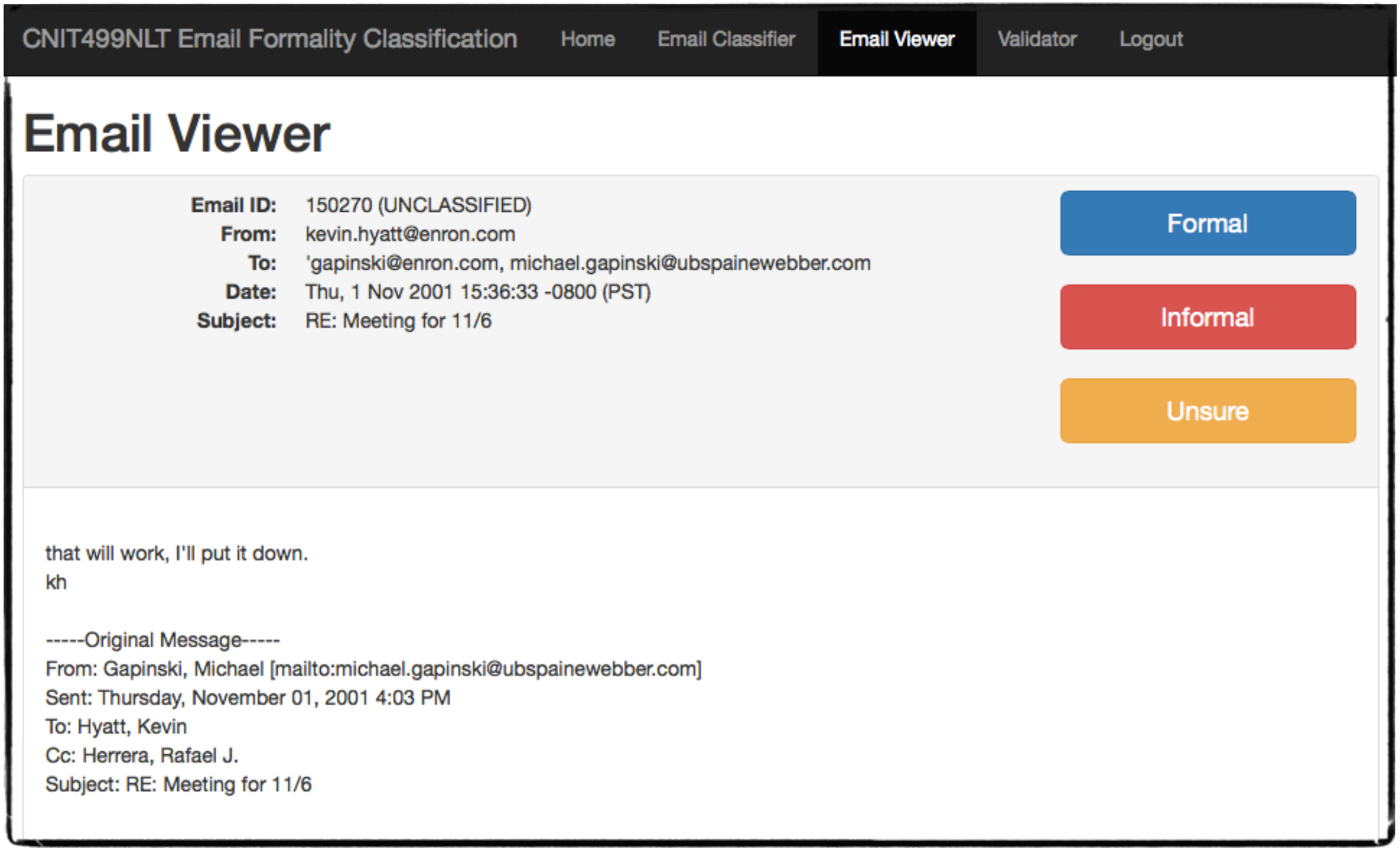
Methodology

- Write emails to database
- Develop feature extraction framework
    - Determine current message
    - Validate
- Classify emails as formal or informal

NL Technologies & Techniques Applied

- Sentence boundary disambiguation
- POS tagging
- Stylometric features deemed relevant
  - Use of capitalization
  - Spelling
  - Vocabulary
- Heuristic features

Classification



Validation

CNIT499NLT Email Formality Classification | Home | Email Classifier | Email Viewer | **Validator** | Logout

Email Validator

The email on the left is the original message. The email on the right is the same email after running the get_current_message() method. The goal of this method is to eliminate forwarded messages and other previous messages still existing in the email body so that feature extraction only takes place on the *current* message. Please determine whether or not the email was correctly rendered by this method. If there is no previous message, then the email should be the same on both sides—that would be correct. However, if the email is a forward or reply, the original message should be removed from the email body on the right.

Email ID: 414199 (UNCLASSIFIED)
From: jeffrey.shankman@enron.com
To: jennifer.burns@enron.com
Date: Sun, 17 Dec 2000 23:58:00 -0800 (PST)
Subject: Refined Products Line--North American Markets - CERA Alert: Decem

Correct

Incorrect

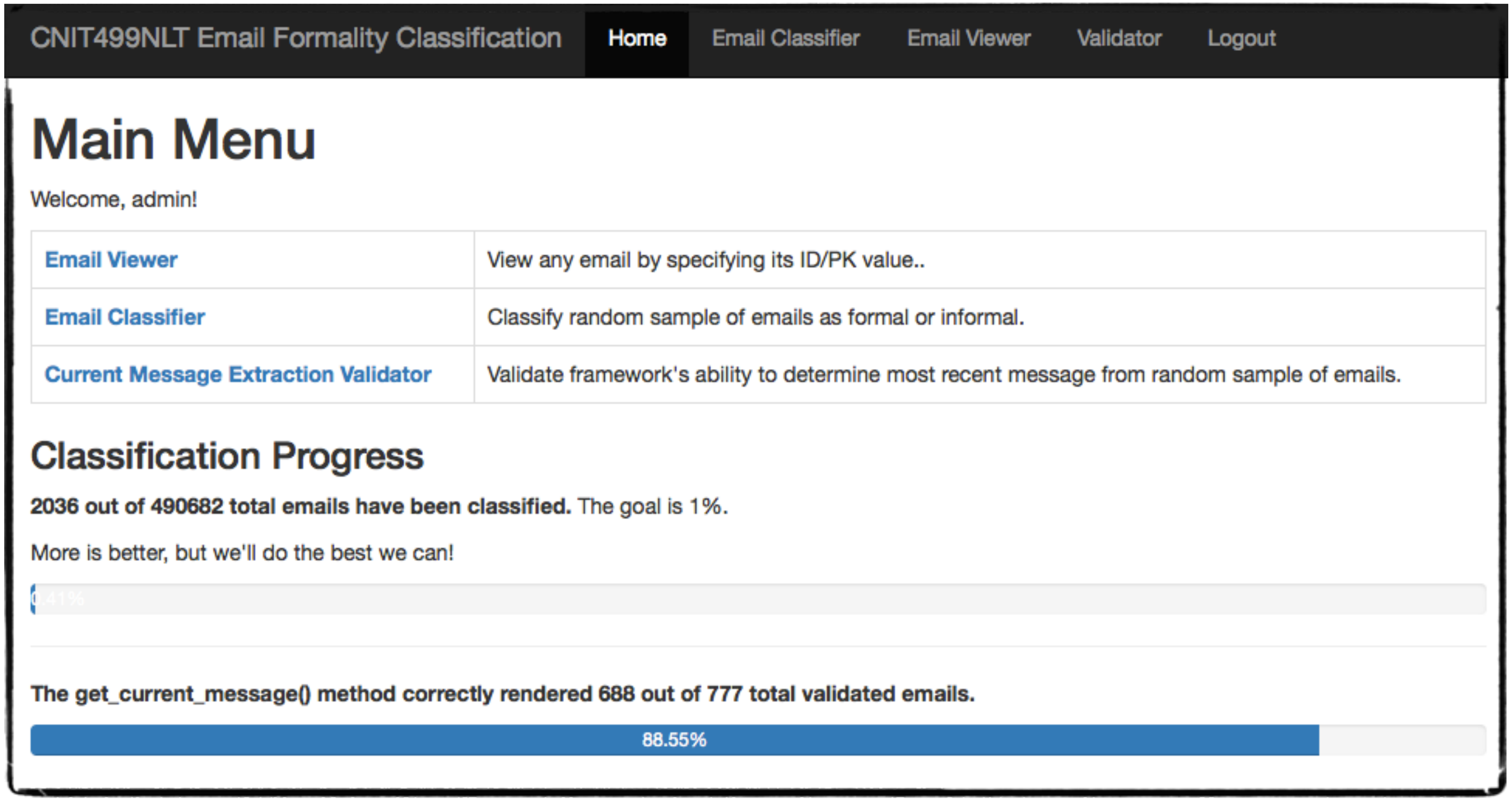
print

---------- Forwarded by Jeffrey A Shankman/HOU/ECT
on 12/18/2000
08:03 AM ----------

From: Doug Leach 12/18/2000 07:18 AM

print

LOTS of Samples!



Why were so few classified?

SQLite

MySQL

Training the Model

- Feature extraction
- Used 10% of classified data for training (random sampling, no SMOTE)
- 10-fold cross-validation also conducted (did not significantly alter results)
- Attempted different algorithms and compared results

Limitations

- More complex features needed
- Numerous problems inherent in this task
  - Subjectivity of formal vs. informal
  - Ability to detect greetings and closings
  - Improve ability to detect most current message

Limitations

- Ability to detect email signatures
    - Unable to implement for this project
    - How to detect signature?
        - Last ten lines?
        - Signature block near reply line?
        - Always includes author’s name?

Cohen, W. (2004). *Learning to Extract Signature and Reply Lines from Email.*

Limitations

- How to set rules for email signature detection
- Separate machine learning problem tackled by others
  - Position feature
  - Person’s name feature
  - Ending feature
  - Phone and/or fax # feature
  - Email address feature
  - Sequence of these features

Cohen, W. (2004). *Learning to Extract Signature and Reply Lines from Email.*

Features

Simple Counts

- # Characters
- # Syllables
- Average # Syllables per Word
    - Uses Carnegie Mellon Pronouncing Dictionary corpus
- # Words
- # Verbs

Heuristic Counts & Ratios

- Net Lingo Count
  - “LOL”
  - Net Lingo (http://www.netlingo.com)
    - Top 50 Most Popular Text Terms used in Business
    - Top 50 Most Popular Text Terms

Heuristic Counts & Ratios

- Closing Statement
  - Formality of email contents
- Implementation of the feature:
  - Parameter of closing statement - True / False

Heuristic Counts & Ratios

- Recipients Count
  - One-to-one vs. one-to-many emails
  - One-to-many emails tend to be more formal than one-to-one emails

Politeness indicators per message	
one-to-many native	3.22
one-to-one native	2.28
one-to-many non native	1.09
one-to-one non-native	1.31

Contractions			
	Possible contractions	Full forms	Contractions
one-to-many native	116	115 (99.13%)	1 (0.87%)
one-to-one native	111	109 (98.19%)	2 (1.81%)
one-to-many non-native	47	42 (89.36%)	5 (10.64%)
one-to-one non-native	79	72 (91.13%)	7 (8.87%)

Sabater, Carmen Pérez, Ed Turney, and María Begoña Montero Fleta. "Orality and literacy, formality and informality in email communication." *Ibérica: Revista de la Asociación Europea de Lenguas para Fines Específicos (AELFE)* 15 (2008): 71-88, tables 3 & 4.

Use of Capitalization

- Ratio of Sentences Not Beginning With Capital Letters
  - ‘hello there. what’s up?’
  - Account for exceptions (’Mr.’, ‘i.e.’, ‘e.g.’, ‘viz.’) to properly identify sentence boundaries

Use of Capitalization

- Ratio of Contiguous Capital Letters to Total Letters
  - “Please DO NOT go there!”
  - ‘HOW ARE YOU? WASSUP?’
- Capitalization of First Person Singular Pronoun (“I”)
  - “sue and i can’t wait to leave!”

Spelling

- Misspelled Words
  - “Did you recieve my email?”
  - (http://www.oxforddictionaries.com/words/common-misspellings)

Correct Spelling	Incorrect Spelling
Accommodate	Accomodate
Achieve	Acheive
Across	Accross

Email Metadata

- Date/Time
  - Day
  - Time Category
    - Weekend
- Forward or Reply (boolean & counts)

Additional Grammatical Features

- Use of Contractions Ratio
  - “Won’t you get that for me? Can’t you?”
- Excessive Punctuation Ratio
  - “Let’s get ‘er done!!!!!!!!!!!”

Words as Features

- First trained model only with heuristic features
- Then attempted unigrams and bigrams, but too computationally expensive
  - Could not afford more Amazon Web Services CPU hours
  - File system limitations prevented writing large feature files required + consequent database connection issues

Words as Features

- Next attempted only unigrams with length > 2
    - Not ideal, but allowed us to work within file system limitations (vfat restricts file size to 4GB)


Results

Validation

CNIT499NLT Email Formality Classification | Home | Email Classifier | Email Viewer | **Validator** | Logout

Email Validator

The email on the left is the original message. The email on the right is the same email after running the get_current_message() method. The goal of this method is to eliminate forwarded messages and other previous messages still existing in the email body so that feature extraction only takes place on the *current* message. Please determine whether or not the email was correctly rendered by this method. If there is no previous message, then the email should be the same on both sides—that would be correct. However, if the email is a forward or reply, the original message should be removed from the email body on the right.

Email ID: 414199 (UNCLASSIFIED)
From: jeffrey.shankman@enron.com
To: jennifer.burns@enron.com
Date: Sun, 17 Dec 2000 23:58:00 -0800 (PST)
Subject: Refined Products Line--North American Markets - CERA Alert: Decem

Correct

Incorrect

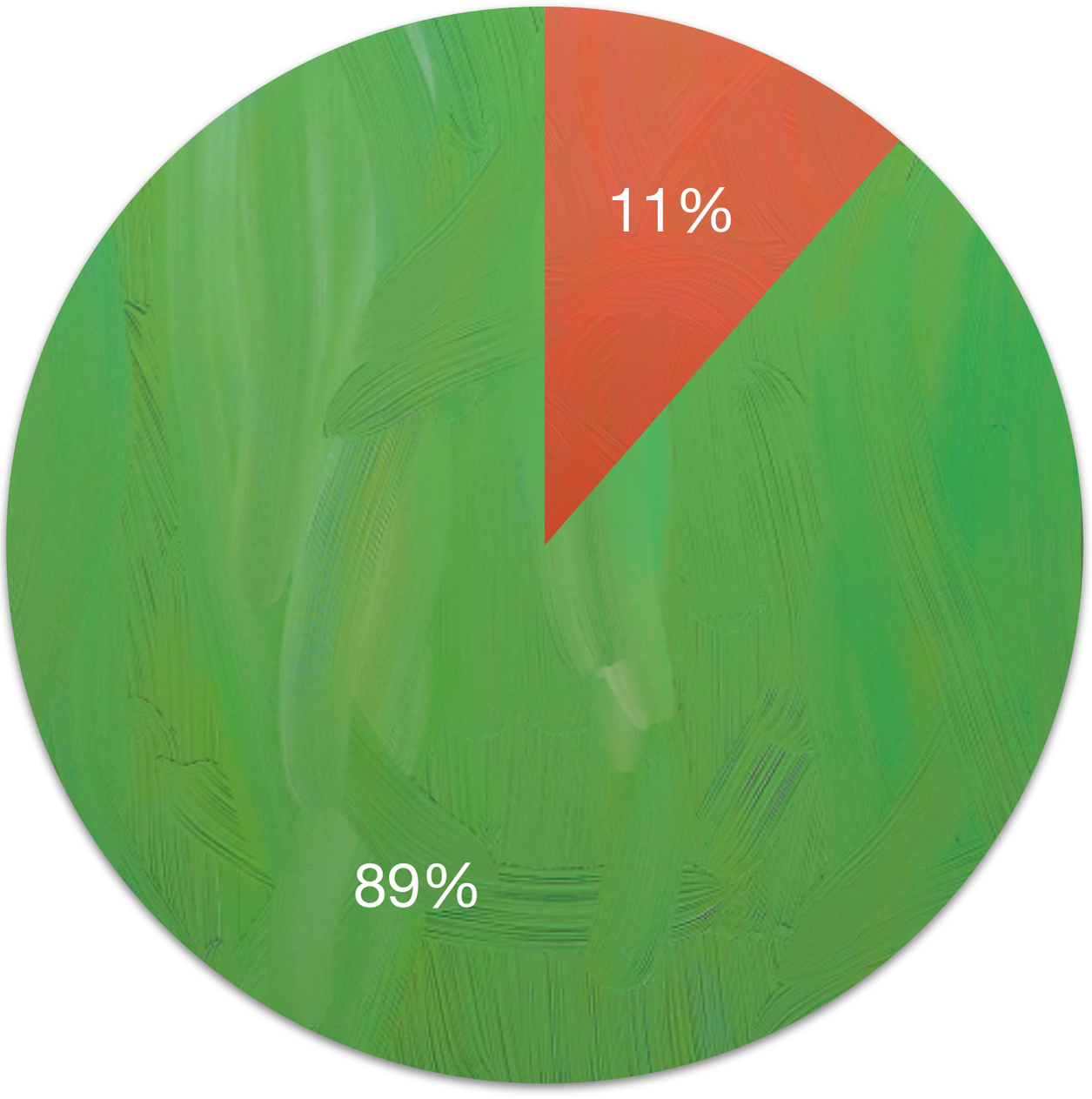
print

---------- Forwarded by Jeffrey A Shankman/HOU/ECT
on 12/18/2000
08:03 AM ----------

From: Doug Leach 12/18/2000 07:18 AM

print

Get Current Message Validation



The get_current_message() method correctly rendered 688 out of 777 total validated emails.

88.55%

Machine Learning Algorithms

- Naive Bayes
- Linear Discriminant Analysis
- C5.0 Decision Trees
- kNN

Machine Learning

- First ran results *without* words as features
- Then ran again *with* words as features

Naive Bayes

63.55% correctly classified

Classified Data

NB Prediction	Informal	Formal	Sum
Informal	596	256	852
Formal	486	698	1184
Sum	1082	954	

Linear Discriminant Analysis

66.7% correctly classified

LDA Prediction	Classified Data: Informal	Classified Data: Formal	Classified Data: Sum
Informal	750	346	1096
Formal	332	608	940
Sum	1082	954	

C5.0 Decision Trees

70.33% correctly classified

DT Prediction	Classified Data: Informal	Classified Data: Formal	Sum
Informal	834	248	1082
Formal	356	598	954
Sum	1190	846	

C5.0 Decision Trees

- Rules chosen overfit the training samples

```
if (Character_Count <= 101) {
   Informal;
}
```

```
if (Character_Count > 101) {
   if (!Is_Reply) {
      Informal;
   } else {
      if (Syllable_Count <= 8) {
         Informal;
      } else {
         Formal;
      }
   }
}
```

kNN, k=1

64.29% correctly classified

Classified Data

k=1 Prediction	Informal	Formal	Sum
Informal	719	364	1083
Formal	363	590	953
Sum	1082	954	

kNN, k=3

62.52% correctly classified

Classified Data

k=3 Prediction	Informal	Formal	Sum
Informal	686	367	1053
Formal	396	587	983
Sum	1079	954	

kNN, k=5

64.04% correctly classified

Classified Data

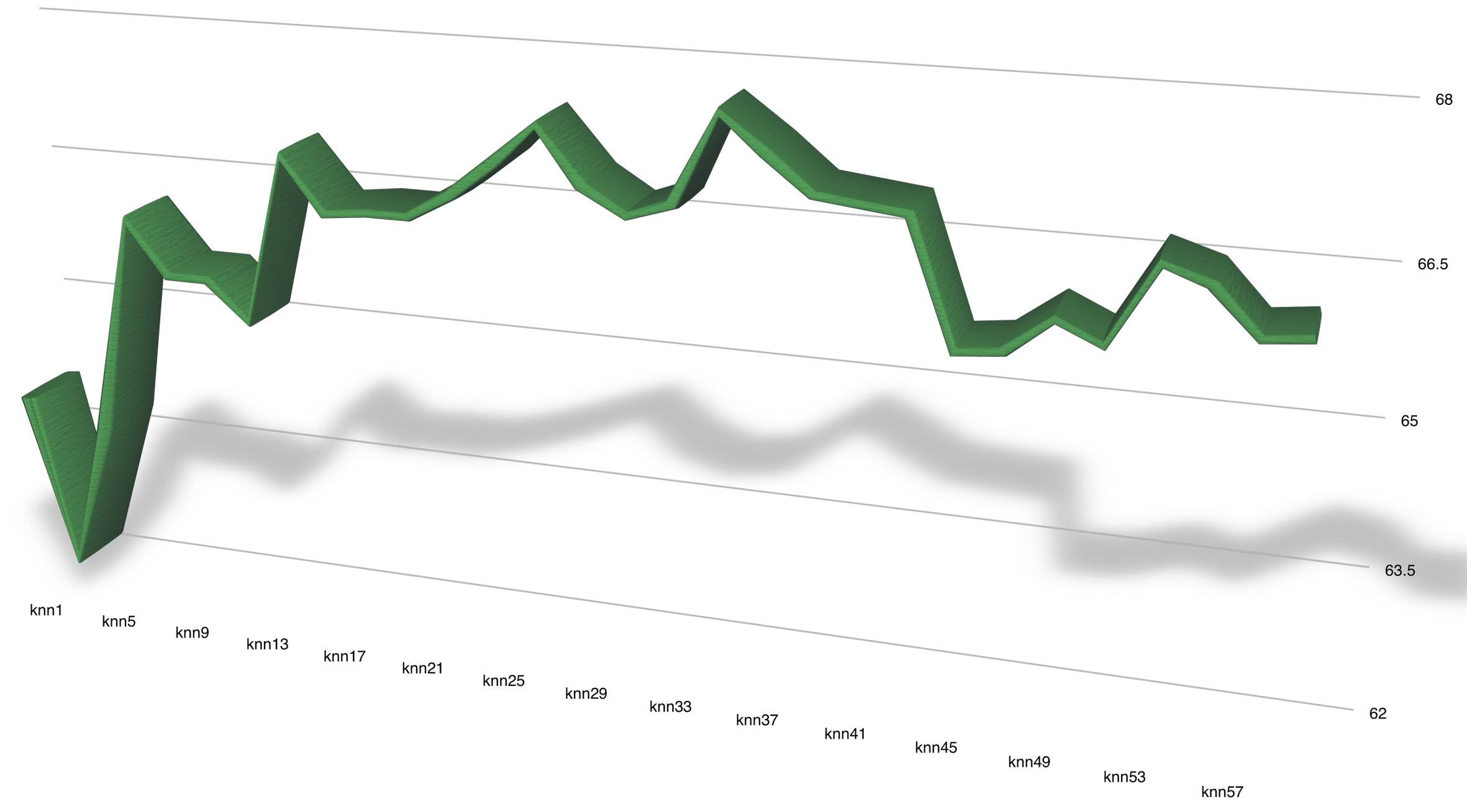
k=5 Prediction	Informal	Formal	Sum
Informal	711	361	1072
Formal	371	593	964
Sum	1082	954	

	k values	trial1	trial2	trial3	trial4	trial5	trial6	trial7	trial8
1	knn1							67.43615	66.50295
2	knn3							65.66798	65.47151
3	knn5							64.29273	66.01179
4	knn7							65.76621	66.50295
5	knn9							65.91356	66.25737
6	knn11							65.81532	66.50295
7	knn13							65.66798	66.79764
8	knn15							66.94499	66.74853
9	knn17							67.53438	67.04322
10	knn19							65.86444	66.94499
11	knn21							67.58350	67.33792
12	knn23	66.84676	66.94499	67.19057	64.83301	66.06090	65.81532	67.23969	67.28880
13	knn25	67.19057	66.99411	66.89587	65.52063	66.01179	66.45383	67.63261	67.09234
14	knn27	67.58350	66.65029	67.14145	64.98035	66.79764	66.50295	67.82908	67.43615
15	knn29	67.04322	66.65029	67.28880	65.91356	66.65029	66.69941	67.82908	67.53438
16	knn31	66.79764	66.99411	67.28880	67.09234	66.40472	66.25737	67.87819	67.73084
17	knn33	66.94499	66.79764	67.33792	67.48527	66.55206	66.84676	67.63261	67.48527
18	knn35	67.82908	66.50295	67.48527	66.94499	66.45383	66.11002	67.58350	67.48527
19	knn37	67.48527	66.15914	67.33792	67.68173	66.60118	66.50295	67.63261	67.53438
20	knn39	67.14145	66.89587	67.23969	66.84676	66.30648	66.60118	67.68173	67.09234
21	knn41	67.09234	66.35560	67.77996	66.50295	67.09234	66.06090	67.82908	66.25737
22	knn43	67.04322	66.15914	67.73084	65.66798	67.48527	66.01179	67.63261	66.40472
23	knn45	65.81532	66.35560	67.82908	66.15914	67.38703	66.30648	68.07466	66.50295
24	knn47	65.86444	66.55206	67.97642	65.66798	67.19057	65.56974	67.73084	66.74853
25	knn49	66.20825	66.84676	67.48527	65.17682	66.99411	66.01179	66.89587	66.40472
26	knn51	66.01179	66.40472	67.73084	64.63654	66.94499	65.66798	67.38703	66.40472
27	knn53	66.79764	66.20825	67.63261	64.53831	67.04322	65.66798	66.65029	66.45383
28	knn55	66.65029	66.40472	67.33792	64.83301	66.94499	63.80157	66.74853	66.50295
29	knn57	66.20825	66.40472	67.68173	65.02947	67.04322	63.85069	66.60118	66.40472
30	knn59	66.25737	66.45383	67.53438	65.47151	67.19057	63.60511	66.60118	66.65029

	k values	trial1
1	knn1	64.29273
2	knn3	62.52456
3	knn5	64.04715

	k values	trial1	trial2	trial3	trial4	trial5	trial6	trial7	trial8
1	knn1	64.29273	63.26130	65.17682	63.06483	63.21218	63.45776	67.43615	66.50295
2	knn3	62.52456	65.37328	64.93124	62.27898	62.72102	63.16306	65.66798	65.47151
3	knn5	64.04715	65.66798	65.32417	62.13163	62.96660	65.22593	64.29273	66.01179
4	knn7	66.30648	67.19057	66.06090	62.47544	63.45776	64.78389	65.76621	66.50295
5	knn9	65.76621	68.22200	65.56974	64.14538	63.31041	65.42240	65.91356	66.25737
6	knn11	65.76621	67.87819	66.15914	63.80157	63.99804	66.20825	65.81532	66.50295
7	knn13	65.37328	67.14145	66.40472	64.48919	65.27505	66.25737	65.66798	66.79764
8	knn15	67.09234	66.60118	67.63261	65.12770	65.27505	66.15914	66.94499	66.74853
9	knn17	66.55206	66.15914	67.19057	64.63654	64.88212	66.20825	67.53438	67.04322
10	knn19	66.60118	66.30648	66.30648	64.48919	65.56974	66.60118	65.86444	66.94499
11	knn21	66.60118	66.94499	66.74853	64.93124	65.71709	66.20825	67.58350	67.33792
12	knn23	66.84676	66.94499	67.19057	64.83301	66.06090	65.81532	67.23969	67.28880
13	knn25	67.19057	66.99411	66.89587	65.52063	66.01179	66.45383	67.63261	67.09234
14	knn27	67.58350	66.65029	67.14145	64.98035	66.79764	66.50295	67.82908	67.43615
15	knn29	67.04322	66.65029	67.28880	65.91356	66.65029	66.69941	67.82908	67.53438
16	knn31	66.79764	66.99411	67.28880	67.09234	66.40472	66.25737	67.87819	67.73084
17	knn33	66.94499	66.79764	67.33792	67.48527	66.55206	66.84676	67.63261	67.48527
18	knn35	67.82908	66.50295	67.48527	66.94499	66.45383	66.11002	67.58350	67.48527
19	knn37	67.48527	66.15914	67.33792	67.68173	66.60118	66.50295	67.63261	67.53438
20	knn39	67.14145	66.89587	67.23969	66.84676	66.30648	66.60118	67.68173	67.09234
21	knn41	67.09234	66.35560	67.77996	66.50295	67.09234	66.06090	67.82908	66.25737
22	knn43	67.04322	66.15914	67.73084	65.66798	67.48527	66.01179	67.63261	66.40472
23	knn45	65.81532	66.35560	67.82908	66.15914	67.38703	66.30648	68.07466	66.50295
24	knn47	65.86444	66.55206	67.97642	65.66798	67.19057	65.56974	67.73084	66.74853
25	knn49	66.20825	66.84676	67.48527	65.17682	66.99411	66.01179	66.89587	66.40472
26	knn51	66.01179	66.40472	67.73084	64.63654	66.94499	65.66798	67.38703	66.40472
27	knn53	66.79764	66.20825	67.63261	64.53831	67.04322	65.66798	66.65029	66.45383
28	knn55	66.65029	66.40472	67.33792	64.83301	66.94499	63.80157	66.74853	66.50295
29	knn57	66.20825	66.40472	67.68173	65.02947	67.04322	63.85069	66.60118	66.40472
30	knn59	66.25737	66.45383	67.53438	65.47151	67.19057	63.60511	66.60118	66.65029

kNN, trial 1


68
66.5
65
63.5
62
knn1
knn5
knn9
knn13
knn17
knn21
knn25
knn29
knn33
knn37
knn41
knn45
knn49
knn53
knn57

Words (Unigrams) as Features

Words as Features

- 98,257 (including heuristic features) with *unigrams*
- Significant performance hurdles; limited computing resources
- Only able to run Naive Bayes

Naive Bayes

66.59% correctly classified

Classified Data

NB Prediction	Informal	Formal	Sum
Informal	796	175	971
Formal	437	424	861
Sum	1233	599	

Final ML Algorithm Comparison

Heuristic Features Only

Algorithm	% Correctly Classified
Naive Bayes	63.6%
Linear Discriminant Analysis	66.7%
Decision Trees	**70.3%**
kNN, k=35, trial 1	67.8%

Heuristic + Unigrams

Algorithm	% Correctly Classified
Naive Bayes	66.6%

Conclusion

- Given current work, framework is **not** effective at classifying emails as formal vs. informal

- Subjectivity of classification (formal vs. informal)

- Quality and quantity of training data

    - A collaborative approach could be used requiring numerous ‘votes’ for each classification

Future Work

- Better tools for identifying heuristic ‘parts’ of emails (greeting, closing, signature block, etc.)
- Larger database of net lingo terms and misspellings
- Emoticon detection
- More email contextual features using metadata
- Experiment with additional ML algorithms (esp. deep learning techniques)