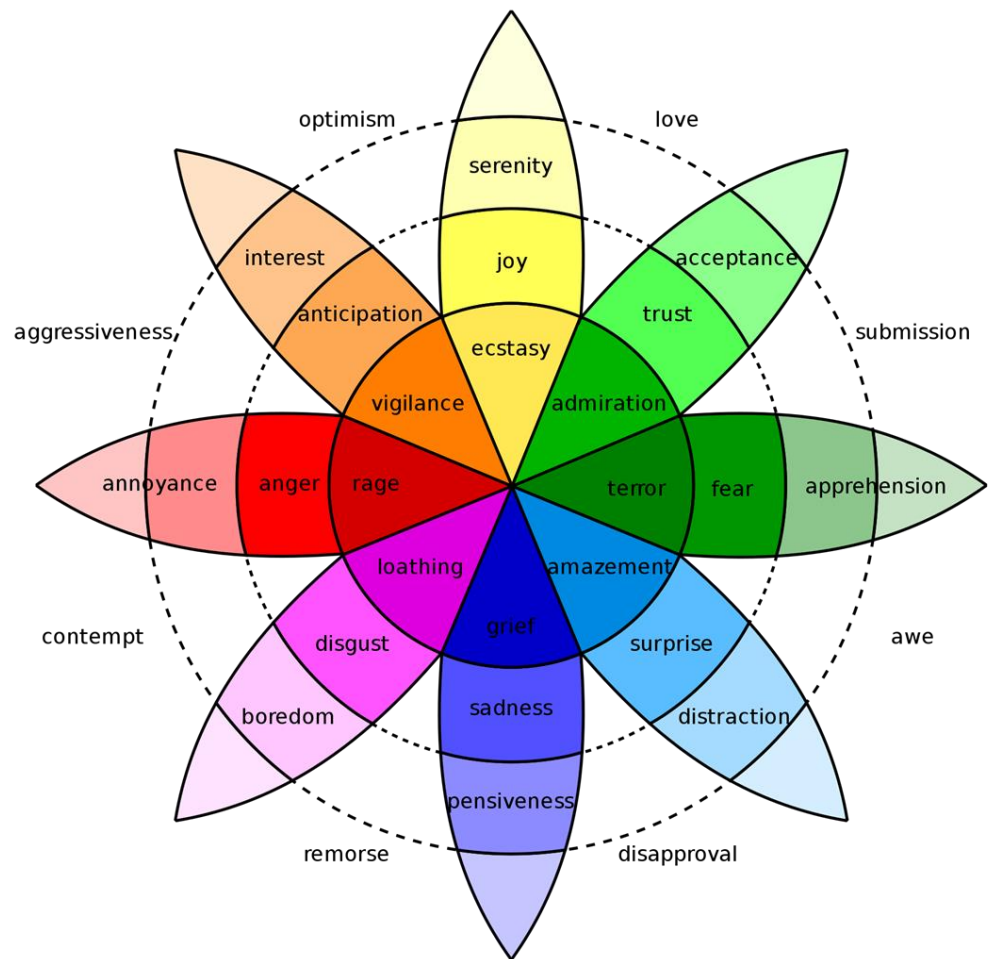


# Cross-Domain Sentiment Analysis

**AUTHORS:** Atulya Shetty and Payton Walker



**THE UNIVERSITY OF  
ALABAMA AT BIRMINGHAM**

Knowledge that will change your world

# of Product Review Data Using Flair

## ABSTRACT

Sentiment analysis is a popular technique used to gauge consumer sentiment from text data. Businesses can identify consumer perception towards their brand by analyzing product reviews, movie reviews, and social media comments. We use different pre-trained language models to create a sentiment classifier. We fine-tune the language models on consumer reviews obtained from one domain and then evaluate the model's performance on data from a different domain.

Our implementation is done using the Flair and HuggingFace frameworks, and we mainly use BERT and the XLNET model to fine-tune our classifier. We find that our model performs best when the training data and evaluation data are from the same subject and the model performance decreases when the training data and evaluation data are from different domains.

# and Language Model Embeddings



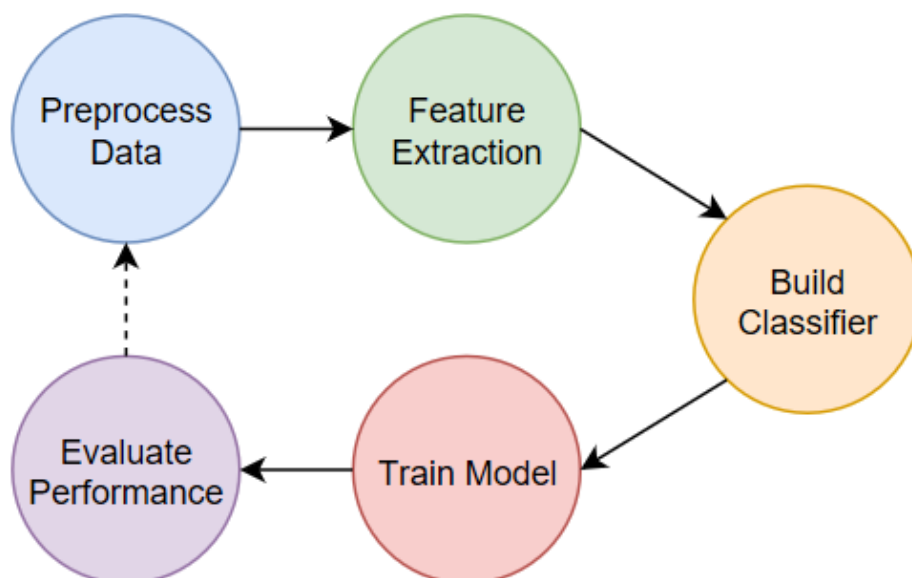
flair

## BACKGROUND

- Sentiment Analysis - “detection of **attitudes**, or dispositions towards objects or persons.”
- Considerations:
  - **Holder (source)** of attitude
  - **Target (aspect)** of attitude
  - **Type** of attitude
  - **Text** containing the attitude
- Task:
  - **Simple** – binary classification (POS/NEG)
  - **Complex** – Ranking classification (1-5)
  - **Advanced** – Detecting target, source, or attitude types
- Why?
  - Movies
  - Products
  - Public Sentiment
  - Politics
  - Prediction (Elections, Market trends, etc.)

# METHODOLOGY

- 1** Preprocessing Data
  - Pickle, gzip to compress data
  - Pandas to read and filter reviews
- 2** Feature Extraction
  - Word/Document Embeddings
- 3** Build Sentiment Classifier
- 4** Train Classification Model
- 5** Evaluate Performance
  - Data sets from different domain



## DATA SETS

- Amazon (190k) – Electronics reviews
  - IMBD (50k) – Movie Reviews
  - Yelp (10k) – Business Reviews
- Filtered to contain only verified reviews and reviews voted helpful by the users
  - Extracted only reviews and ratings information
  - Converted ratings to binary classes by categorizing 1-3 stars reviews as negative and 4-5 stars as positive
  - Split into train, test and holdout sets made up of various combinations

Dataset	Positive	Negative	Total
Yelp	6,863 (68.6%)	3,137 (31.4%)	10,000
IMDB	25,000 (50%)	25,000 (50%)	50,000
Amazon	139,681 (73.6%)	50,105 (26.4%)	189,786

# CLASSIFIERS

## - Flair

- NLP Framework on PyTorch
- Built-in text embedding library
- Developed by Zalando Research

## - HuggingFace

- State-of-the-art NLP Framework on PyTorch and TensorFlow 2.0
- Embedded Transformer library
- Pre-trained Model library

### Steps (Flair):

- 1) Build Corpus  
(**CSVClassificationCorpus**)
- 2) Create Label Dictionary
- 3) Define Word Embeddings (**XLNET**)
- 4) Define Document Embeddings [3]
- 5) Create Text Classifier Object [2, 4]
- 6) Create Model Trainer Object [1, 5]
- 7) Train Model (trainer.train())
  - LR = 0.1
  - Batch Size = 32
  - Epochs = 5

### Steps (HuggingFace):

- 1) Read training data
- 2) Tokenize training data
- 3) Preprocess data
- 4) Create Classifier
- 5) Set Optimizer
- 6) Train Model
  - LR = 0.00002
  - Batch Size = 32
  - Epochs = 4

➔ Varied Train and Test data from different review data sets to compare performance.

# RESULTS

Flair - XLNET							
ID	Train Data	Test Data	Label	Accuracy	Precision	Recall	F1-Score
F1	IMDB (pre-trained)	Amazon	-	0.6796	0.6962	0.6374	0.6655
F2	Yelp	Amazon	NEG	0.5475	0.7170	0.6984	0.7076
			POS	0.5565	0.7060	0.7244	0.7151
F3	IMDB+Yelp	Amazon	NEG	0.5343	0.7474	0.6520	0.6964
			POS	0.5783	0.6914	0.7796	0.7329
F4	IMDB+Yelp+Amazon	Amazon	NEG	0.6143	0.7415	0.7818	0.7611
			POS	0.5971	0.7692	0.7274	0.7477

- Similar evaluation scores for [POS] label for Yelp trained model even with class imbalance in dataset (70% POS, 30% NEG).
- Slight improvement in evaluation scores when trained on combination data.

HuggingFace - BERT						
ID	Train Data	Test Data	Accuracy	Precision	Recall	F1 Score
H1	Amazon	Amazon	0.8895	0.9531	0.9030	0.9274
H2	Amazon	Yelp	0.6465	0.8641	0.6975	0.772
H3	Amazon	IMDB	0.5517	0.6353	0.698	0.665

- Model trained and evaluated on Amazon reviews produces a better result as compared to the model trained on Amazon reviews and evaluated on Yelp or IMDB reviews.



## CONCLUSIONS

- Sentiment Analysis works best when Train and Test data are from the same domain.
- Data preprocessing is essential for cross-domain sentiment analysis
  - Potential linguistic differences among review sets (i.e., slang, domain specific terminology).
  - Different character sets (i.e. special characters)
- Classifier overcomes class bias (Yelp Dataset)
- More training and validation iterations (epochs) may improve these scores.

## LIMITATIONS

- Unsupported packages in Cheaha
  - Lack of permissions to install
  - Required updates (i.e., flair)
- Memory issues
- Inconsistent runtime errors
- Data errors mid-training
- Longer training times