

Project 3 – NLP

Group 2: Alex, Luis, Kat



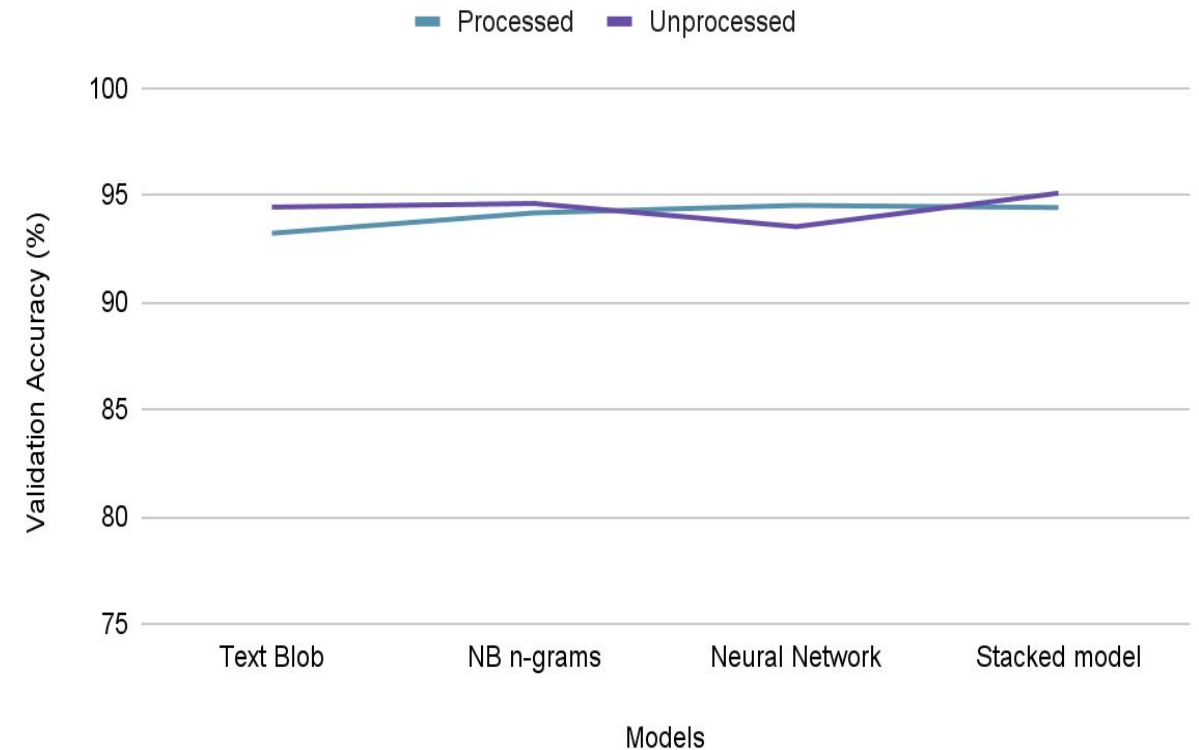
BERLIN

Stacked NB¹⁾ with Random Forest performed best for classifying news headlines – val. accuracy of 95.09%

Executive summary

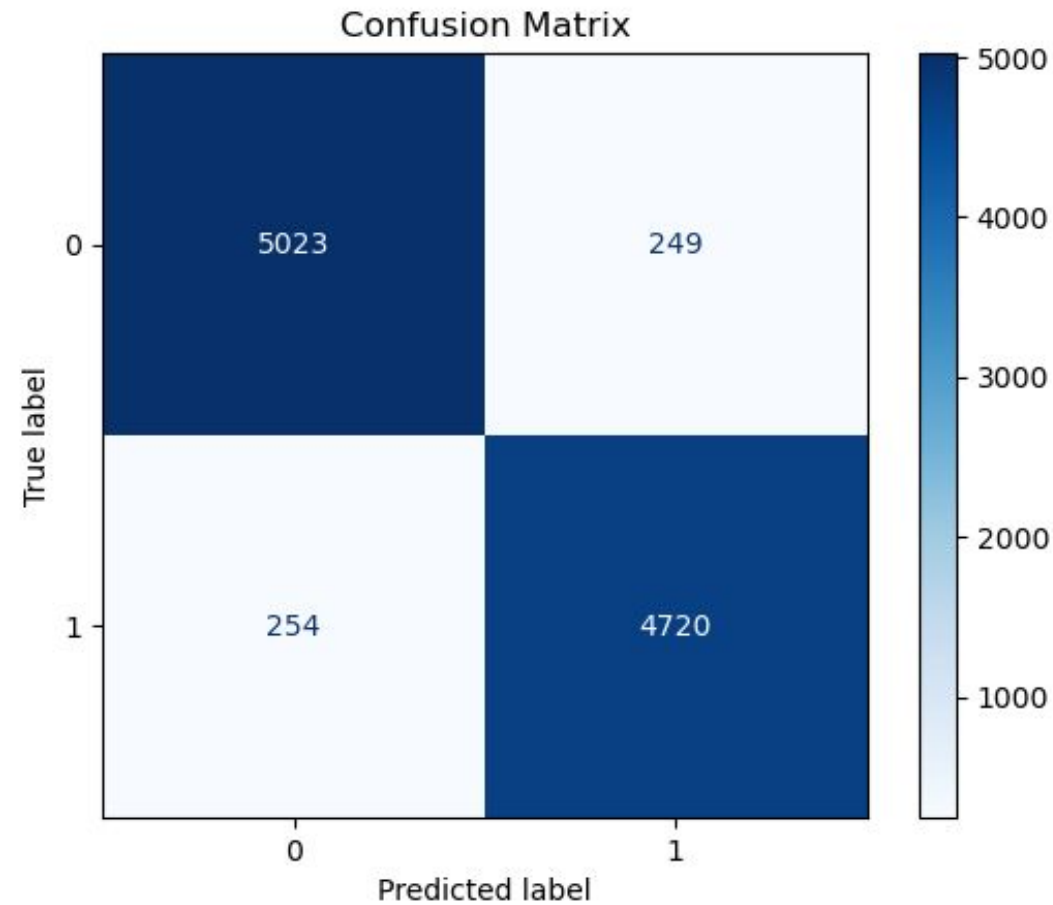
- Final validation accuracy: **95.09%**
- Model: **Stacked NB** model with Random Forest
 - Multinomial NB ○ Complement NB
 - Bernoulli NB ○ Random ForestMeta Classifier = Logistic Regression
- **Alternatives tested:**
 - Various types of Naive Bayes, incl. Bag of Words, n-grams, Voting Classifier
 - Other common models
 - Visualization techniques (UMAP, TSNE)

Best Models



1) NB = Naive Bayes 2) LogReg = Logistic Regression

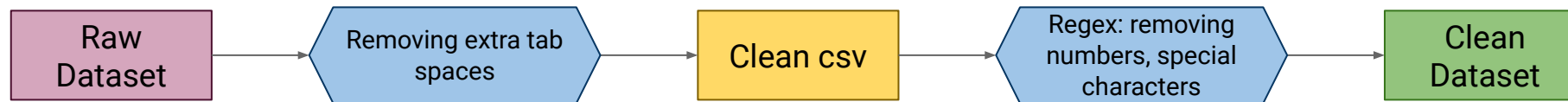
Confusion Matrix for our best model (validation set):



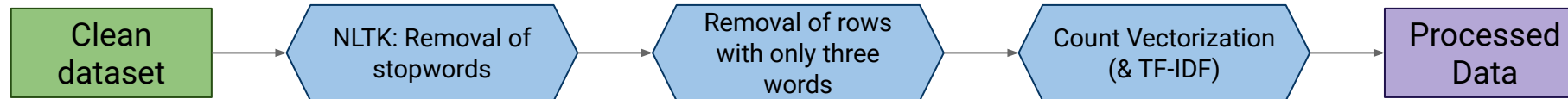
Best model on basic data cleaning – stopwords and TF-IDF worsened model performance

Preprocessing

Basic data cleaning (**best results**)



Full processing (**worse results**)



Additional techniques (**no significant improvement**):

- Lemmatization (worked well on K-Means and KNN without applying stopwords)
- Word2Vec and Word Embedding (discarded)

Many different models were tested, stacked model with best validation accuracy

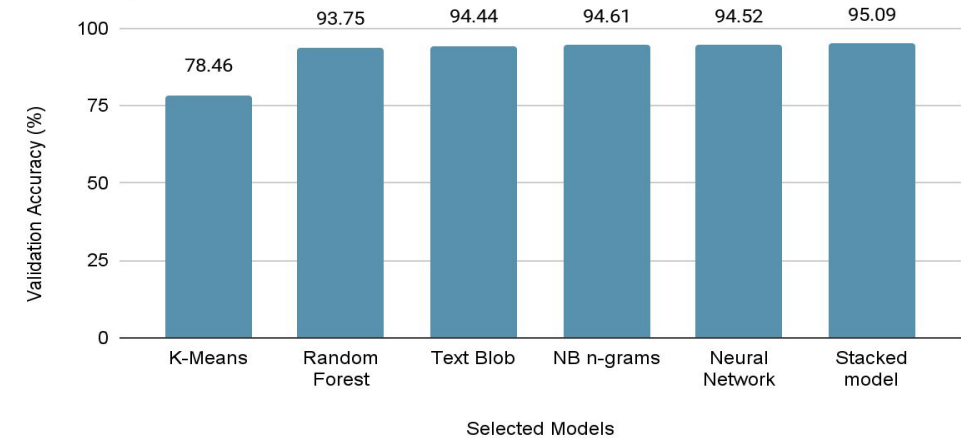
Model specifications

- Text Blob (needed some preprocessing)
- Naive Bayes with Bag of Words
- Naive Bayes with n-grams
- Models with TF-IDF:
 - Naive Bayes
 - Random Forest
 - K-Means, K-Nearest Neighbours (best with lemmatization, but without stopword)
 - SVC
 - UMAP, TSNE
- Others:
 - Word Embedding
 - Ensemble Voting (SVC, NB and LogReg¹⁾)
 - Ensemble Stacking (best)
 - Neural Network (2x Dense/BatchNorm/Dropout)

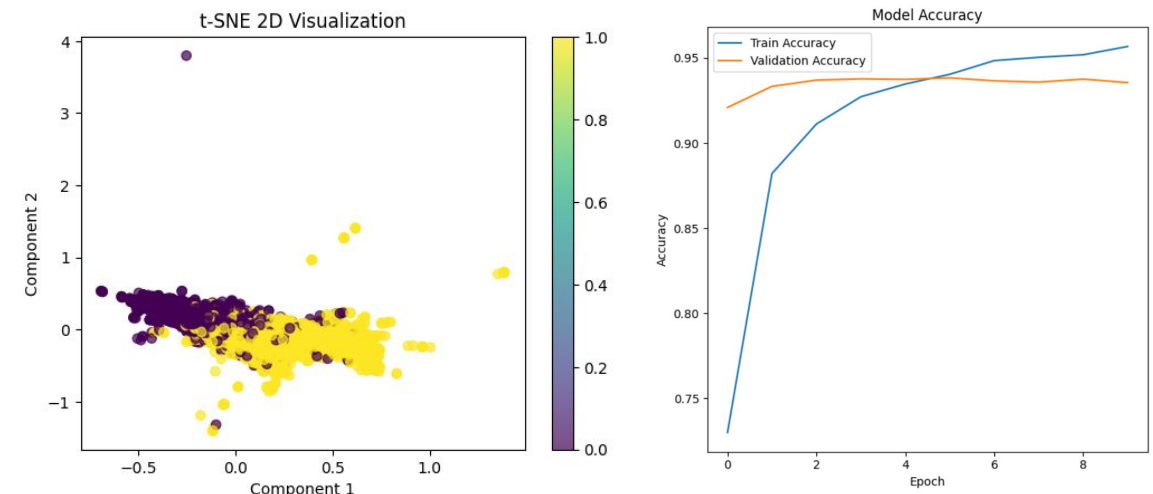
1) LogReg = Logistic Regression

Model Performance

Best configurations



Selected graphs other models



Takeaways

Key takeaways

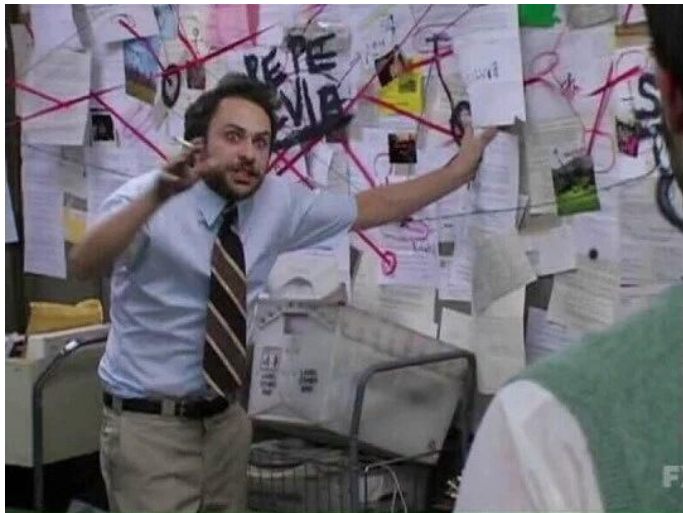
● Challenges:

- **Many model options** -> a lot of trial and error, choosing is difficult
- **Overthinking** techniques to use (tempting to apply EVERYTHING)

● Key learnings:

- Simple models **performed well** (in comparison to word embedding etc..)
- **No improvement** with tuned parameters -> only highly time consuming to run
- K-Means and KNN **performed worst**
- **Best results** with only basic preprocessing
- Stacking models can **improve performance**

Time for Questions



We haven't tried this parameter yet...

