Tutorial #2 for Assignment 1: Feature Extraction

Jan 25th, 2017

TA: Aryan Arbabi
arbabi@cs.toronto.edu

(Slides adapted from Stefania Raimondo, Erin Grant, Siavash Kazemian, Varada Kolhatkar and Ka-Chun Won)
Goal

Perform sentiment analysis on individual tweets:
Binary classification as displaying either positive or negative sentiment

Input

“I love my BrandNameProduct!”

“Never shop at X-Store. Total garbage.”

Output

Positive

Negative
Methodology

Tweet Corpus

Tweet Corpus

Preprocessing

Normalized Tweets

Feature Extraction

Feature Vectors

Classification

Classes
Methodology

Tweet Corpus → Preprocessing → Normalized Tweets → Feature Extraction → Feature Vectors → Classification → Classes

Today’s class
Why Do Feature Extraction?

**Goal:** Identify the groups (positive & negative affect) in the feature space.
Feature Extraction: Input

Table 2: Conversion from raw tweets to tagged tweets

Raw tweet:


Output from twtt.py:

...<A=4>
Meet/ VB me/ PRP today/ NN at/ IN the/ DT FEC/ NN in/ IN DC/ NN at/ IN 4/ NN ./ .
Wear/ VB a/ DT carnation/ NN so/ RB I/ PRP know/ VB it/ PRP 's/ POS you/ PRP . ./
<A=0>
What Features to Extract?

You need to extract **20 features** from the tweet data.

---

Table 3: Features to be computed for each text

- Counts:
  - First person pronouns
  - Second person pronouns
  - Third person pronouns
  - Coordinating conjunctions
  - Past-tense verbs
  - Future-tense verbs
Feature Definitions

• First/second/third person pronouns
  • Look at /u/cs401/Wordlists/*-person

• Coordinating conjunctions (CC):
  • and, but, for, nor, or, so, and yet

• Past and future tense verbs
  • You should be able to come up with some rules for most cases...
  • But watch out for irregular verbs!
  • Perfective aspect (has/have eaten) should be counted as one token.
Feature Definitions

• Number of common nouns (NN, NNS)

• Number of proper nouns (NNP, NNPS)

• Number of Adverbs (RB, RBR, RBS)

• Number of wh-words (WDT, WP, WP$, WRB)

  • Use the tagger output!
Feature Definitions

• Average length of sentences

• Average length of tokens
  • Exclude punctuation tokens!

• Number of sentences

• Number of commas, colons, semi-colons, dashes, parentheses, ellipsis
Feature Definitions

• Slang
  • Look at /u/cs401/Wordlists/Slang

• Words all in upper case
  • For this, only look at words at least 2 characters long
ARFF Format

@relation twit_classification

@attribute 1st_person_pro numeric
@attribute 2nd_person_pro numeric
@attribute 3rd_person_pro numeric
...
@attribute emotion numeric

@data
0,0,0,0,0,1,1,0,...,13.0,5.2,1,4
0,0,1,0,0,0,0,0,...,19.0,4.2,1,0
0,1,0,0,0,0,0,0,...,5.0,3.75,2,4
ARFF Format

@relation twit_classification

@attribute 1st_person_pro numeric
@attribute 2nd_person_pro numeric
@attribute 3rd_person_pro numeric
...

@attribute emotion numeric

@data
0,0,0,0,0,1,1,0,...,13.0,5.2,1,4
0,0,0,0,0,0,0,0,...,19.0,4.2,1,0
0,1,0,0,0,0,0,0,...,5.0,3.75,2,4
ARFF Format

@relation twit_classification

@attribute 1st_person_pro numeric
@attribute 2nd_person_pro numeric
@attribute 3rd_person_pro numeric
...

@attribute emotion numeric

@data
0,0,0,0,0,1,1,0,...,13.0,5.2,1,4
0,0,1,0,0,0,0,0,...,19.0,4.2,1,0
0,1,0,0,0,0,0,0,...,5.0,3.75,2,4

Each "data" line is one tweet.
Implementation

• Create a python file named `buildarff.py`

• Write standalone functions named `feat1` to `feat20`
Command-line Example

buildarff.py test.twt some.arff 50

- input filename
- output filename
- Number of tweets per class
Command-line Arguments

• Extract argument values from the list `sys.argv`

• Arguments are:
  • Input file
  • Output file
  • Optional: Maximum number of tweets from each class (positive or negative)
    • Default: Use all tweets
Implementation

You will be evaluated on:

• Correct implementation of all features and creation of .arff

• Modularity, efficiency, clean and well-documented code
  • If the automarker fails, we cannot inspect your code to see what went wrong if it's hard to read!
Tips

- Python features you may want to use:
  - Dictionaries, regular expressions
  - String formatting (% operator):

    "@ attribute twit {%(s)\n" % ",". join([1, '2', '3'])"

- Do not hardcode filepaths in your home directory.
  - Instead, reference CSC401 folders such as /u/cs401/Wordlists/.
  - Or place the necessary data somewhere in the files you submit.

- Finish Part 1 ASAP and move on to this and the other parts of the assignment!
Questions?