Lexicons

National Research Council Canada (NRC) Emotion Lexicon

IMPORTANT NOTE: remember to cite as per Terms of Use in their readme

This is the lexicon that we are most interested in as it is most directly related to our project. There are two forms of this lexicon: the "word-sense" lexicon is the original annotated at the word-sense level and the "word" lexicon is a baked version which condenses all word-senses for a word.

The two are quite similar at a glance, but they certainly differ at some level. We might want to mesure that with some sort of comparison script. We can write something simple to start which will just compare the list of associations (as the "word-level" naturally omits the symonym section). We will count any entry in the association list of one form that isn't in the other as different, which leads to a surprisingly high 17.9% difference.

```
TOTAL NUM DIFFERENT 2541
        17.917% different
```

This result throws us off a bit at the level of intuition, so we print a few things out and realize the "sense-level" writes "anticipation" as "anticip" while the "word-level" supports full words. We then clean our dataset and only see a four-hundred word shrink.

```
TOTAL NUM DIFFERENT 2149
        15.153% different
```

Which still may seem a bit shocking, but if we take a cursory glance at the differences, we see that they're usually rather similar as the "word-level" is a superset of the "sense-level". The two association lists often differ by either one or all entries. We calculated difference by count both separating the category "ALL" from the rest of the data and including it--excluding paints a far better picture though. We have over half of the differences present in the "ALL" category--meaning that there was no affect at the word-sense level from the associated lexicon, whereas the union of all word-senses had some sort of affect. The count for having one difference also gives some insight, perhaps we only get small differences at the word-sense level. Regardless, the similarity between the two methods of counting suggests that affect is markedly similar at the word vs word-sense level, and we may be able to overlook word-sense detection for our model.

```
# Including "ALL" in data
# numdiff   count       percent
  1         1270        0.591
  ALL       1157        0.538
  2         488         0.227
  3         223         0.104
  4         114         0.053
  5         43          0.02
  6         8           0.004
```

```
  7         2           0.001
  9         1           0.0
```

```
# Excluding "ALL" from data
# numdiff   count       percent
  ALL       1157        0.538
  1         585         0.272
  2         235         0.109
  3         110         0.051
  4         48          0.022
  5         11          0.005
  6         2           0.001
  9         1           0.0
```

Methodology

Saif M. Mohommad and Peter D. Tourney compiled this lexicon with crowdsourcing through Amazon's Mechanical Turk (an online crowdsourcing platform); they chose crowdsourcing as it is quick and inexpensive (costing them $2100 for the Turkers). As a deterrent of bad responses, they included a filtering question in each survey that asked for the best synonym for the given word, allowing them to identify either lack of word knowledge or probabilitically filtering random responders. They selected joy, sadness, anger, fear, trust, disgust, surprise, and anticipation as per Robert Plutchik's wheel of basic emotions, as well as drawing from the present emotion lexicons WordNet Affect Lexicon, General Inquirer, and Affective Norms for English Words and both the Maquarie Thesaurus and Google's N-Gram corpus. They generated questions with the Macquarie Thesaurus with the aforementioned filtering-question followed by questions asking for alignment with the various emotions. They also included polarity (positive vs negative valence) in the lexicon, giving us 10 categories to work with.

Our Representation

We wanted to preserve their data, but bring it into our database (MongoDB). This transfer was relatively painless, as their lexicon was in consistent TSV. We borrowed a decent amount of JSON utilities and structure from our thesaurus-scraper, writing to files by first letter as we go; all that changes is the shift from making http requests and parsing HTML to loading a local file and parsing TSV. We did this for both the "word-level" and the "sense-level" forms resulting in the following scheme:

```
{
  "<word>": {
    "synonyms": [
      "<list",
      "<of>",
      "<synonyms>",
    ],
    "associations": [
      "<list",
      "<of>",
      "<associations>",
    ],
```

```
    "word": "<word>"
  },
  ...
}
```

Note that the "synonyms" field is only present in the "sense-level" form, otherwise the schema are equivalent. This is then easily POST-ed to our API and can be accessible!