



<http://tinyurl.com/2dzbea59>

2 - 3:15PM



<http://tinyurl.com/46czd2ex>

5 - 6:15PM

- Activity 4 (mini)
- Sentiment analysis

Natural Language Processing (NLP)

- Branch of AI focused on how computers can process language like humans do (from Harvard Business Review)
- Branch of AI that helps computers understand, interpret and manipulate human language; draws from many disciplines, including computer science and computational linguistics, in its pursuit to fill the gap between human communication and computer understanding (SAS Institute)

- “I love Deadpool! I am so excited for Deadpool and Wolverine!”
- “I am not a fan of comics, so I bleh about Deadpool”
- “Deadpool is coming out in July”

What is sentiment? What are we measuring?

According to VADER's creators, C.J. Hutto and Eric Gilbert,

“Sentiment analysis, or opinion mining, is an active area of study in the field of natural language processing that **analyzes people's opinions, sentiments, evaluations, attitudes, and emotions** via the computational treatment of subjectivity in text”

Types of sentiment analysis

- Lexicon-based (least accurate; least processing)
 - E.g. VADER, textblob
- Machine learning
 - Using training sets of labeled data, decision trees, neural networks
- Pre-trained transformer-based deep learning ((usually) most accurate; MUCH more processing)
 - E.g. Chat GPT-4, BERT

VADER: Lexicon-based

- VADER
 - (Valence Aware Dictionary and sEntiment Reasoner) is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media

How VADER was built

Rate the following words from -4 (extremely negative) to 4 (extremely positive):

- Awesome
- Bad
- Clever
- Discard
- Excellence

How VADER was built

Enlisted 10 different people to rate thousands of words positively or negatively on a scale of -4 to 4

- Awesome 3.1 0.83066 [3, 4, 2, 3, 2, 2, 4, 4, 4, 3]
- Bad -2.5 0.67082 [-3, -2, -4, -3, -2, -2, -3, -2, -2, -2]
- Clever 2.0 0.7746 [2, 1, 2, 2, 2, 1, 3, 1, 3, 3]
- Discard -1.0 0.44721 [-1, -1, -1, -1, 0, -1, -1, -1, -2, -1]
- Excellence 3.1 0.9434 [4, 3, 4, 3, 2, 3, 1, 4, 3, 4]

How VADER was built

1. If there's punctuation, especially exclamation points, the sentiment intensity should be increased (e.g., "Mochi ice cream is bliss" 👍 vs "Mochi ice cream is bliss!!!" 👍👍👍)

2. If there's capitalization, especially all caps, the sentiment intensity should be increased (e.g., "Mochi ice cream is bliss" 👍 vs "Mochi ice cream is BLISS" 👍👍👍)

3. If there are words like "extremely" or "absolutely", the sentiment should be increased ("Mochi ice cream is good" 👍 vs "Mochi ice cream is extremely good" 👍👍👍)

How VADER was built

4. If there's a "but" in a sentence, the polarity of the sentiment should shift, and the sentiment that follows the "but" should be prioritized (e.g., "Mochi ice cream is nice" 👍 vs "Mochi ice cream is nice, but it's a little blah" 👎)

5. If there's a negation before an important word, the sentiment polarity should be flipped ("Mochi ice cream is my favorite" 👍 vs "Mochi ice cream is not my favorite" 👎)

How VADER is built

“Deadpool is coming out in
July”

“Deadpool is coming out in
July 😁”

How VADER was built

- Cutsies 1.0 1.18322 [2, 2, 1, 1, 0, 2, 2, -1, 2, -1]
- Boldly 1.5 1.62788 [3, -2, -1, 1, 2, 3, 2, 3, 2, 2]
- Does it matter that we're missing out on specific nuances? Is this the best tool for capturing what we're trying to study and understand?

VADER's sentiment analysis

- Works very fast
- Doesn't require any training or set up
- Works best with social media texts/sentences

Interpreting VADER's output

```
sentimentAnalyser.polarity_scores("I like the Marvel movies")
```

```
{'neg': 0.0, 'neu': 0.361, 'pos': 0.639, 'compound': 0.6486}
```


Setting up the python environment

- `!pip install vaderSentiment`
- `from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer`
- `import pandas as pd`
- `pd.options.display.max_colwidth = 400`

Using VADER on a sentence

- `sentimentAnalyser = SentimentIntensityAnalyzer()`
- `sentimentAnalyser.polarity_scores("")`

Try a sentence with exclamation point, capital letters, and words like “very” and “extremely.” How does the score change?

Using VADER on a list of sentences

- `sentences = [<list of sentences>]`
- `for sentence in sentences:`
 - `sentiment_scores = sentimentAnalyser.polarity_scores(sentence)`
 - `print(f"{sentence}: {sentiment_scores}")`

Practice: Sentiment analysis on your favorite song

Pick a verse from your favorite song and conduct a sentiment analysis.

- Before running sentiment analysis, do you think your favorite song is a happy or sad song?
- After running sentiment analysis, did the result match your expectation?

Applying on a dataset

Drop Columns

- `.drop(columns= "column name"/[<list of column names>])`
 - `vac_df = vac_df.drop(columns="user_id")`

Creating a function to apply sentiment analysis

- ```
def calculate_sentiment(text):
 scores = sentimentAnalyser.polarity_scores(text)
 compound_score = scores['compound']
 return compound_score
```

# Adding a new column and applying function

- `df[“<new_column_name>”] = <content of new column>`
- `df[“<column>”].apply(<function to apply>)`
  - `vac_df['sentiment_score'] = vac_df['text'].apply(calculate_sentiment)`



# Setting date as index for dataframe

- `df.set_index('<column>')`
  - `vac_df.set_index('date')`

## Plot average sentiment score by month

- `vac_df.resample('M')['sentiment_score'].mean().plot(title="Tweet Sentiment by Month")`
- Saves the plot
  - `.get_figure().savefig('tweetsentiment_jun21')`