# A comparative analysis between ML models for sarcasm detection

CS3244 Project Group 21


Joshua Chew Jian Xiang
Cao Ngoc Linh
Chen Yiyang
Ernest Lian Qi Quan
Quah Xi Wen

# Literature Review

## Multi-rule based ensemble feature selection model for sarcasm type detection in Twitter.

Author: Sundararajan, K., & Palanisamy, A. (2020)

- Focuses on feature selection
- 15 features identified to be used in our own project

# SARCASM detection using machine learning algorithms in Twitter: A systematic review

Author: Sarsam, S. M., Al-Samarraie, H., Alzahrani, A. I., & Wright, B. (2020)

- **Model:** SVM and CNN
- **Features:** lexical, pragmatic, frequency, and part-of-speech tagging
- **Ensembling:** Combining SVM and CNN gives the best performance

# A Deeper Look into Sarcastic Tweets Using Deep Convolutional Neural Networks

Author: Poria, S., Cambria, E., Hazarika, D., Vij, P. (2017)

- **Model:** CNN
- **Features:** sentiment, emotion and personality features
- **Ensembling:** Combining SVM and CNN also give the best performance

# Our Project

# Questions We Will Answer

1. Which **ML models** work best for detecting sarcastic comments on their own **without any context**? NEW

2. How do we **improve** these ML models to better detect sarcasm without context?

3. **Why** do certain ML models **not perform as well** for this specific task? NEW

# Dataset Exploration

# Sarcasm on Reddit | Kaggle

| | label | comment | author | subreddit | score | ups | downs | date | created_utc | parent_comment |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | NC and NH. | Trumpbart | politics | 2 | -1 | -1 | 2016-10 | 2016-10-16 23:55:23 | Yeah, I get that argument. At this point, I'd ... |
| 1 | 0 | You do know west teams play against west teams... | Shbshb906 | nba | -4 | -1 | -1 | 2016-11 | 2016-11-01 00:24:10 | The blazers and Mavericks (The wests 5 and 6 s... |
| 2 | 0 | They were underdogs earlier today, but since G... | Creepeth | nfl | 3 | 3 | 0 | 2016-09 | 2016-09-22 21:45:37 | They're favored to win. |
| 3 | 0 | This meme isn't funny none of the "new york ni... | icebrotha | BlackPeopleTwitter | -8 | -1 | -1 | 2016-10 | 2016-10-18 21:03:47 | deadass don't kill my buzz |
| 4 | 0 | I could use one of those tools. | cush2push | MaddenUltimateTeam | 6 | -1 | -1 | 2016-12 | 2016-12-30 17:00:13 | Yep can confirm I saw the tool they use for th... |

(unused)

**X** = df['comment']

**y** = df['label']

<u>Ground Truth:</u>
Comment is **sarcastic** if it was accompanied by a '/s' tag (removed from dataset)

# Sarcasm on Reddit | Kaggle - Parent Comments

| | label | comment | author | subreddit | score | ups | downs | date | created_utc | parent_comment |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | NC and NH. | Trumpbart | politics | 2 | -1 | -1 | 2016-10 | 2016-10-16 23:55:23 | Yeah, I get that argument. At this point, I'd ... |
| 1 | 0 | You do know west teams play against west teams... | Shbshb906 | nba | -4 | -1 | -1 | 2016-11 | 2016-11-01 00:24:10 | The blazers and Mavericks (The wests 5 and 6 s... |
| 2 | 0 | They were underdogs earlier today, but since G... | Creepeth | nfl | 3 | 3 | 0 | 2016-09 | 2016-09-22 21:45:37 | They're favored to win. |
| 3 | 0 | This meme isn't funny none of the "new york ni... | icebrotha | BlackPeopleTwitter | -8 | -1 | -1 | 2016-10 | 2016-10-18 21:03:47 | deadass don't kill my buzz |
| 4 | 0 | I could use one of those tools. | cush2push | MaddenUltimateTeam | 6 | -1 | -1 | 2016-12 | 2016-12-30 17:00:13 | Yep can confirm I saw the tool they use for th... |

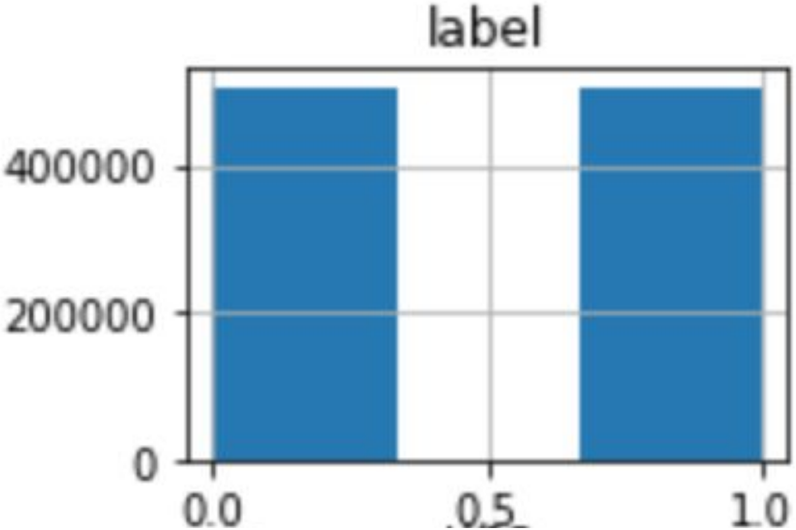
(unused)

`X = df['comment']`

`y = df['label']`

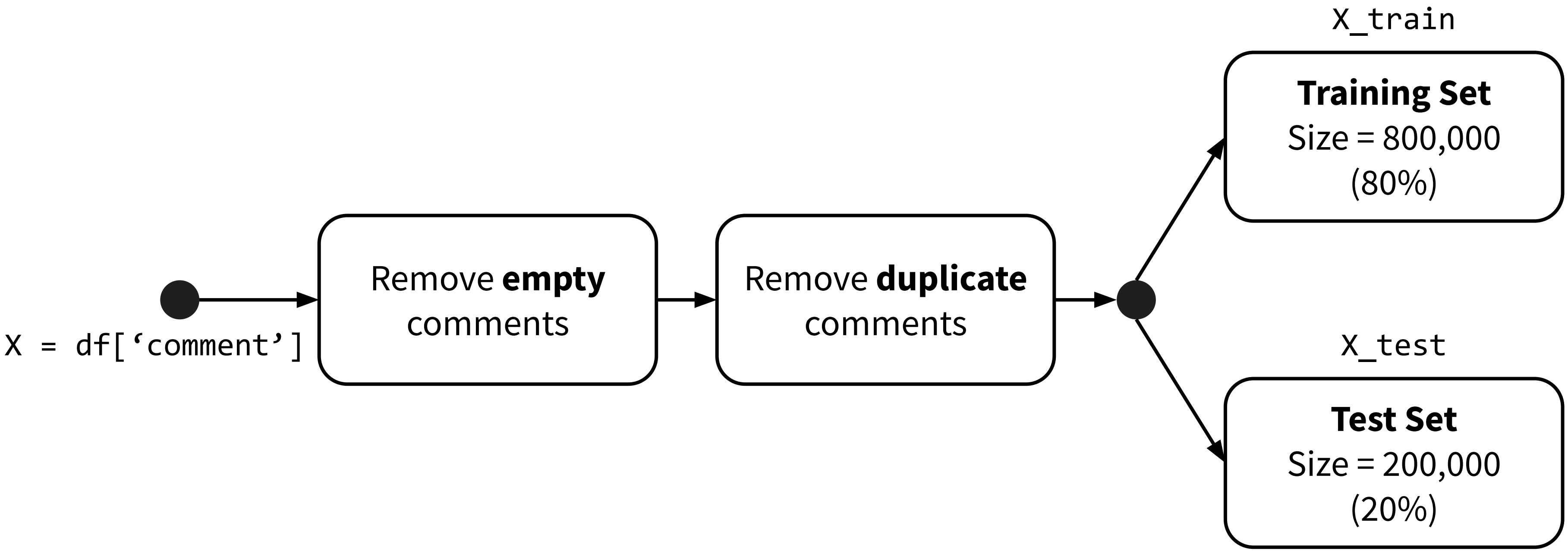
<u>Ground Truth:</u>
Comment is **sarcastic** if it was accompanied by a '/s' tag (removed from dataset)

# Balanced Data



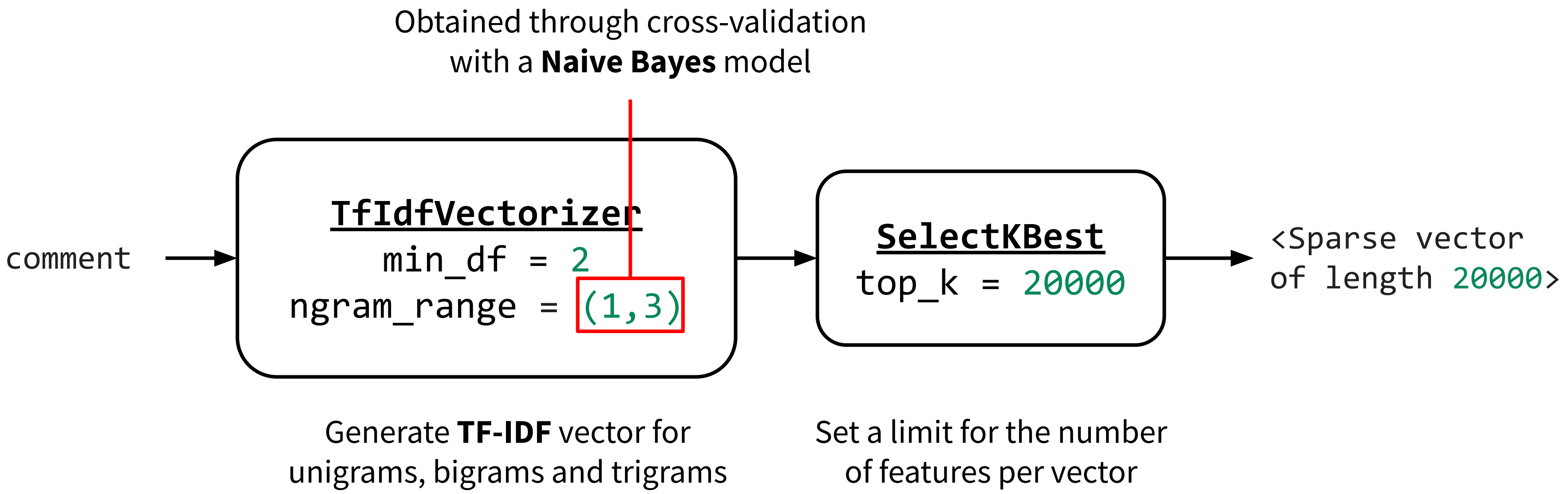

**Not sarcastic** **Sarcastic**

# Data Preprocessing

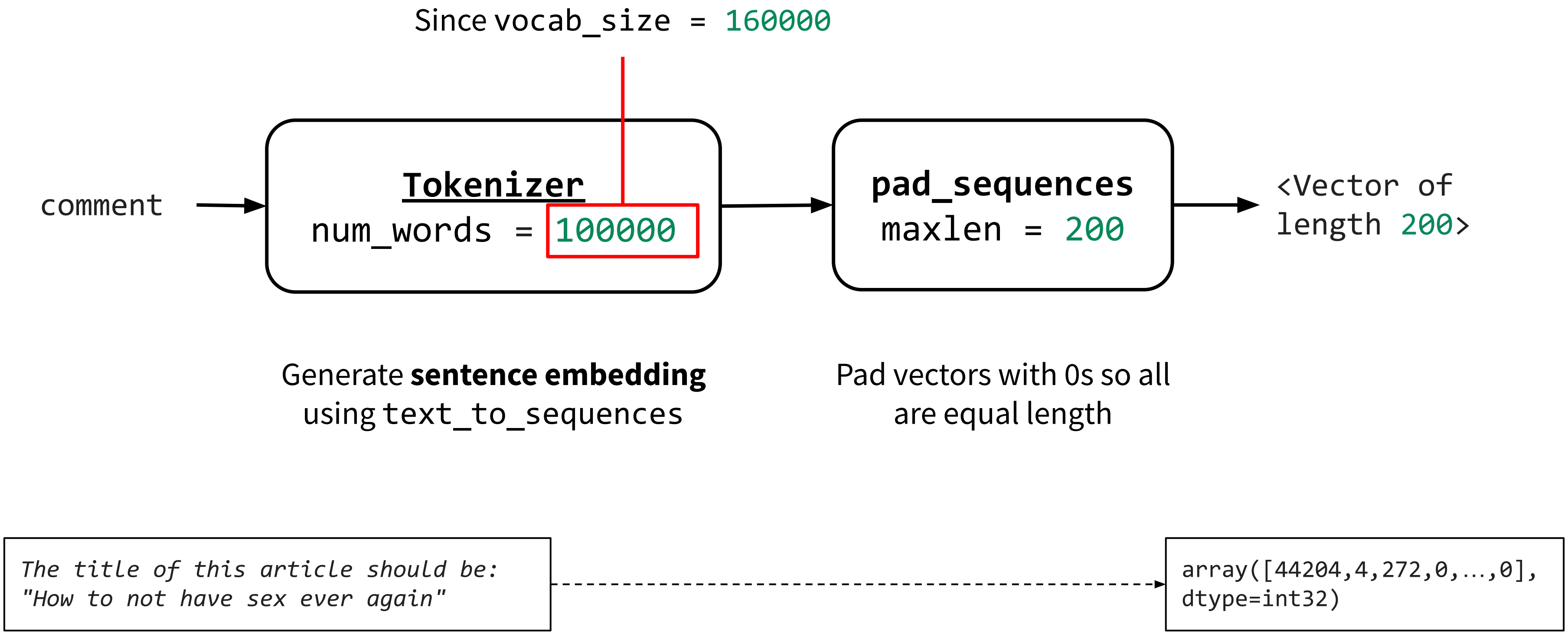


# Vector Representations

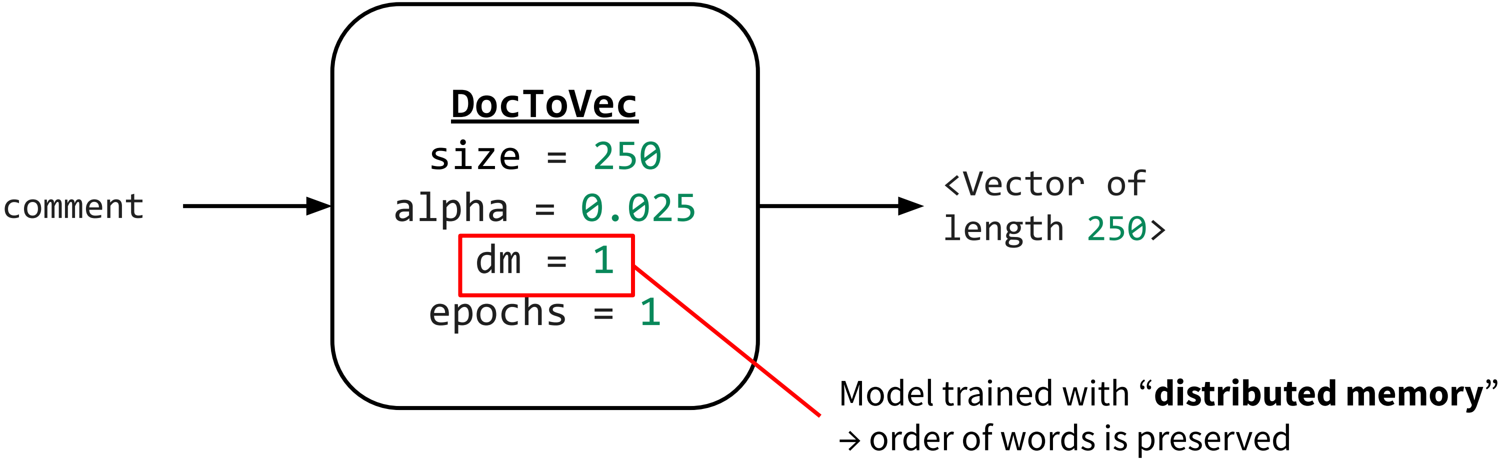
# 1. TF-IDF Vectors



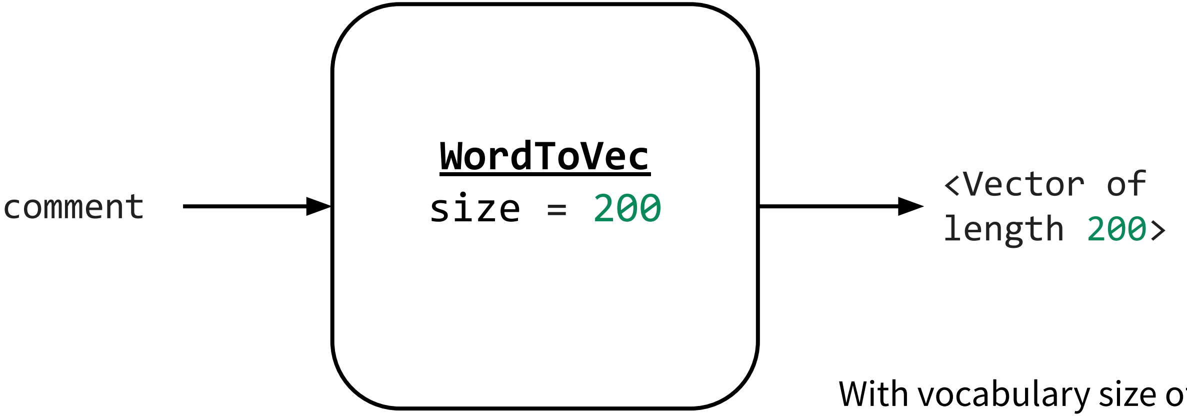
# 2. Keras Sentence Embeddings



# 3. DocToVec Sentence Embeddings



# 4. WordToVec Sentence Embeddings




With vocabulary size of |v|, embedding matrix will be of size |v| x 200

# 5. Manual Feature Extractions

```
I love Machine Learning! I am totally not going to totally fail! WOOHOO! :)
```

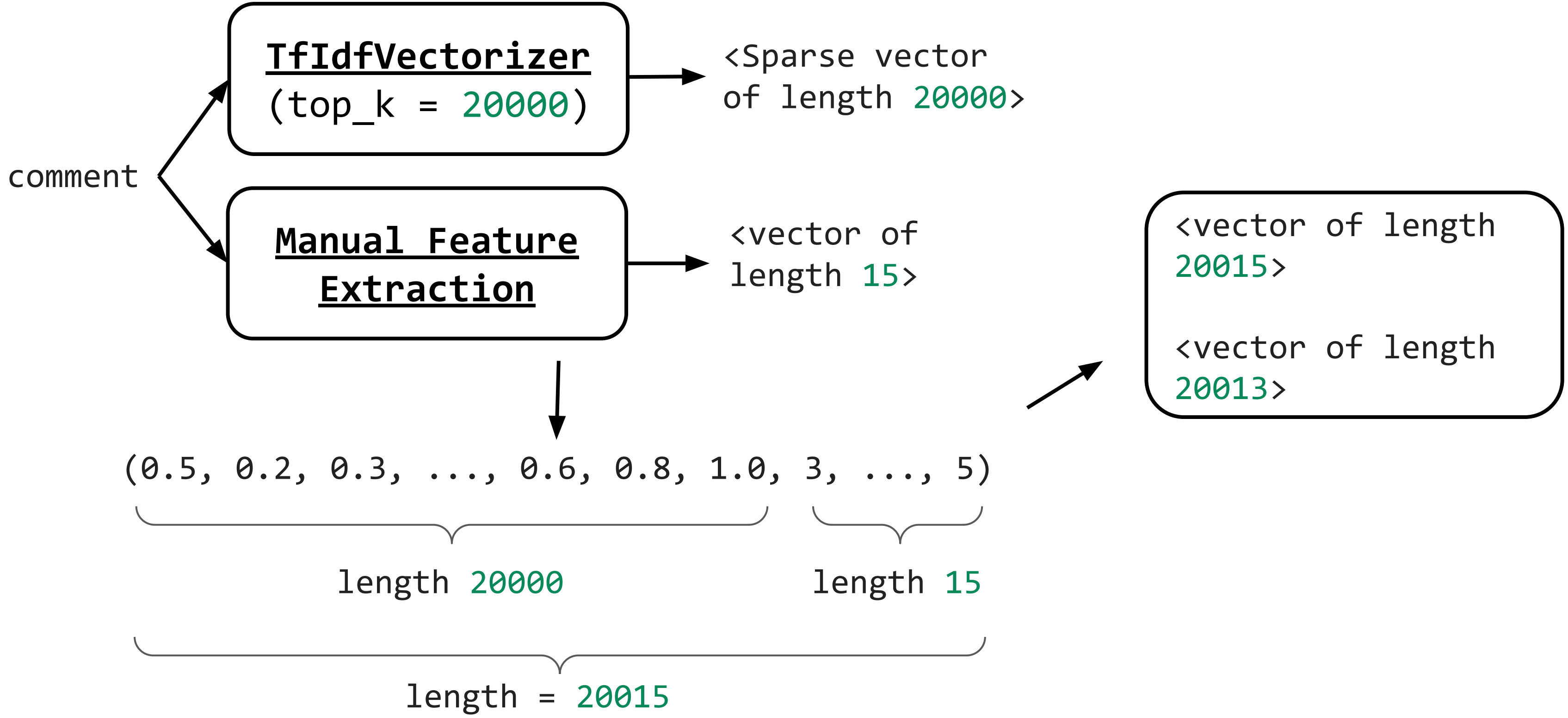
↓

| | | | | | |
|---|---|---|---|---|---|
| Nouns | 2 | Exclamation marks | 3 | Intensifiers | 3 |
| Verbs | 4 | Question marks | 0 | Positive Intensifiers | 0 |
| Positive Words | 2 | Uppercase letters | 3 | Negative Intensifiers | 3 |
| Negative Words | 11 | >3 Repeated letters | 0 | Emoticon Sentiments | 1 |
| Polarity Flips | 1 | Interjections | 1 | Sentiment Score | 0.7646 |

↓

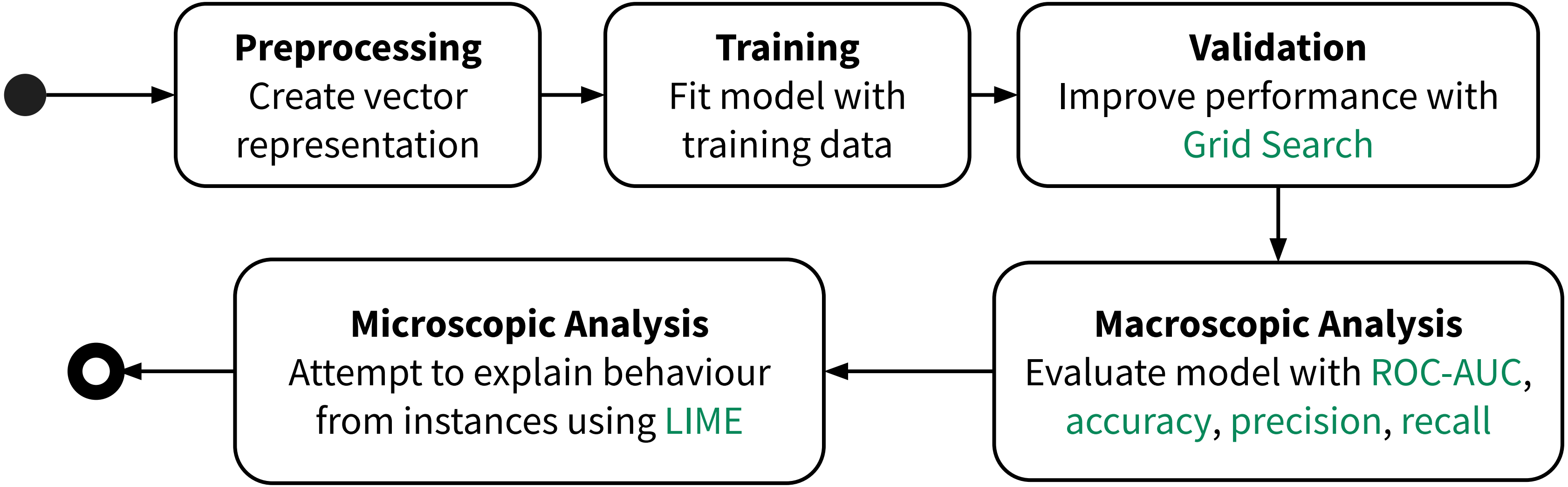
```
(2, 4, 2, 11, 1, 0.7646, 3, 0, 3, 0, 1, 3, 0, 3, 1)
```

# 6. Combining Manual Features and TF-IDF



# Models

# Technical Approach



```python
from sklearn.naive_bayes import MultinomialNB
```

# 1. Naive Bayes (NB)

# Macroscopic Analysis - Improving Performance

| Input Vectors | Hyperparameters | Training ROC-AUC | Testing ROC-AUC |
|---|---|---|---|
| **TF-IDF** | ngram_range=(1,1), alpha=1.0, fit_prior=True | 0.723 | 0.669 |
| **TF-IDF** | ngram_range=(1,1), alpha=1.0, fit_prior=False | 0.725 | 0.674 |
| **TF-IDF** | ngram_range=(1,3), top_k=20000, alpha=0.01, fit_prior=False | 0.719 | 0.705 |
| **TF-IDF + 15 Manual** | ngram_range=(1,3), top_k=1000000, alpha=0.3, fit_prior=False | 0.829 | 0.711 |
| **TF-IDF + 13 Manual** | ngram_range=(1,3), top_k=1000000, alpha=0.3, fit_prior=False | 0.830 | 0.712 |
| **TF-IDF** | ngram_range=(1,3), top_k=1000000, alpha=0.8, fit_prior=False | 0.827 | 0.717 |

# Macroscopic Analysis - TF-IDF Model

### Best hyperparameters

| | | | |
|---|---|---|---|
| ngram_range | (1,3) | top_k | 1000000 |
| alpha | 0.8 | fit_prior | False |

### Confusion Matrix

Actual

| Predicted | Sarcastic | Not Sarcastic |
|---|---|---|
| Sarcastic | 70408 | 27869 |
| Not Sarcastic | 30578 | 73300 |

### Classification Report (wrt. Sarcastic)

| | |
|---|---|
| Precision | 0.720 |
| Recall | 0.700 |
| F1-Score | 0.710 |

```python
from sklearn.linear_model import LogisticRegression
```

# 2. Logistic Regression (LR)

# Macroscopic Analysis - Improving Performance

| Input Vectors | Hyperparameters | Training ROC-AUC | Testing ROC-AUC |
|---|---|---|---|
| **15 Manual Features** | Default | 0.573 | 0.572 |
| **TF-IDF + 15 Manual** | C=1, ngram_range=(1,3), top_k=20000 | 0.748 | 0.649 |
| **TF-IDF + 15 Manual** | C=1000, ngram_range=(1,3), top_k=20000 | 0.686 | 0.685 |
| **Keras Embeddings** | Embed size=100 | 0.741 | 0.692 |
| **TF-IDF** | ngram_range=(1,3) top_k=20000 | 0.726 | 0.715 |
| **TF-IDF** | C=10, ngram_range=(1,3) top_k=20000 | 0.733 | **0.716** |

# Macroscopic Analysis - Manual Feature Model

**Worst input vector:** 15 Manual Features

**Confusion Matrix**

Actual / Predicted

| | Sarcastic | Not Sarcastic |
|---|---|---|
| Sarcastic | 34015 | 19421 |
| Not Sarcastic | 66971 | 81748 |

**Low recall** for Sarcastic

**High recall** for Non-Sarcastic

**For Sarcastic label:**

| | |
|---|---|
| Precision | 0.640 |
| Recall | 0.340 |
| F1-Score | 0.440 |

**For Non-Sarcastic label:**

| | |
|---|---|
| Precision | 0.550 |
| Recall | 0.810 |
| F1-Score | 0.650 |

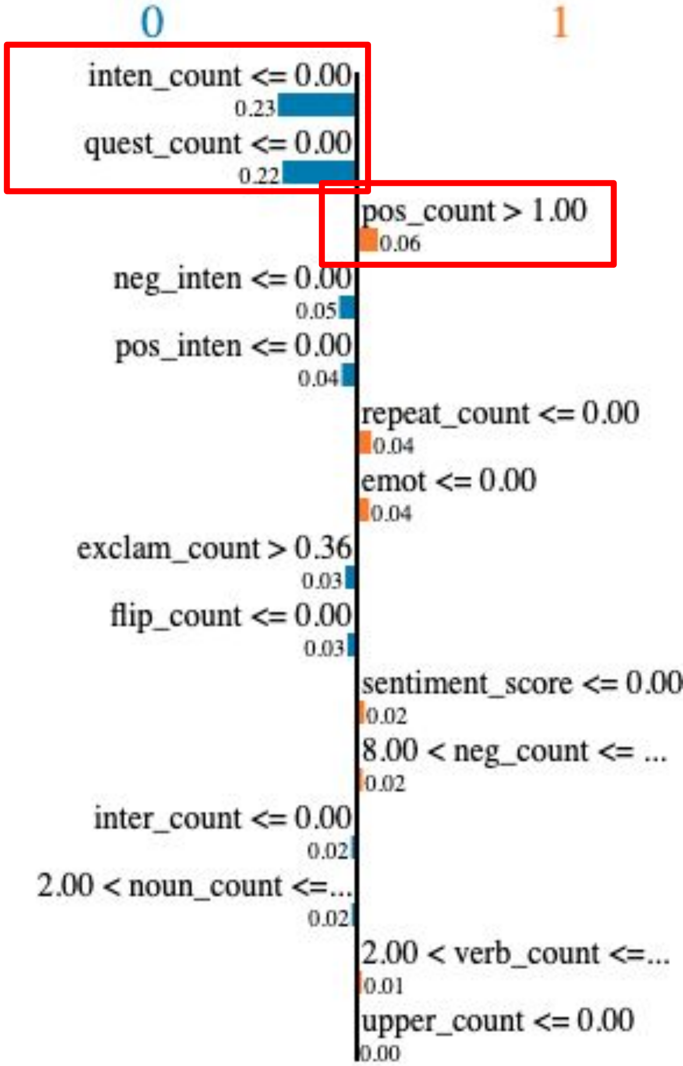
# Microscopic Analysis - FN

**Low %** of actual **sarcastic** comments detected.

*Why?* Performing LIME on a **false negative** case:

| Actually sarcastic; Predicted as non-sarcastic |
| --- |
| `Good to see my tax dollars are going to a good cause.` |

- LR model seems to place large focus on *intensifiers and questions.*
- *Positive words* not given a high weightage despite being an indication of sarcasm → possibly due to lack of context

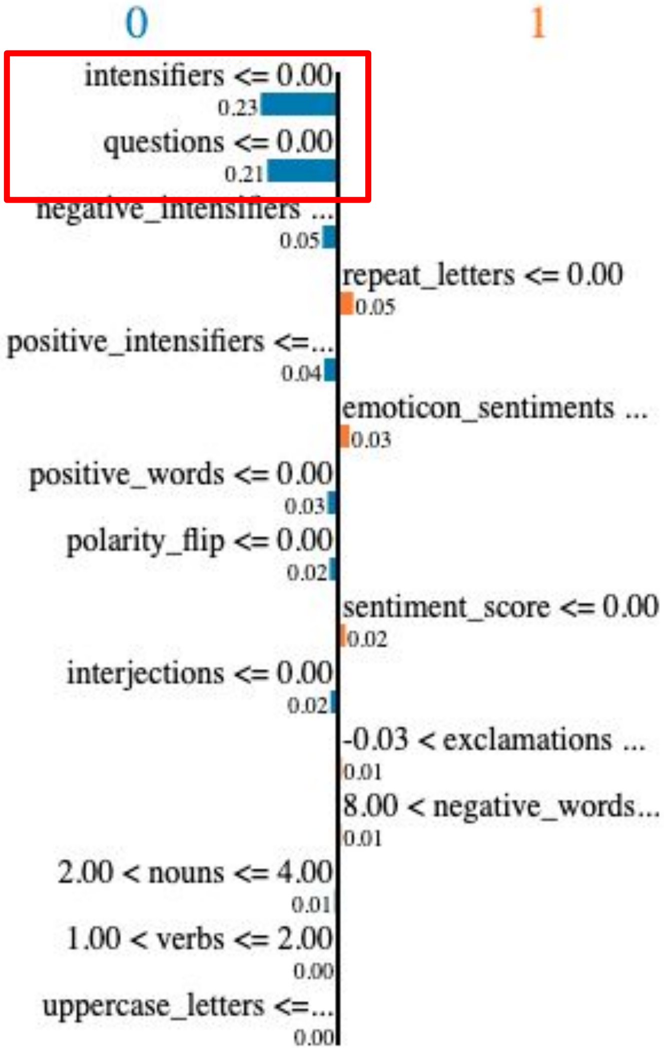


# Microscopic Analysis - TN

**High %** of actual **non-sarcastic** comments detected.
*Why?* Performing LIME on a **true negative** case:

| Actually non-sarcastic; Predicted as non-sarcastic |
|---|
| `In Newfoundland they're called 'Mother in law doors'.` |

- Lack of intifiers and questions
  → Likely to be a *factual statement*
  → Not Sarcastic!
- Small difference in probabilities between labels

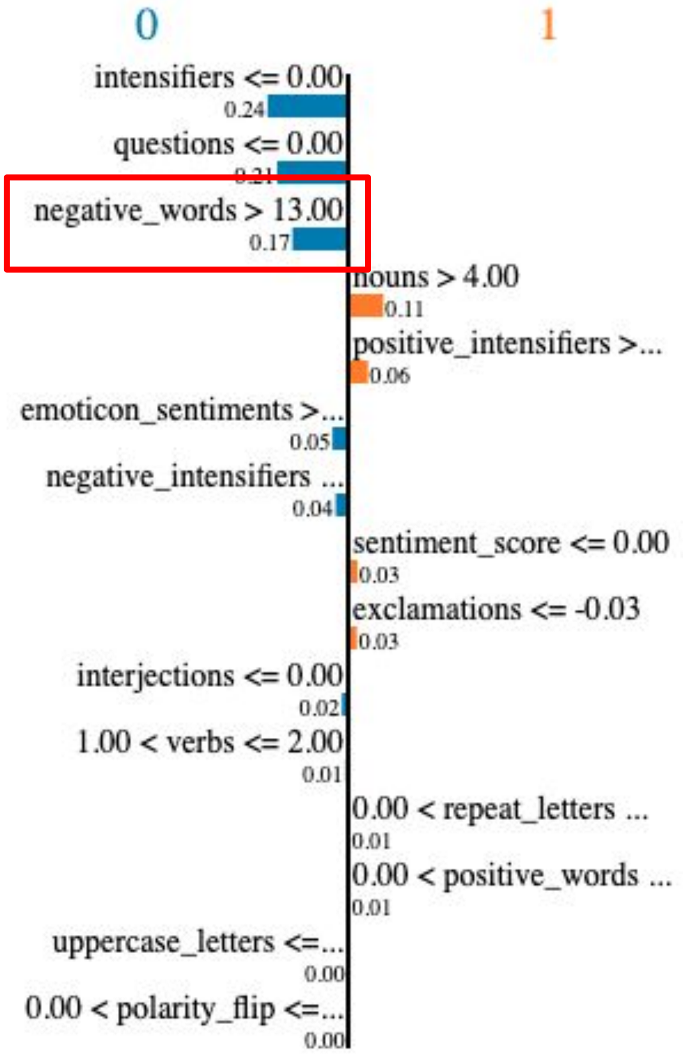
# Microscopic Analysis - TN

**High %** of actual **non-sarcastic** comments detected.

*Why?* Performing LIME on a **true negative** case:

| Actually non-sarcastic; Predicted as non-sarcastic |
|---|
| Yeah and France has a crappy army compared to Germany and the UK cant conduct naval invasions yet since its too early. |

- More negative words used
  → Likely to be an *honest opinion*
  → Not Sarcastic!



```
from sklearn.svm import SVC
```

# 3. Support Vector Machine (SVM)

# Macroscopic Analysis - Improving Performance

Likely the culprit. Failed to converge!

| Input Vectors | Hyperparameters | Training ROC-AUC | Testing ROC-AUC |
|---|---|---|---|
| **TF-IDF** | ngram_range=(1,3) top_k=50000, max_iter=20000 | 0.487 | 0.484 |
| **TF-IDF + 15 Manual** | ngram_range=(1,3) top_k=50000, max_iter=20000 | 0.517 | 0.518 |
| **15 Manual Features** | Default | - | 0.590 |

# Macroscopic Analysis - Manual Feature Model

**Analysed model:** 15 Manual Features, `max_iter=1000`

### Confusion Matrix

Actual (columns) / Predicted (rows)

| | Sarcastic | Not Sarcastic |
|---|---|---|
| Sarcastic | 84606 | 81351 |
| Not Sarcastic | 16380 | 19818 |

**Very high recall** for Sarcastic
**Very low recall** for Non-Sarcastic

### For Sarcastic label:

| | |
|---|---|
| Precision | 0.510 |
| Recall | 0.840 |
| F1-Score | 0.630 |

### For Non-Sarcastic label:

| | |
|---|---|
| Precision | 0.550 |
| Recall | 0.200 |
| F1-Score | 0.290 |

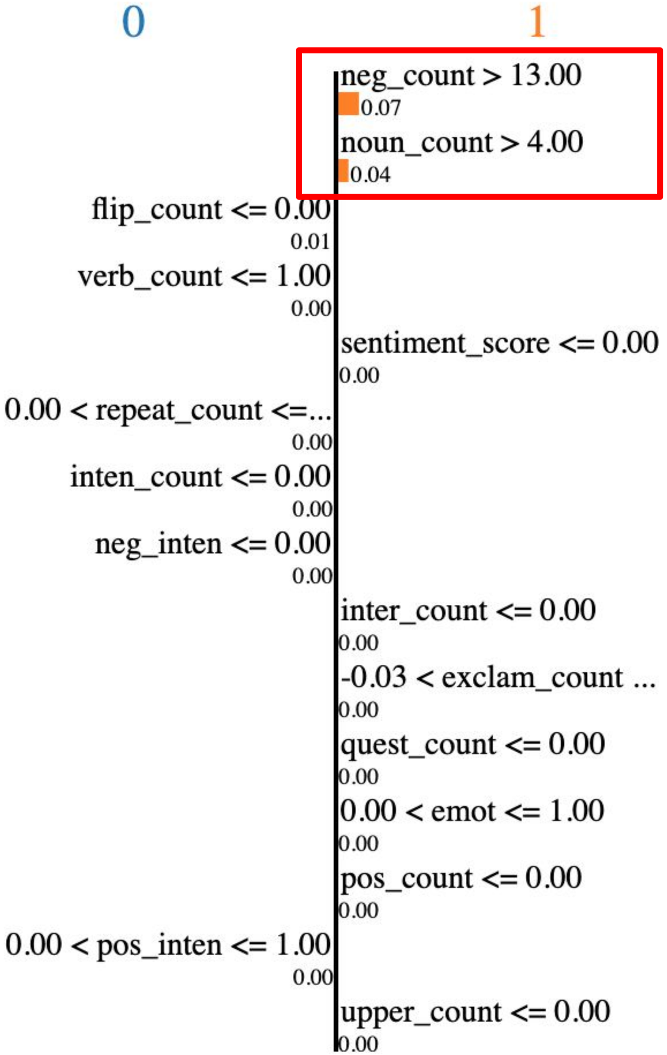
# Microscopic Analysis - TP

**High %** of actual **sarcastic** comments detected.

*Why?* Performing LIME on a **true positive** case:

| Actually sarcastic; Predicted as sarcastic |
| --- |
| `Viagra is a cover up for the real cause of ED, fluoride in the tap water` |

- SVM model regards *negative words* and *nouns* as indicators of sarcasm, unlike LR
- *Noun count* given a higher weightage

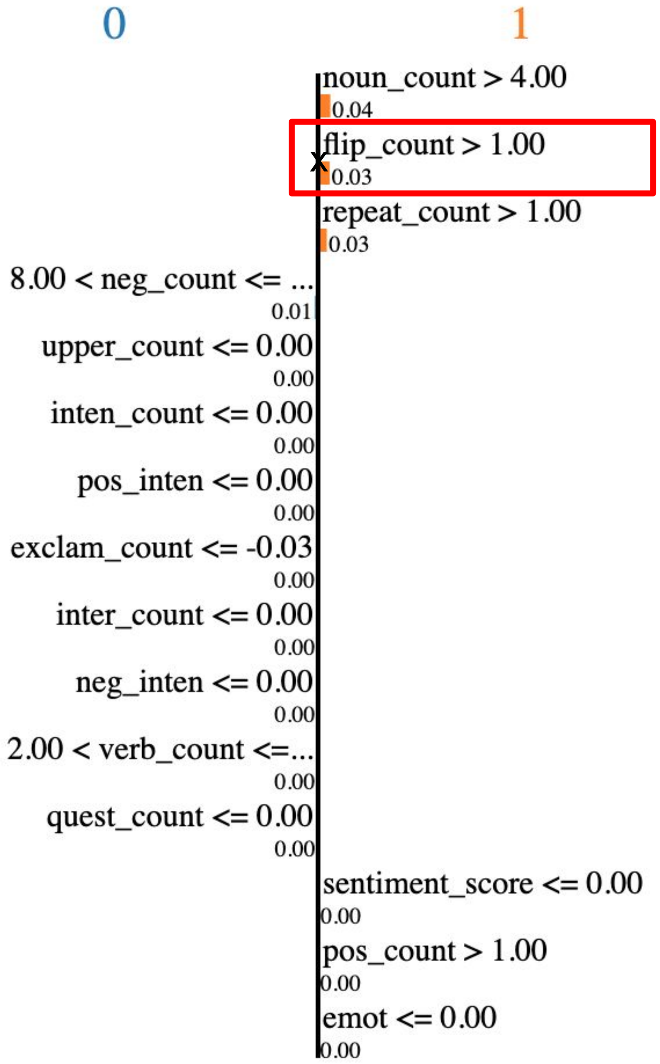


# Microscopic Analysis - TP

**High %** of actual **sarcastic** comments detected.
*Why?* Performing LIME on a **true positive** case:

| Actually sarcastic; Predicted as sarcastic |
|---|
| `Shit I had *excellent* credit when I was an 18 year old dumb ass, Gunny.` |

- *Polarity flips* are also well-detected

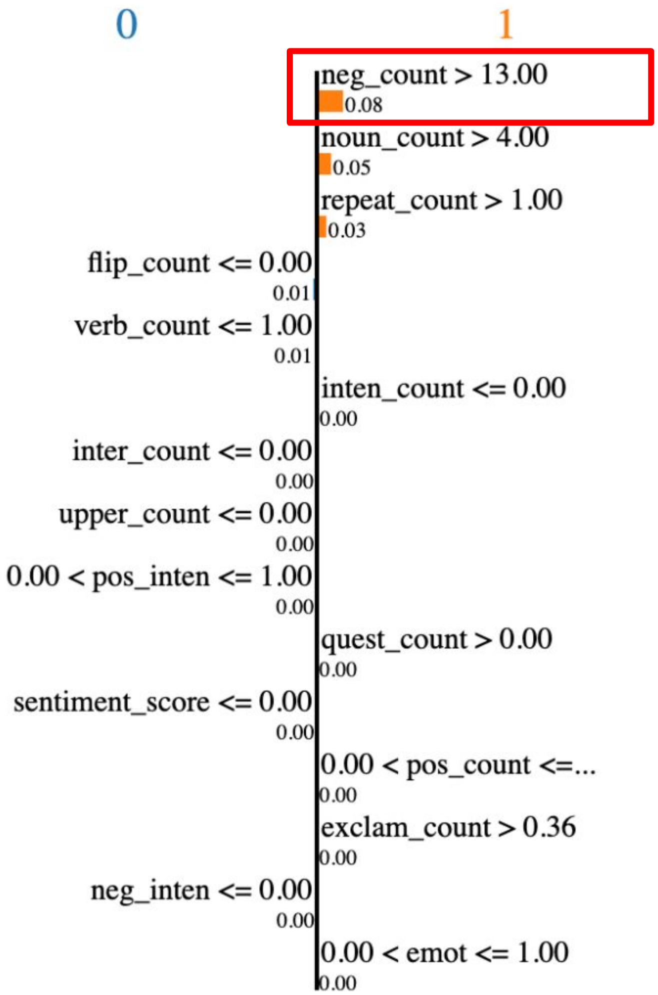


# Microscopic Analysis - FP

**Low %** of actual **non-sarcastic** comments detected.
*Why?* Performing LIME on a **false positive** case:

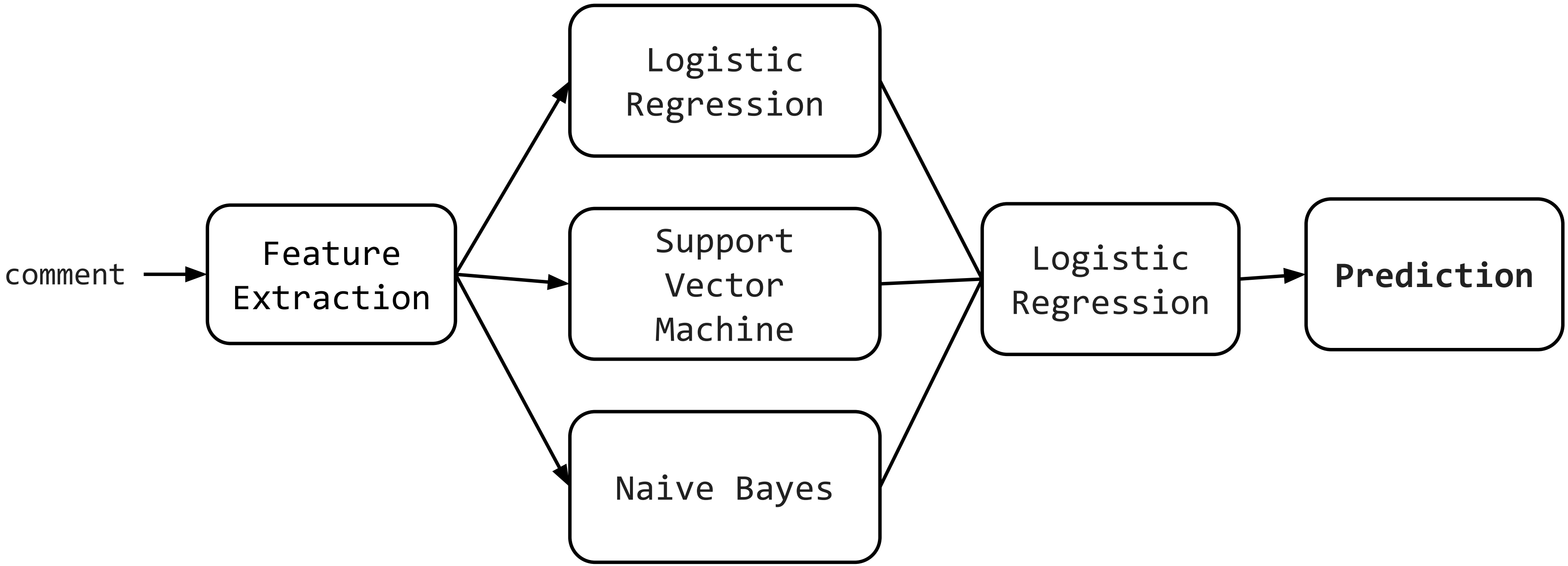
| Actually non-sarcastic; Predicted as sarcastic |
|---|
| `it is a sci[ENT]ific FACT that people just hand out PhD's in religious studies!` |

- SVM over-emphasizes on *negative words*
- Comments that are critical but non-sarcastic can be mislabelled



# 4. Stacked Model

# Stacking Classifier



# Macroscopic Analysis

| Model | Input Vectors | Hyperparameters | Training ROC-AUC | Testing ROC-AUC |
|---|---|---|---|---|
| Stacked | TF-IDF | ngram_range=(1,3) top_k=50000 | 0.743 | 0.719 |
| Stacked | TF-IDF + 13 Manual | ngram_range=(1,3) top_k=50000 | 0.744 | 0.725 |
| Stacked | 15 Manual Features | Default | 0.578 | 0.579 |
| Support Vector Machine | 15 Manual Features | Default | - | 0.590 |
| Logistic Regression | TF-IDF | C=10, ngram_range=(1,3) top_k=20000 | 0.733 | 0.716 |
| Naive Bayes | TF-IDF | ngram_range=(1,3), top_k=1000000, alpha=0.8, fit_prior=False | 0.827 | 0.717 |

# Macroscopic Analysis - TF-IDF + Manual Feature

**Analysed model:** TFIDF + 15 Manual Features

**Confusion Matrix**

Actual

| Predicted | Sarcastic | Not Sarcastic |
|---|---|---|
| Sarcastic | 76686 | 24483 |
| Not Sarcastic | 31197 | 69789 |

**Improved Metrics** for Sarcastic

**For Sarcastic label:**

| | |
|---|---|
| Precision | 0.73 |
| Recall | 0.73 |
| F1-Score | 0.73 |

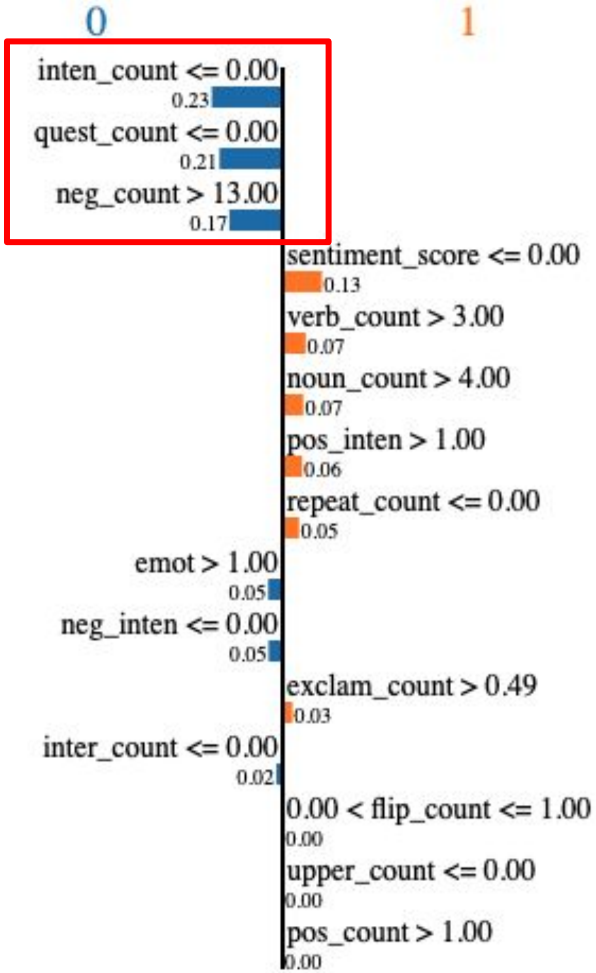
| | SVM | NB | LR |
|---|---|---|---|
| Precision | 0.510 | 0.72 | 0.72 |
| Recall | 0.840 | 0.70 | 0.72 |
| F1-Score | 0.630 | 0.71 | 0.72 |

# Microscopic Analysis - TN

*Why?* Performing LIME on a **true negative** case:

| Actually non sarcastic; Predicted as non sarcastic |
| --- |
| `Wow, amazing trade considering tochkin is a massive bust, and we only loose out on a 4th rounder for an already developed 23 year old, and ahl depth.` |

- Emphasis on intensifiers and questions (similar to logistic regression) for non sarcastic labels

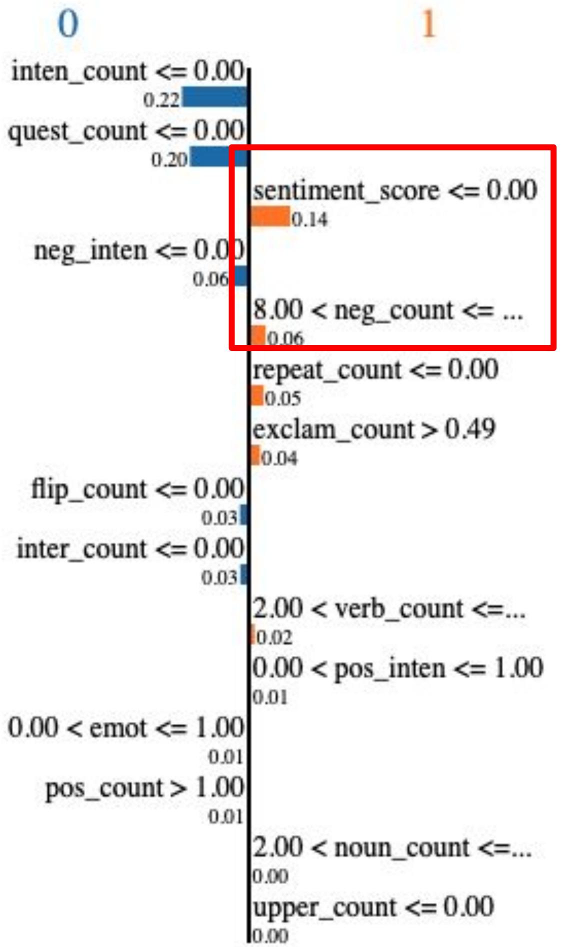


# Microscopic Analysis - TP

*Why?* Performing LIME on a **true positive** case:

| Actually sarcastic; Predicted as sarcastic |
|---|
| `Yeah because those people are running our government now and doing such a great job.` |

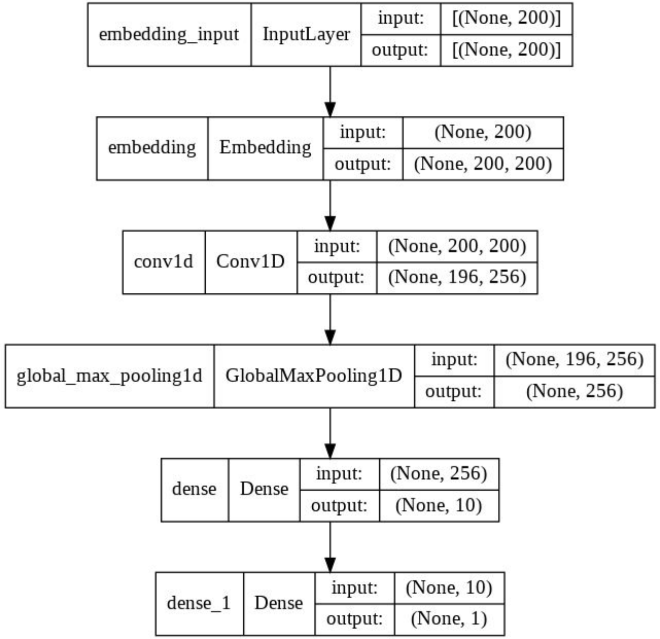
- Regards sentiment score and negative words as indicators of sarcasm (like SVM) though to a smaller extent



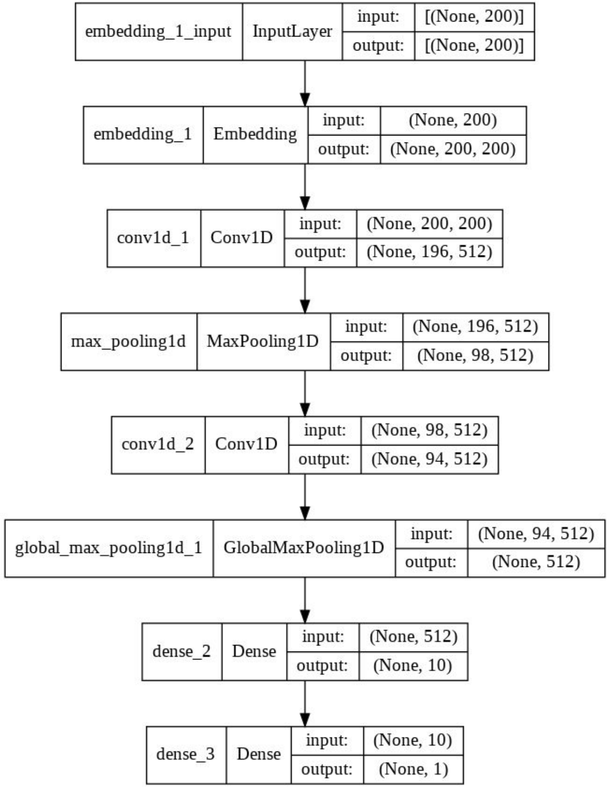
```python
from keras.models import Sequential
from keras import layers
```

# 5. Convolutional Neural Networks (CNN)

# CNN Architecture




CNN Model 1




CNN Model 2

# Macroscopic Analysis - Keras Embedding, 2 Models

**Hyperparameters:** epoch=1, vocab_size=160000, embedding_dim=200, maxlen=200

| Evaluation Metrics | Model 1 | Model 2 |
|---|---|---|
| Precision | 0.7748 | 0.7624 |
| Recall | 0.7092 | 0.7018 |
| F1 Score | 0.7406 | 0.7309 |
| ROC-AUC | 0.8327 | 0.8183 |
| Confusion Matrix | see below | see below |

Confusion Matrix — Model 1

| Actual / Predicted | Sarcastic | Not Sarcastic |
|---|---|---|
| Sarcastic | 320892 | 83344 |
| Not Sarcastic | 117601 | 286781 |

Confusion Matrix — Model 2

| Actual / Predicted | Sarcastic | Not Sarcastic |
|---|---|---|
| Sarcastic | 315816 | 88420 |
| Not Sarcastic | 120591 | 283791 |

# CNN with Word2Vec

Architecture

| | | | |
|---|---|---|---|
| main_input | InputLayer | Input | (None, 200) |
| | | Output | (None, 200) |
| embedding | Embedding | Input | (None, 200) |
| | | Output | (None, 200, 300) |
| conv1d | Conv1D | Input | (None, 200, 300) |
| | | Output | (None, 197, 50) |
| max_pooling_1d | MaxPooling 1D | Input | (None, 197, 50) |
| | | Output | (None, 98, 50) |
| conv1d_1 | Conv1D | Input | (None, 98, 50) |
| | | Output | (None, 96, 100) |
| max_pooling_1d_1 | MaxPooling 1D | Input | (None, 96, 100) |
| | | Output | (None, 48, 100) |
| flatten | Flatten | Input | (None, 48, 100) |
| | | Output | (None, 4800) |
| fully_connected | Dense | Input | (None, 4800) |
| | | Output | (None, 100) |
| dense | Dense | Input | (None, 100) |
| | | Output | (None, 2) |

| Hyperparameters | Training | Test |
|---|---|---|
| epoch=3, vocab_size=159591, embedding_dim=300, weights=W2V_embedding_matrix | 0.7924 | 0.7223 |

| Precision | Recall | F1 |
|---|---|---|
| 0.7223 | 0.7223 | 0.7223 |

| Actual / Predicted | Sarcastic | Not Sarcastic |
|---|---|---|
| **Sarcastic** | 39541 | 11177 |
| **Not Sarcastic** | 16891 | 33468 |

# Macroscopic Analysis - Keras Embedding

| Hyperparameters | Training Accuracy | Testing Accuracy |
|---|---|---|
| epoch=10, vocab_size=30000, embedding_dim=200, maxlen=100 | 0.8524 | 0.6922 |
| epoch=1, vocab_size=100000, embedding_dim=200, maxlen=100 | 0.7233 | 0.7116 |
| epoch=1, vocab_size=30000, embedding_dim=200, maxlen=100 | 0.7257 | 0.7123 |
| epoch=3, vocab_size=200000, embedding_dim=200, maxlen=200 | 0.8282 | 0.7213 |
| epoch=1, vocab_size=200000, embedding_dim=200, maxlen=200 | 0.7507 | 0.7262 |
| epoch=1, vocab_size=20000, embedding_dim=200, maxlen=200 | 0.7470 | 0.7264 |

# Macroscopic Analysis - Keras Embedding

**Analysed model:** epoch=1, vocab_size=20000, embedding_dim=200, maxlen=200

### Confusion Matrix

Actual

| Predicted | Sarcastic | Not Sarcastic |
|---|---|---|
| Sarcastic | 79763 | 21406 |
| Not Sarcastic | 34332 | 66654 |

### For Sarcastic label:

| | |
|---|---|
| Precision | 0.760 |
| Recall | 0.660 |
| F1-Score | 0.710 |

### For Non-Sarcastic label:

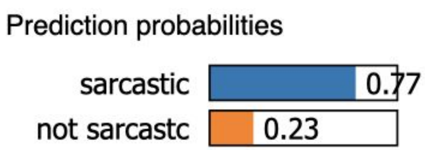
| | |
|---|---|
| Precision | 0.700 |
| Recall | 0.790 |
| F1-Score | 0.740 |

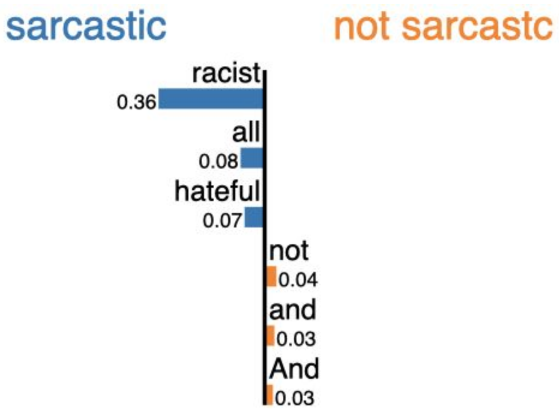
# Microscopic Analysis - TP

**High %** of actual **sarcastic** comments detected.
*Why?* Performing LIME on a **true positive** case:

| Actually sarcastic; Predicted as sarcastic |
|---|
| `And you are not at all racist and hateful.` |

- LIME gives the “weight” for each word in the embeddings that may contribute to sarcasm



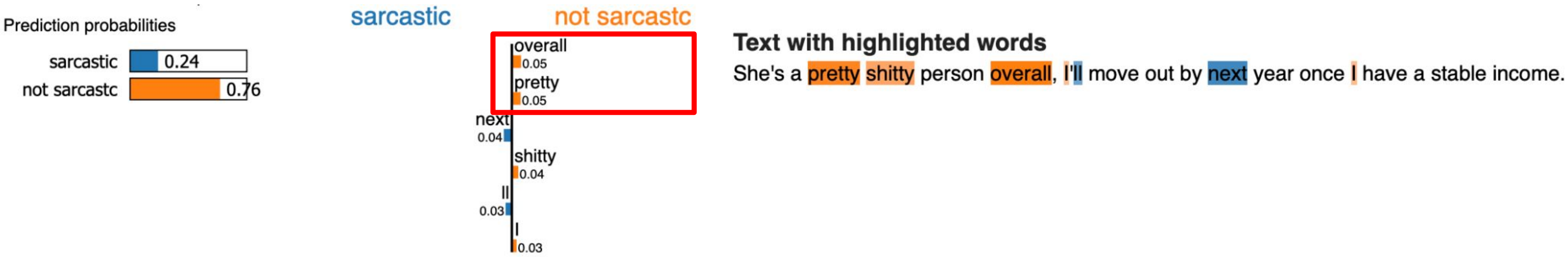

# Microscopic Analysis - TN

**High %** of actual **sarcastic** comments detected.

*Why?* Performing LIME on a **true negative** case:

| Actually not sarcastic; Predicted as not sarcastic |
| --- |
| `She's a pretty shitty person overall, I'll move out by next year once I have a stable income.` |

- LIME suggests that intensifiers in this sentence gives most contribution to the sentence being not sarcastic.



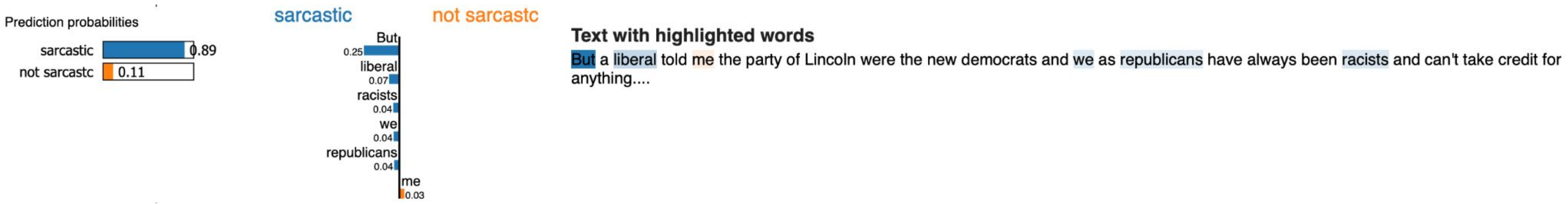
# Microscopic Analysis - FP

**%** of actual **non-sarcastic** comments detected.

*Why?* Performing LIME on a **false positive** case:

| Actually not sarcastic; Predicted as sarcastic |
|---|
| `But a liberal told me the party of Lincoln were the new democrats and we as republicans have always been racists and can't take credit for anything....` |

- The model over-emphasises on the words that are likely to be sarcastic
- For example, the same word 'racist' that contribute to sarcasm in the TP example before.



# Microscopic Analysis - FN

**%** of actual **sarcastic** comments detected.

*Why?* Performing LIME on a **false negative** case:

| Actually sarcastic; Predicted as not sarcastic |
| --- |
| `If everyone was a Trump supporter the world would be a better place.` |

- In this example, all word choices seem normal and non-sarcastic.
- The model fails as it does not know the real world context.

# Discussion

# Why do Manual Features work poorly?

- Our manual features were found to have **low linear correlation** with sarcasm, with an $R^2$ score of **0.036**.
    - Manual features are tailored to sarcasm context but perform worse than TF-IDF and embeddings
- Embeddings perform better in CNN
    - The embedding layer is trainable
    - Words likely contributing to sarcasm are assigned greater weights
    - Embeddings are able to capture specific words while manual features do not
- Manual features work better for SVM
    - Smaller dimensions, allowing SVM to converge much faster

# Answers to Our Questions

- Sarcasm detection without context is difficult even for humans - much less machines.
- **LR** and **SVM** are better at detecting non-sarcasm and sarcasm respectively
    - Ensembling for better overall performance
- **CNN** performs the best for detecting sarcasm without context
    - Marginally better
    - Longer training time
    - Overfitting and higher model complexity.
- **Recall** is a good metric to evaluate a model for sarcasm detection

# Future Research and Improvements

- Feed CNN output into SVM for final prediction
- Training the CNN with Doc2Vec embeddings
- Multicollinearity analysis on manual features

A baby learns to crawl, walk and then run. We are in the crawling stage when it comes to applying machine learning.

*Dave Waters*