Sentiment Analysis Using BERT and Multi-Instance Learning

Iryna Burak

Pablo Restrepo

May 25, 2020

Presentation Overview: What did we do on the last 2 weeks?

1. **Creating baselines four our task:**
   1.1 Sentiwordnet.
   1.2. NLTK SentimentAnalyzer.
   1.3. NLTK VADER Sentiment Intensity Analyzer.
   1.4 Building a binary classifier with scikit-learn.
   1.5 Score to beat: numbers obtained by our baselines
2. **Feeding BERT embeddings to our Milnet.**
3. **Plan for the next two weeks.**
4. **References.**

1. Creating baselines four our task:

- Difficult task for our baselines, since they are learning from comment sentiments and not sentence sentiment.
- Better results for binary classification

1.1 Sentiwordnet:

Lexical resource explicitly devised for supporting sentiment classification and opinion mining applications.

Procedure:

- get post tagging (we only work with nouns, adjectives and adverbs).
- get synset and obtain position [0] (the most common synonym).

Determining sentiment:

- `sentiment += swn_synset.pos_score() - swn_synset.neg_score()`
- sum greater than 0 => positive sentiment

1.1 Sentiwordnet:

Binary classification (removing neutral sentences):

- accuracy:0.612
- f1_micro:0.612
- f1_macro:0.6114

Multiclass classification:

- accuracy:0.4371
- f1_micro:0.4371
- f1_macro:0.4197

```
alpha = 0.5
elif -alpha <= sentiment <= alpha:
      result = "0"
    elif sentiment >= 0:
      result = "p"
    else:
      result = "n"
```

1.3 NLTK VADER:

- VADER (**V**alence **A**ware **D**ictionary and s**E**ntiment **R**easoner)
- Lexicon and rule-based sentiment analysis tool specifically created for working with messy social media texts

- dictionary: {'neg': 0.0, 'neu': 0.889, 'pos': 0.111, 'compound': 0.3612}
- Compound: metric that calculates the sum of all the lexicon ratings which have been normalized between -1(most extreme negative) and +1 (most extreme positive).

  positive sentiment : (compound score >= 0.05)

  neutral sentiment : (compound score > -0.05) and (compound score < 0.05)

  negative sentiment : (compound score <= -0.05)

1.3 NLTK VADER:

Binary classification (removing neutral sentences):

- accuracy:0.652
- f1_micro:0.652
- f1_macro:0.6515

Multiclass classification:

- accuracy:0.456
- f1_micro:0.456
- f1_macro:0.454

1.2 NLTK SentimentAnalyzer:

- **Training Data:** Amazon Data.
- **Test Data:** Annotated Organic Dataset, (validation)

1.2 NLTK SentimentAnalyzer:

Binary classification (removing neutral sentences):

- accuracy: 0.632
- F-measure [p]: 0.673
- F-measure [n]: 0.577

Multiclass classification:

- accuracy:0.446
- F-measure [p]: 0.4717
- F-measure [0]: 0.470
- F-measure [n]: 0.382

1.4 Building a binary classifier with scikit-learn:

- SVM model from Scikit-Learn:
    - Unigram classifier
    - Bigram classifier
    - Unigram and bigram classifier
- **Training Data:** Amazon Data.
- **Test Data:** Annotated Organic Dataset, (validation)

1.4 Building a binary classifier with scikit-learn:

Binary classification (removing neutral sentences):

- **Unigram:** accuracy: 0.52
- **Bigram:** accuracy: 0.52

Multiclass classification:

- **Unigram:** accuracy:0.31
- **Bigram:** accuracy: 0.323

2. Feeding BERT embeddings to our MiInet

- Imbalanced data: used weights when computing the loss.

- Training for:
  - ratings
  - sentiments

2. Feeding BERT embeddings to our MiInet: on ratings

metric		amazon	organic
rating	F1 (micro)	0.561	-
	F1 (macro)	0.448	-
sentiment	F1 (micro)	0.756	0.453
	F1 (macro)	0.582	0.455

Amazon

True	1	2	3	4	5
1	52	17	5	4	6
2	12	42	12	4	2
3	16	28	60	38	9
4	13	25	71	111	71
5	61	37	47	203	605

Organic dataset

True	-	0	+
-	930	443	125
0	860	666	447
+	472	432	709

2. Feeding BERT embeddings to our Milnet: on sentiments

metric	amazon	organic
F1 (micro)	0.672	0.450
F1 (macro)	0.530	0.447

Amazon

True	-	0	+
-	92	34	14
0	26	100	24
+	130	281	850

Organic dataset

True	-	0	+
-	805	643	50
0	682	971	320
+	349	753	511

2. Feeding BERT embeddings to our MiInet

- Try less aggressive weights, downsampling/upsampling.
- Fine-tune on the organic dataset.
- Make the network wider.

3. Plan for the next two weeks

- Try to improve our model a lot.
- Try to improve our baselines a little bit.
- Run our network on word embeddings.

4. References

- Baccianella S., Esuli A, & Sebastiani F, SENTIWORDNET3.0: An Enhanced Lexical Resourcefor Sentiment Analysis and Opinion Mining,Istituto di Scienza e Tecnologie dell'InformazioneConsiglio Nazionale delle Ricerche
- George-Bogdan Ivanov, Getting Started with Sentiment Analysis, https://nlpforhackers.io/sentiment-analysis-intro/
- How to Sentiment analysis NLTK, https://www.nltk.org/howto/sentiment.html