Sentiment Analysis on Twitter
25 Nisan 17

Current sentiment analyser

1) Data
   a) Current dataset:
      i) 3000 polar tweets (http://www.kemik.yildiz.edu.tr/data/File/3000tweet.rar)
      ii) Label: positive, negative
      iii) Mostly brand-specific
   b) New annotations - *not ready yet*
      i) Initial set
         (1) Raw data: tweets randomly selected from the dump of 2013 01-03, having the keywords (belge, karne, turkcell, garanti, halkbank..) (Cengiz'den)
         (2) 700 tweets randomly selected from each month
         (3) Two sets of 500 tweets; each set will be annotated by two people
         (4) Todo
            (a) Get the annotations
            (b) Measure inter-annotator agreement
               (i) If not reliable, re-annotate
            (c) Get majority labels; remove disagreed texts
      ii) Alternatives:
            (a) We determine the keywords and the temporal range
            (b) 20 million tweets from kemik-yildiz but no metadata, all texts in one file.
   c) Preprocessing:
      i) Twitter-specific
         (1) Replace any username with <@USER>
         (2) Replace any link with <@URL>
         (3) Remove "RT @xxx:"
         (4) Don't touch hashtags
         (5) Remove duplicates
         (Ref: 1) https://www-cs.stanford.edu/people/alecmgo/papers/TwitterDistantSupervision09.pdf
         2) www.cs.columbia.edu/~julia/papers/Agarwaletal11.pdf)
      ii) Text-general
         (1) Stopwords removed
         (2) No stemming : our previous tests showed it is of no use and misses -me,-ma.

(3) Numbers removed
(4) Deasciified
(5) Lowercase
(6) Punctuation removed

2) Single-word prediction
   a) Output NEUTRAL if the word not in lexicon
      i) Lexicon is from (https://www.cmpe.boun.edu.tr/~ozgur/papers/polarity-seed-words.zip)
         (1) Some words removed / added.
   b) Improved the lexicon with SentiTurkNet (research.sabanciuniv.edu/27677/)
   c) Todo
      i) Use wordnet
         (1) Current sentiturknet is not promising.
      ii) Use word embeddings
         (1) Handle out-of-vocab words

Results (the best one highlighted):

Dataset	**Size**	**Feature setting**	**classifier**	**acc**	**fscore**	**model_folder**	**Tested on tweets**
tweets	3K	Word (uni+bi); char (bi) tfidf	NB	0.76	0.67 (negative recall:0.39)		-
tweets	3K	Word (uni+bi); char (bi) tfidf	SVM	0.78	0.75	tr_tweet1_svm	-
tweets	3K	Word (uni+bi); char (bi) tfidf; **lexicon**	NB	0.73	0.65	tr_tweet2_nb4	
tweets	3K	Word (uni+bi); char (bi) tfidf; **lexicon**	SVM	0.77	0.75	tr_tweet2_svm4	-

movie + product	16K	Word (uni+bi); char (bi) tfidf	NB	0.86	0.86	tr_reviews1_nb	
movie + product	16K	Word (uni+bi); char (bi) tfidf	SVM	0.84	0.84	tr_reviews1_svm	
movie + product	16K	Word (uni+bi); char (bi) tfidf; **lexicon**	NB	0.86	0.86	tr_reviews2_nb	
movie + product	16K	Word (uni+bi); char (bi) tfidf; **lexicon**	SVM	0.84	0.84	tr_reviews2_svm	
movie	10K	Word (uni+bi); char (bi) tfidf	SVM	0.88	0.88	tr_movie1_svm	Acc:0.69; f-sc: 0.67 (positive prec.and recall low)
movie	10K	Word (uni+bi); char (bi) tfidf; **lexicon**	SVM	0.89	0.89	tr_movie1_svm	Acc:0.69; f-sc: 0.67 (positive prec.and recall low)