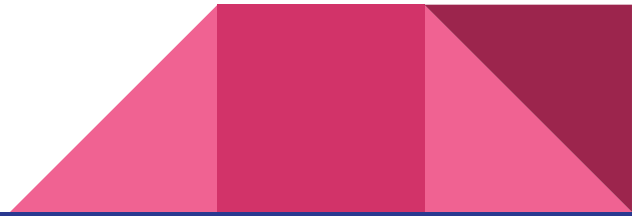
Cyberbullying Detection Model

Zeraiz Shabbir

Refresher

Previous goals:

- Develop a machine learning model that can detect cyberbullying in tweets
- Model can classify tweets by different types of cyberbullying (gender, religion, age, ethnicity, etc.)
- Achieve accuracy in classification



Design / Implementation

Data collection: Kaggle

Cloud environment: Databricks Community Edition

Preprocessing, model training, model evaluation: Apache Spark via PySpark API

Design / Implementation: Preprocessing

- Text cleaning
- Tokenization
- Stopwords removal
- Label encoding
- Sentiment analysis
- Trigram generation

Cmd 2

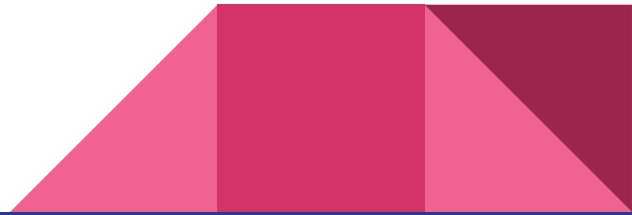
```
1  from pyspark.sql.functions import lower
2  from pyspark.ml.feature import Tokenizer, StopWordsRemover, StringIndexer, NGram, HashingTF, IDF
3  from pyspark.sql.functions import udf
4  from pyspark.sql.types import FloatType
5  from textblob import TextBlob
6  from pyspark.ml.classification import LogisticRegression
7  from pyspark.ml import Pipeline
8  from pyspark.ml.tuning import TrainValidationSplit
9  from pyspark import StorageLevel
```

Design / Implementation: Model training

Pipeline: Combined all processing steps into one workflow (preprocessing, feature transformation, model training)

Feature transformation: Converted trigrams to TF-IDF features, which helped weigh the importance of words in a tweet

Training algorithm: Logistic regression (statistical method used for binary classification)

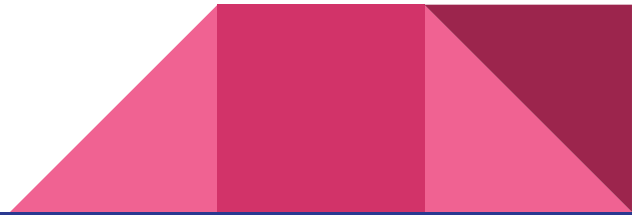


Setbacks / Challenges

Environment set up: Azure Databricks vs. Databricks community edition

Model training: Long training times made it hard to tweak preprocessing; high complexity of training for multiple categories

Memory issues: Running out of memory due to large dataset, had to adjust sample size and complexity



Experiment Results

Accuracy (proportion of correctly predicted instances among all instances):

0.6156711083843207

Area Under ROC (measures performance of model in distinguishing between positive and negative classes):

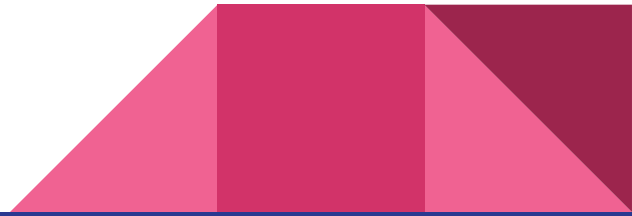
0.5880056777856636

F1 Score (average of both false positives and false negatives):

0.5883863403713325

Room for Improvement

- Model accuracy
- Training algorithm selection
- Further preprocessing
- Memory optimization



What have I learned?

- How to set up an environment in Databricks (cluster creation, DBFS usage, etc.)
- How to use preprocessing techniques to extract important information from a dataset
- Using Apache Spark’s MLlib to design, build, and train a model
- How to compute evaluation metrics of a model