Harnessing Context Incongruity for Sarcasm Detection (Joshi et al 2015)

Gist

- The key part of this paper is that incongruity e.g. clashes in sentiment are central to the detection of sarcasm
- "It must be noted that our system only handles incongruity between the text and common world knowledge (i.e. the knowledge that '*being stranded*' is an undesirable situation and, hence, '*Being stranded in traffic is the best way to start my week*' is a sarcastic statement)." (p 758)
- "This leaves out an example like '*Wow! You are so punctual*' which may be sarcastic depending on situational context" (p 758)
- Explicit Incongruity is where there are polarity signifying words that make the clash in sentiment apparent
- Implicit incongruity is where there are phrases that imply a particular sentiment conventionally. **These are the ones that seem the most interesting to see how they deal with them**.

Dataset

Primarily focused on tweets.

- Tweet-A (5208 Tweets, 4170 sarcastic) Downloaded by looking for certain hash tags (#sarcasm, #sarcastic adn #notsarcastic) and then did a rough quality control check to make sure that they made sense, removing wrongly labeled examples.
- Tweet-B (2278 tweets, 506 sarcastic) manually labeled for Riloff et.al 2013. I suspect what they're doing here is trying to balance the class distributions for this since predicting

sarcastic tweets using the Tweet-B dataset would be quite difficult.

Discussion board datasets

- Discussion-A (1502 discussion board posts, 752 sarcastic). Obtained from the Internet Argument Corpus (Walker et al. 2012). Manually annotated,. 752 sarc and non-sarc posts are selected randomly.

ML System

Detecting incongeruity

- Identifying phrases with implicity sentiment
- Obtained using algorithm given in Riloff et al. (2013) but extract both possible polarities for both nouns and verbs
- Keeping subsumed phrases "(i.e. `being ignored' subsumes 'being ignored by a friend')"
- Riloff et al. 2013 used these phrase as part of rules while this approach is a ML approach that uses them as features.

Features

- Unigrams
- Number of capital letters
- Number of emoticons and lol's
- Number of Punctuation marks
- Boolean feature indicating whether implicitly incongruous phrases were extracted.

Explicit Incongruity features

""""

- Number of times a word is followed by a word of opposing polarity
- Length of largest series of words with polarity unchanged
- Number of positive words
- Number of negative words

- Polarity of tweet based on words present """

Analysis

- Ran into errors with subjective things (Maybe this would be resolved if they wre able to look more closely at a user's history)
- Errors when there was incongruity but it was not within the text
- Incongruity due to numbers causes errors, here's the example they provide "*going in to work for 2 hours was totally worth the 35 minute drive*"
- Pieces of sarcastic text embedded in a larger non-sarcastic text were harder to identify.
- Politeness of sarcasm introduced difficulties.

Revision #3

Created Thu, Feb 28, 2019 9:41 PM by kenneth

Updated Tue, Mar 26, 2019 3:26 PM by kenneth