Sentiment Analysis

Week 6

Panggih Kusuma Ningrum

Contents

- What is TextBlob?
- What is VADER?
- What is Flair?
- Comparison between TextBlob vs. VADER vs. Flair.
- Coding Exercise: How to use Python library TextBlob, VADER, and Flair's pre-trained model for sentiment analysis.

What is Sentiment Analysis?

Sentiment analysis is a Natural Language Processing (NLP) technique used to determine whether data is positive, negative or neutral.

Sentiment analysis is also known as opinion mining, opinion extraction, sentiment mining, subjectivity analysis, effect analysis, emotion analysis, review mining, etc (although the tasks are slightly different).

Sentiment Analysis

My experience so far has been fantastic!

POSITIVE

The product is ok I guess

NEUTRAL

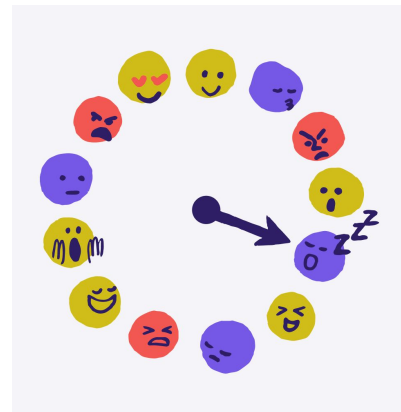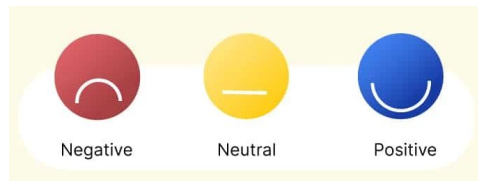Your support team is useless

NEGATIVE

Graded Sentiment Analysis

Sentiment analysis focuses on the polarity of a text (*positive, negative*) but it also goes beyond polarity to detect specific feelings and emotions (*angry, happy, sad*, etc), urgency (*urgent, not urgent*) and even intentions (*interested, not interested*).

Binary Class

Multi-class

Challenges

People express emotions in different ways. Some words that typically express anger, like **bad** or **kill** (e.g. *your product is so bad* or *your customer support is killing me*) might also express happiness (e.g. *this is bad ass* or *you are killing it*).

More challenging examples of sentiment analysis

- I ***do not dislike*** horror movies. (phrase with negation)
- Disliking horror movies is **not** **uncommon**. (negation, inverted word order)
- ***Sometimes*** I really hate the show. (adverbial modifies the sentiment)
- I love having to wait two months for the next series to come out! (sarcasm)
- The final episode was surprising with a **terrible** **twist** at the end (negative term used in a positive way)
- The film was easy to watch but I would not recommend it to my friends. (difficult to categorize)
- I *LOL'd* at the end of the cake scene (often hard to understand new terms)

Benefits of Sentiment Analysis

- **Social media sentiment analysis:** analyze the sentiments of Facebook posts, twitter tweets, etc.
- **Brand Experience Insights:** gather valuable brand experience insights that can give you a peep into hidden market sentiment about your brand and what customers expect from you.
- **News Trend Analysis:** Industries such as banking, insurance, real estate, automotive, cosmetics, etc. use ML-based sentiment analysis to understand and analyze such news in order to speculate, plan, and be ready for any situation.
- **Competitor Analysis:** Through analyzing sentiment in publically available data you can find out why competitor products are more successful than yours, and why people prefer certain products from you but go to your competitors for others.
- etc

Approaches and Algorithms

Sentiment analysis can be conducted either with or without constructing a machine learning model. Various sentiment analysis systems employ lexicons, which are essentially lists of words along with the emotions they express. In addition, complex machine learning algorithms are often used. Alternatively, a transfer learning approach involving the use of pre-trained models is also common in sentiment analysis.

Let's take a look at TextBlob, VADER, and Flair!

TextBlob

TextBlob is a Python library for Natural Language Processing (NLP). Sentiment analysis is one of many NLP tasks that TextBlob supports.

The sentiment property in TextBlob returns a polarity score and a subjectivity score for the input text.

- The polarity score ranges from -1 to 1, where -1 means extremely negative, and 1 means highly positive. A score near 0 means neutral sentiment.
- The subjectivity score ranges from 0 to 1, where 0 means extremely objective and 1 means highly subjective.

TextBlob

Pros:

- Simplicity: TextBlob is very user-friendly, making it easy for beginners to implement basic NLP tasks, including sentiment analysis.
- Integration: It integrates a lexicon-based approach with a simple Naive Bayes classifier, offering a balanced solution.
- Multilingual Support: Supports various languages, which is useful for multi-language projects.

Cons:

- Textblob will ignore the words that it doesn't know, it will consider words and phrases that it can assign polarity to and averages to get the final score.
- Accuracy: May not be as accurate as more advanced methods, especially in handling complex language constructs or contextual nuances.
- Performance: While generally efficient, it might not be the best choice for very large datasets or for tasks requiring highly nuanced sentiment analysis.

VADER

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*. VADER uses a combination of A sentiment lexicon is a list of lexical features (e.g., words) which are generally labeled according to their semantic orientation as either positive or negative. It also has a built-in algorithm to change sentiment intensity based on punctuations, slang, emojis, and acronyms.

The output of VADER includes four scores: compound score, negative score, neutral score, and positive score.

- The pos, neu, and neg represent the percentage of tokens that fall into each category, so they add up together to be 100%.
- The compound score is a single score to measure the sentiment of the text. Similar to TextBlob, it ranges from -1 (extremely negative) to 1 (extremely positive). The scores near 0 represent the neural sentiment score.
- The compound score is not a simple aggregation of the pos, neu, and neg scores. Instead, it incorporates rule-based enhancements such as punctuation amplifiers.

VADER

Pros:

- Specially Designed for Social Media: Excellently interprets sentiments in social media text, including slangs, emoticons, and acronyms.
- Real-time Analysis: Fast and doesn't require training data, making it ideal for real-time sentiment analysis.
- Ease of Use: Easy to implement and provides clear positive, neutral, and negative scores.

Cons:

- Context Limitation: While it's good with informal language, it may not accurately capture sentiment in more structured or formal text.
- No Training: It doesn't learn from data. The lexicon and rules are fixed, which can limit its adaptability to new or nuanced expressions.
- The main drawback with the rule-based approach for sentiment analysis is that the method only cares about individual words and completely ignores the context in which it is used.

FLAIR

Flair is built on PyTorch making it easy to train models and experiment with new approaches using Flair embeddings and classes. The pre-trained sentiment model offers a tool for sentiment analysis without training a customized model.
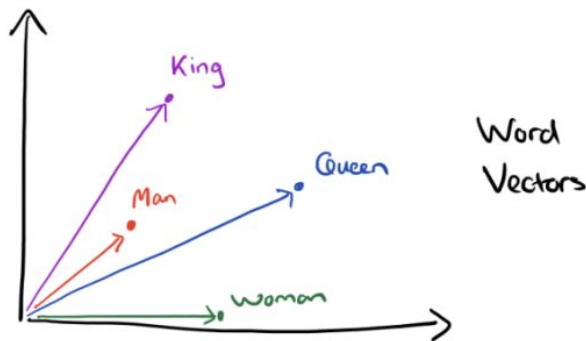
Text embeddings are a form of word representation in NLP in which synonymically similar words are represented using similar vectors which when represented in an n-dimensional space will be close to each other.

Embedding based python packages use this form of text representation to predict text sentiments. This leads to better text representation in NLP and yields better model performance.

Unlike TextBlob and VADER that output a sentiment score between -1 and 1, flair sentiment output the predicted label with a confidence score. The confidence score ranges from 0 to 1, with 1 being very confident and 0 being very unconfident.

FLAIR

Pros:

- Advanced Models: Offering high accuracy and understanding of language nuances compared to TextBlob or VADER
- Contextual Understanding: Better at understanding context and the sequence of words due to its sophisticated neural network models.
- Transfer Learning: Can be fine-tuned on specific datasets, making it adaptable and improving its performance on specialized tasks.

Cons:

- Resource Intensive: Requires more computational resources, especially for training or fine-tuning models.
- Speed: Generally slower than lexicon-based methods like VADER, especially when processing large datasets or using complex models.
- Flair pre-trained model for sentiment analysis is trained on IMDB data and this model might not generalize well on data from other domains like twitter.
- Complexity: More complex to set up and use, particularly when customizing or fine-tuning models.

How to use Python library TextBlob, VADER, and Flair's pre-trained model for sentiment analysis?

It's time for the coding exercise!