Improving Emotion Detection Through Translation of Text to ML Models Trained in Different Languages

Qualifying Exam – Computational Data & Sciences

By Richard Hoehn

MTSU – August 2023

Welcome

Richard Hoehn

•

Living Franklin

•

MTSU Grad 2005, Vanderbilt Grad 2009

•

Last 15 years in SW development for Retail & Logistics Applications

•

Primarily in Databases, API / PubSub Integration Work, and Business Reporting Apps

Introduction & Agenda

Introduction

- Research on Emotion Detection (ED) in Text with an emphasis improving Prediction Rates
- Specifically by extending Data through <u>Translation to Different Languages</u>
- Training Multiple ML Models on Original & Extended Data to <u>Compare Prediction Rates</u>
- and Finally in Real-Time Translate Text to process in Parallel for further expansion

Agenda

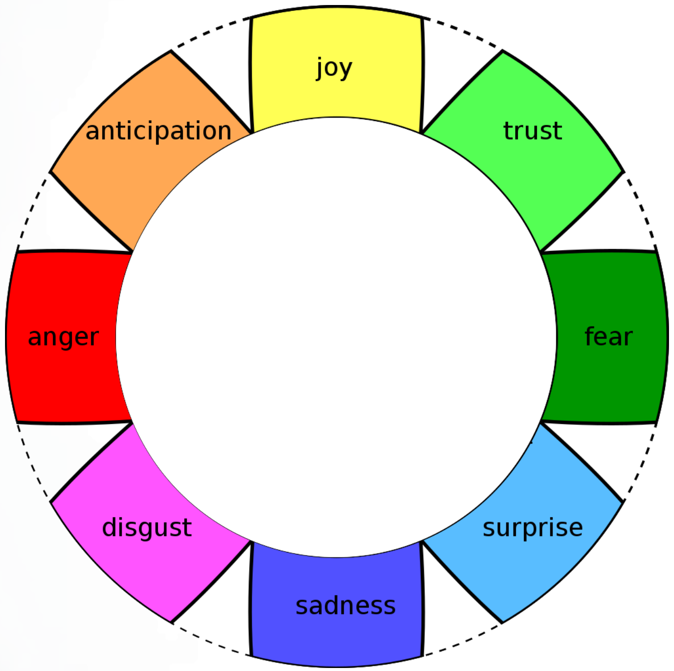
- Significance of Emotion Detection
- Challenges, Motivation and Scope of Research
- Methodology – Including Code Review and Demonstration!
- and finally Analysis of Results, Conclusion, and Future Work

Significance of Emotion Detection

Emotions can vary depending on the framework or model being considered.

One well-known model is the Plutchik's Wheel of Emotions, which proposes eight (8) primary emotions.

- Joy / Happiness
- Sadness
- Anger
- Worry / Fear
- Surprise
- Disgust / Hate
- Trust / Love
- Enthusiasm



By accurately identifying and understanding emotions from text data, ML applications can assist in improving:

- User Experiences (chat-bots),
- Decision-Making Processes
- and Overall human-machine interactions in a positive manner[4, 5], with most of these interactions being processed in real-time.

ED is still a growing field in Text, Video, and Image reading. The market for ED software and services is estimated to reach **$3.8billion**[2] by **2025.**

Challenges: Data Scarcity & Language Fragmentation

- Emotion detection data requires primarily supervised learning data! Meaning human curated and labeled data.
- Unlike Sentiment Analysis (SA) the availability of large datasets for training purposes of ML models is much smaller[3].
- Many datasets that are available are in many different languages
- Emotions in text are contextual in nature; meaning they conform in many cases based on the type and language and culture using them.

Motivation for Research and Evaluation of Dataset Extending for ML Prediction

- ED is still a growing field in Text, Video, and Image reading. The market for ED software and services is estimated to reach $3.8billion[2] by 2025.
- ED spans most all domains such like psychology, business, and education.
- By use of ML models the analysis of human emotions at scale provides valuable insights into individual and collective emotional states both in real-time but also for measuring sentiments from the past versus the current time.
- A recent paper[1] states that "***Emotions hold a paramount role in the conversation, as it expresses context to the conversation.***", this means that emotions are a part of a conversation and with that are needed to ensure valid analysis of a conversation.

Objectives & Scope of Research

The research project’s objectives were three-fold:

1. The first is to translate English data (feature & label) to German in order to extend the original German dataset for ML training purposes. With this will the added text lead to better predictions?
2. Similarly, can by translating German data to English and extending an original English dataset increase the predictability of English ML model?
3. And Lastly shifting the focus to real-time translation and its impact on prediction. Can by translating in real-time an input to multiple languages improve the predictability based on the combined output of two models.

In summary, this research project **investigated innovative ways** to enhance the predictability of Emotion Detection models in English and German.

With these three (3) objectives from above the scope was to **P**rocure, **T**ranslate, **T**rain, and **E**valuate benefits of extending datasets by translation to improve emotion detection.

Literature Review

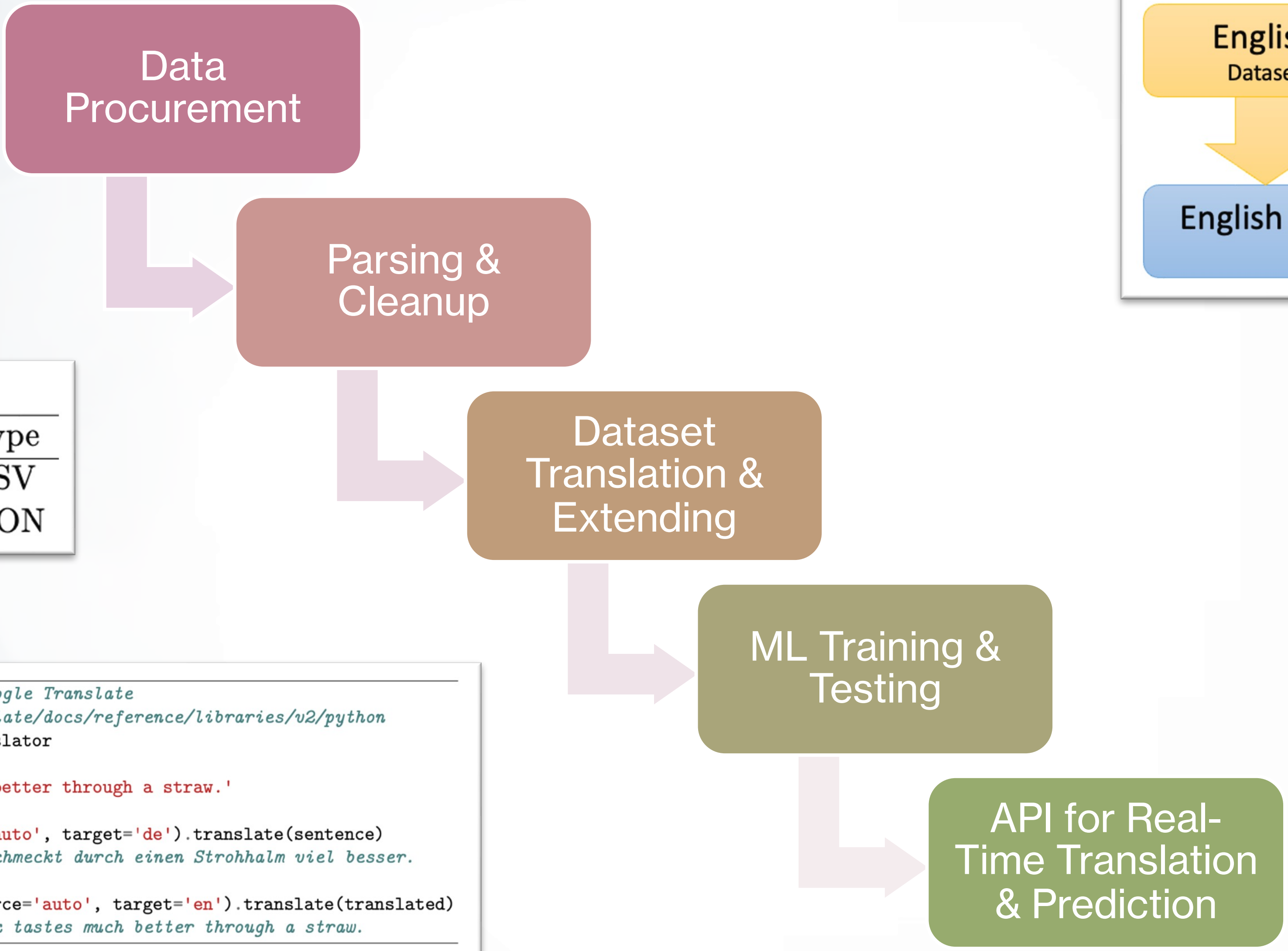
- Our research considered publications only past 2015, thereby providing an up-to-date perspective on ED analysis.

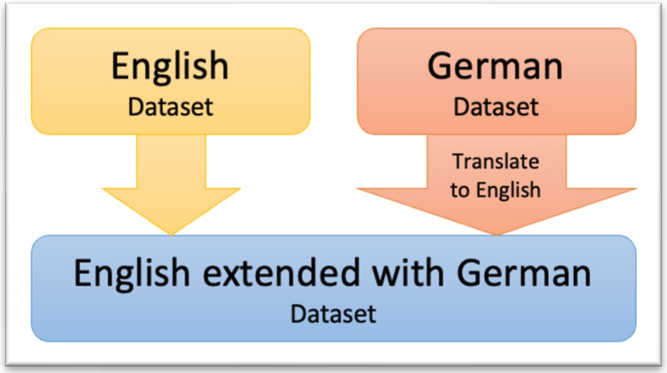
***"Emotions hold a paramount role in the conversation, as it expresses context to the conversation."* [1]**

- This means that emotions are a part of a conversation and with that are needed to ensure valid analysis of a conversation
- Emotions can differ across **A**ge groups, **G**enders, **C**ultures, and **L**anguages[6], which again support our approach of using 2015 and beyond data
- Fragmentation caused by different languages further exacerbates the issue of ED, as it reduces the size and diversity of data available for training, resulting in limited cross-lingual generalization and potentially biased models.[7]

Methodology

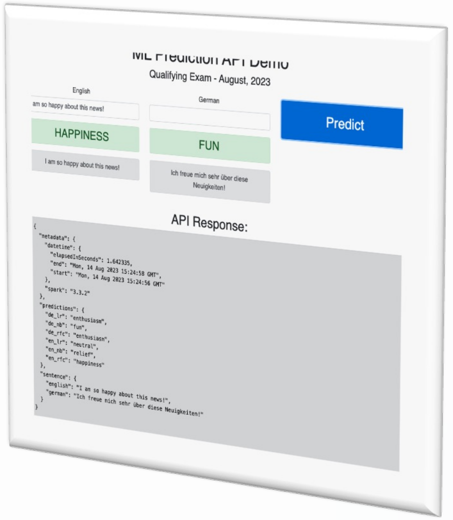
MIDDLE TENNESSEE STATE UNIVERSITY





File Details		
Name	Row Count	Type
English	38,000	CSV
German	2,500	JSON

```python
# Using Deep Translator to leverage Google Translate
# Link: https://cloud.google.com/translate/docs/reference/libraries/v2/python
from deep_translator import GoogleTranslator

sentence = 'Chocolate milk is so much better through a straw.'

translated = GoogleTranslator(source='auto', target='de').translate(sentence)
print(translated) # Schokoladenmilch schmeckt durch einen Strohhalm viel besser.

translated_back = GoogleTranslator(source='auto', target='en').translate(translated)
print(translated_back) # Chocolate milk tastes much better through a straw.
```



MIDDLE TENNESSEE STATE UNIVERSITY

Methodology
Data Procurement

- The data procurement was relatively straight forward and once found by using multiple Google search terms for the Emotion Detection in English and German text languages.
- The German data was obtained from the dataset built by ETH's Emotion and Stance Detection for German Text[4] based on voting feedback.
- The English dataset was downloaded from Kaggle[8] based on Tweets collected in 2021.
- Unfortunately, there was a large quantity difference between German and English!

File Details

Name	Row Count	Type
English	38,000	CSV
German	2,500	JSON

Emotion Datasets Labels

English Name	English Count	German Name	German Count	Used
Boredom	179	—	—	NO
Love	3842	Vertrauen	316	YES
Relief	1526	—	—	NO
Fun	1776	—	—	NO
Hate	1323	Ekel	29	YES
Neutral	8638	Unklar	314	YES
Anger	110	Ärger	226	YES
Happiness	5209	Freude	140	YES
Surprise	2187	Überraschung	369	YES
Sadness	5165	Traurigkeit	184	YES
Worry	8459	Angst	154	YES
Enthusiasm	759	Antizipation	774	YES
Empty	827	—	—	NO

Methodology
Parsing, Cleanup, & Emotion Linkage

- By use of Jupyter Notebooks the English (csv) and German (JSON) files were read and processed ease of demo.
- For processing I opted to use Pandas over PySpark.
- Google Translator was used for Translation – Their API is Free!

```
# Emotion Panda DataFrame
# This was predetermined by review of the emotion from English to German
emotion_key = {
"boredom": "---",
"love": "Vertrauen",
"relief": "---",
"fun": "---",
"hate": "Ekel",
"neutral": "Unklar",
"anger": "Ärger",
"happiness": "Freude",
"surprise": "Überraschung",
"sadness": "Traurigkeit",
"worry": "Angst",
"enthusiasm": "Antizipation",
"empty ": "---",
"---": "Keine"
}

# Create a DataFrame from the emotion_key dictionary
df_emotions = pd.DataFrame(emotion_key.items(), columns=['emotion_en', 'emotion_de'])
```

```
# This google API take a Sentence and converts to German
def translate(sentence, dest_lang):
try:
translator = Translator()
translator.raise_Exception = True
translation = translator.translate(sentence, dest=dest_lang)
time.sleep(0.5) # Add a delay (This is due to rate limit of 1/s)
return translation.text
except Exception as e:
print(f"Translation Error: {e}")
return None
```

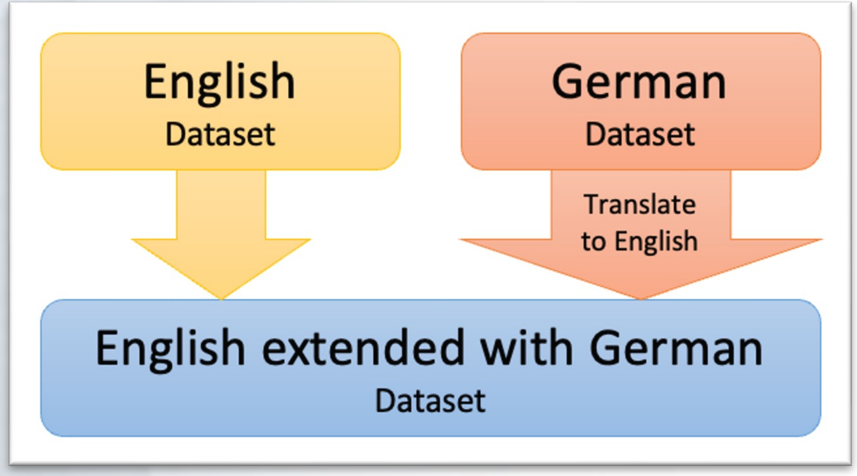
MIDDLE TENNESSEE STATE UNIVERSITY

Methodology
Translation Application

```
# 1
# Merge German Emotions onto English
df_en = pd.merge(df_en, df_emotions, on='emotion_en', how='left’)

# 2
# Add German Sentence Column
df_en["sentence_de"] = ""

# 3
# Randomly select 1500 rows
df_en = df_en.sample(n=1500, random_state=2023)

# 4
# Save original to Disk
df_en.to_csv('./data/pd_en.csv', index=False)


# 5
# Iterate over the rows with tqdm to show the progress
for index, row in tqdm(df_en.iterrows(), total=df_en.shape[0]):
    # 6
    # Call Translation
    sentence = translate(row["sentence_en"], 'de’) # To German (‘de’)


    # 7
    # Save Sentence on Column
    df_en.at[index, 'sentence_de'] = sentence

# 8
#Save the file with all the translations!
df_en.to_csv('./data/pd_en_translated.csv', index=False)
```

Methodology
Dataset Extension

```
# Split the English and German Dataframes for Training and Testing
# We are using a 20/80 Split
df_en_train, df_en_test = df_en.randomSplit([0.85, 0.15], seed=2023)
df_de_train, df_de_test = df_de.randomSplit([0.85, 0.15], seed=2023)
print(f"English Train Row Count: {df_en_train.count()}")
print(f"German Train Row Count: {df_de_train.count()}")


# Create the Extended Dataframe wiht Translated Data
df_en_train_extended = df_en_train.union(df_de.select(*df_en_train.columns))
df_de_train_extended = df_de_train.union(df_en.select(*df_de_train.columns))
print(f"English Extended Train Row Count: {df_en_train_extended.count()}")
print(f"German Extended Train Row Count: {df_de_train_extended.count()}")
```

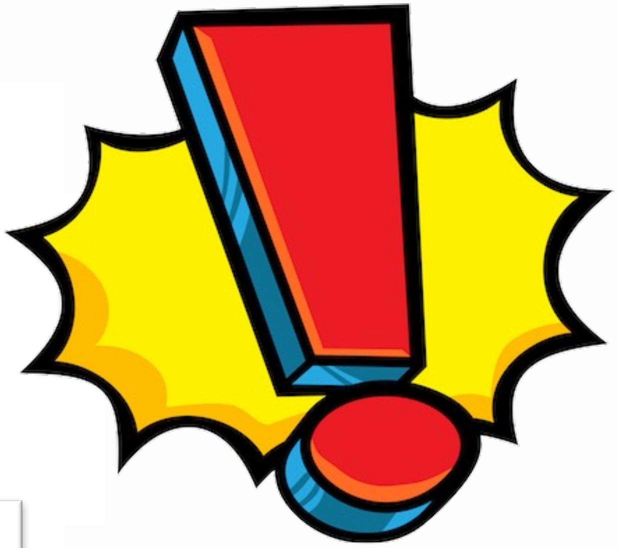


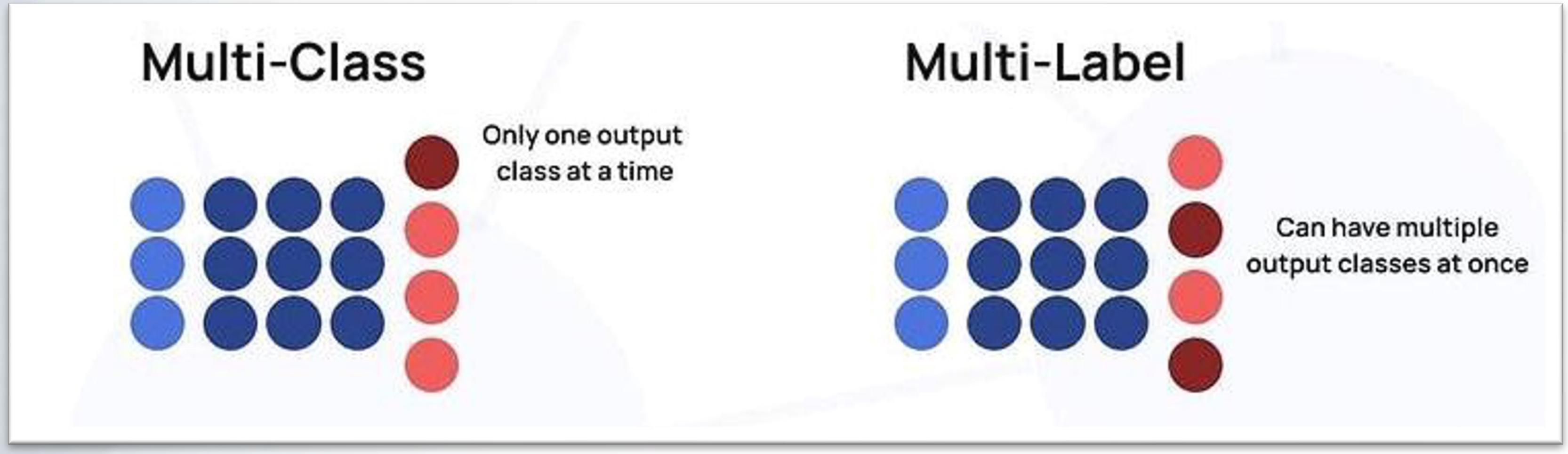
ML Models			
Model	Data	Rows for Training (85%)	Rows for Testing (15%)
A	English Original	1,275	225
B	English Extended By German	2,775	225 same as Model "A"
C	German Original	1,275	225
D	German Extended By English	2,775	225 same as Model "C"

MIDDLE TENNESSEE STATE UNIVERSITY

Methodology
ML Training & Testing with PySpark Multi-Class Classification

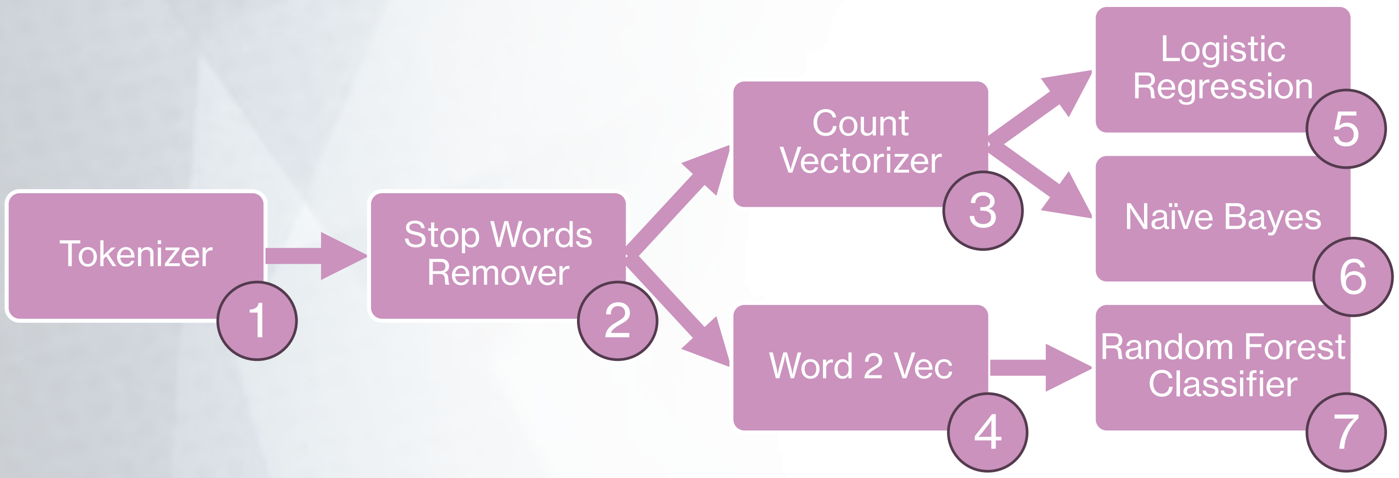
- Due to the eight (8) emotions present our research we decided to build of type **Multi-Class Classification Model**
- **This is due to the research scope to be in search of predictive improvement and measuring a single class of emotion is more distinct than using Multi-Label Classificaiton**





MIDDLE TENNESSEE STATE UNIVERSITY

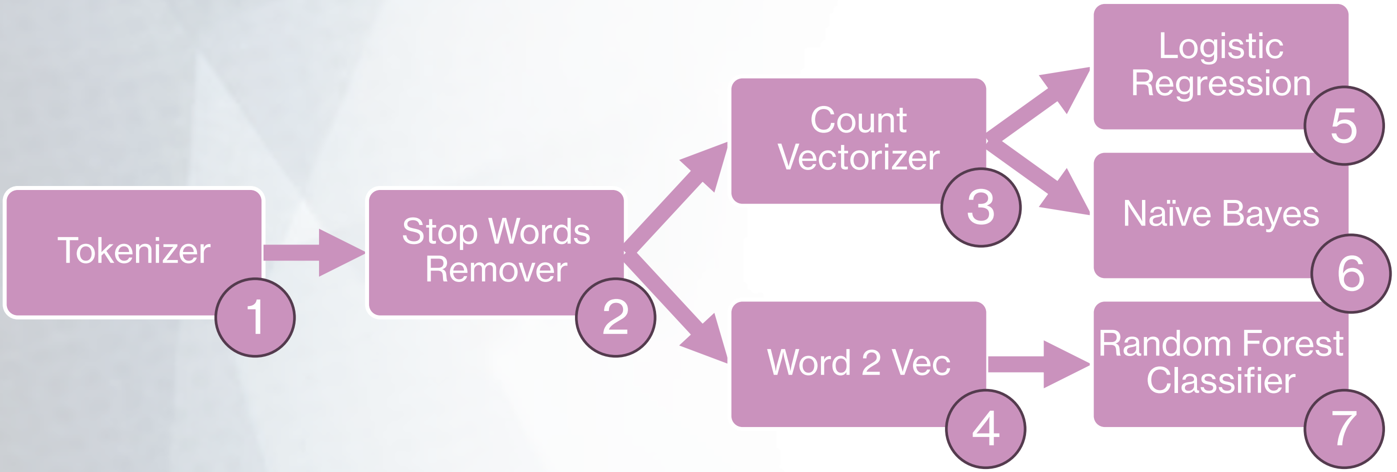
Methodology
ML Training & Testing with PySpark Pipeline: 1



1 - Tokenization

- Used DistilBERT (multi-language) for word tokenization.
- We used the DistilBERT from Hugginface's Transformers[9] due to it's download size and multi-language features.
- We opted to not use case sensitive tokenization due to the nature of tweets generally not following capitalizations and us mixing English with German.

Methodology
ML Training & Testing with PySpark Pipeline: 2, 3, & 4



2 - Stop Words Remover

Removing words that occur commonly across the dataset.

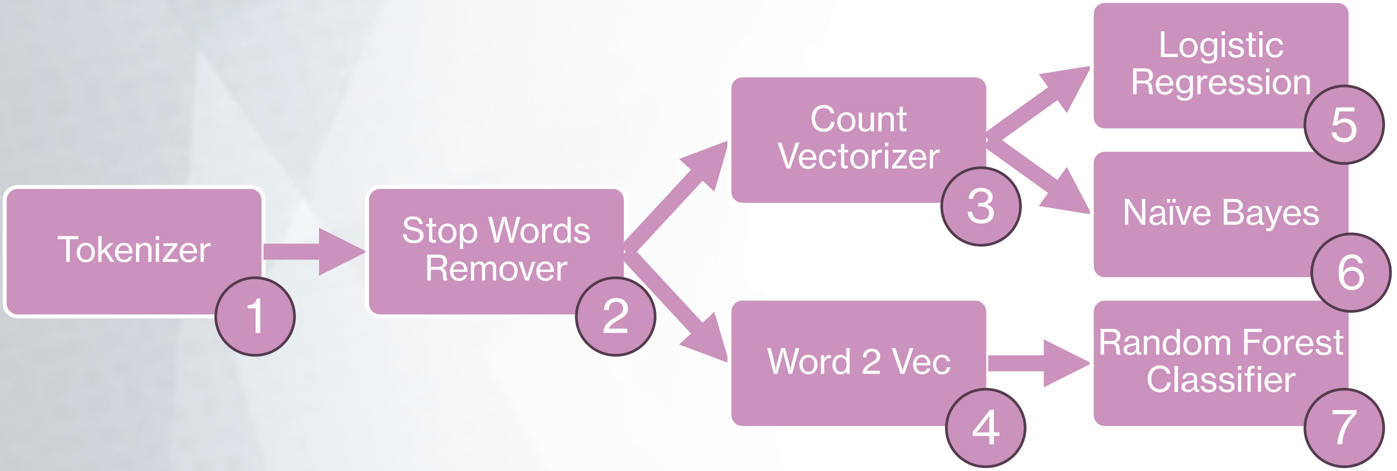
3 - Count Vectorizer

Is a method to convert text to numerical data.

4 - Word 2 Vec

Is a way to group the vectors of similar words together in order to detect similarities mathematically. This was only used on Random Forest Classifier

Methodology
ML Training & Testing with PySpark Pipeline: 5, 6, & 7



Learning Models – All Supervised

5 – Logistic Regression
It uses a logistic function to model the dependent variable. Standard approach but not optimal for Multi-Class approach.

6 – Naïve Bayes
This model is a probabilistic machine learning model that's used for classification task.

7 – Random Forest Classifier
RFC consists of a large number of individual decision trees that operate as a group. Each individual tree spits out a class prediction and the class with the most votes becomes our model's prediction.

MIDDLE TENNESSEE STATE UNIVERSITY

Methodology

Creating an API for Real-Time Testing

In order to test Real-Time translation a simple API was created that exposed a **GET** method and processed a query-string.

The Server Performed:

1. Setting up a Flask Python Server
2. Loaded the Saved ML models
3. And processed by translation a query-string
4. Return the predicted values and translated sentence via JSON back to the calling client.


```json
{
  "metadata": {
    "datetime": {
      "elapsed": 1.651038,
      "end": "Mon, 14 Aug 2023 12:10:35 GMT",
      "start": "Mon, 14 Aug 2023 12:10:33 GMT"
    },
    "spark": "3.3.2"
  },
  "predictions": {
    "de_lr": "enthusiasm",
    "de_nb": "relief",
    "de_rfc": "enthusiasm",
    "en_lr": "neutral",
    "en_nb": "love",
    "en_rfc": "happiness"
  },
  "sentence": {
    "english": "What a wonderful world this is!",
    "german": "Was für eine wundervolle Welt das ist!"
  }
}
```


DEMO

Demo & Code Review

Results & Analysis

- We used the Accuracy and F1 Score for our analysis on the results.
- Taking the F1 Score into is important on Multi-Class models and therefore got higher importance given
- **Recall & Precision – Multi-Class**
  - Due to the Multi-Class nature, we used the Micro Average on the Recall & Precision
  - This in turn gave us an accurate F1 Score
- F1 Score because it assigns equal weight to each emotion regardless of the class label and the number of labels.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Precision_{MicroAvg} = \frac{(TP_1 + TP_2 + \ldots + TP_n)}{(TP_1 + TP_2 + \ldots + TP_n + FP_1 + FP_2 + \ldots + FP_n)}$$

$$Recall = \frac{TP}{TP + FN}$$

$$Recall_{MicroAvg} = \frac{(TP_1 + TP_2 + \ldots + TP_n)}{(TP_1 + TP_2 + \ldots + TP_n + FP_1 + FP_2 + \ldots + FP_n)}$$

$$F1_{Score} = \frac{2 * Precision * Recall}{Precision + Recall}$$

Results & Analysis

Prediction Results of Original & Extended Datasets

- Very low Prediction Rates were achieved during testing! Disappointing
- Negative Improvements
  - Both on Accuracy
  - and F1 Score
- Initally the GLM predicts better but the F1 Score is higher on RFC

Results of English Dataset

	Original Dataset		Extended Dataset		
	Accuracy	F1 Score	Accuracy	F1 Score	Improvement
Random Forrest	29.72%	25.83%	21.23%	19.43%	-28.56%
Naïve Bayes	5.66%	6.92%	3.77%	4.14%	-33.39%
GLM (lr)	32.08%	23.68%	25.47%	16.19%	-19.95%

Results of German Dataset

	Original Dataset		Extended Dataset		
	Accuracy	F1 Score	Accuracy	F1 Score	Improvement
Random Forrest	28.10%	16.84%	25.71%	18.56%	-8.50%
Naïve Bayes	4.29%	4.31%	6.67%	4.21%	+55.48%
GLM (lr)	34.29%	17.57%	24.76%	18.27%	-27.79%

$$Improvement(\%) = \frac{(Accuracy_{Ext}(\%) - Accuracy_{Org}(\%))}{Accuracy_{Org}(\%)}$$

Results & Analysis
Analysis of Original Dataset

Overall the prediction results from all trained models is Disappointing.

Although in the previous slide the GLM model predicted better on both German & English, I believe that:

- **<u>NB on German</u>** performed better due to the distribution of TP values
- **<u>RFC on English</u>** performed better due to the distribution of TP values

There's an absence of context for the models to operate effectively and grasp the intricate nuances of both the English and German languages.

Secondly, in cases where class data is lacking, the models tend to <u>memorize existing</u> sentences[10], rather than <u>learn</u> the underlying patterns of the language.

ML Models Test versus Test Result Counts

English

ID	Label	Test Data	NB Res	GLM Res	RFC Res
0	boredom	0	25 [11]	0	0
1	love	26	8	3	11
2	Relief	0	33	0	0
3	Fun	0	5	0	0
4	hate	9	31	0	0
5	neutral	50	21	152	101
6	anger	0	36	0	0
7	happiness	39	44	5	17
8	surprise	10	9	0	1
9	sadness	29	0	2	19
10	worry	46	0	50	63
11	enthusiasm	3	0	0	0

German

ID	Label	Test Data	NB Res	GLM Res	RFC Res
0	boredom	0	34	0	0
1	love	25	5	0	13
2	Relief	0	28	0	0
3	Fun	0	16	0	0
4	hate	5	16	0	2
5	neutral	30	24	0	5
6	anger	23	20	1	7
7	happiness	10	20	0	5
8	surprise	24	47	0	11
9	sadness	12	0	0	2
10	worry	9	0	209	4
11	enthusiasm	72	0	0	161

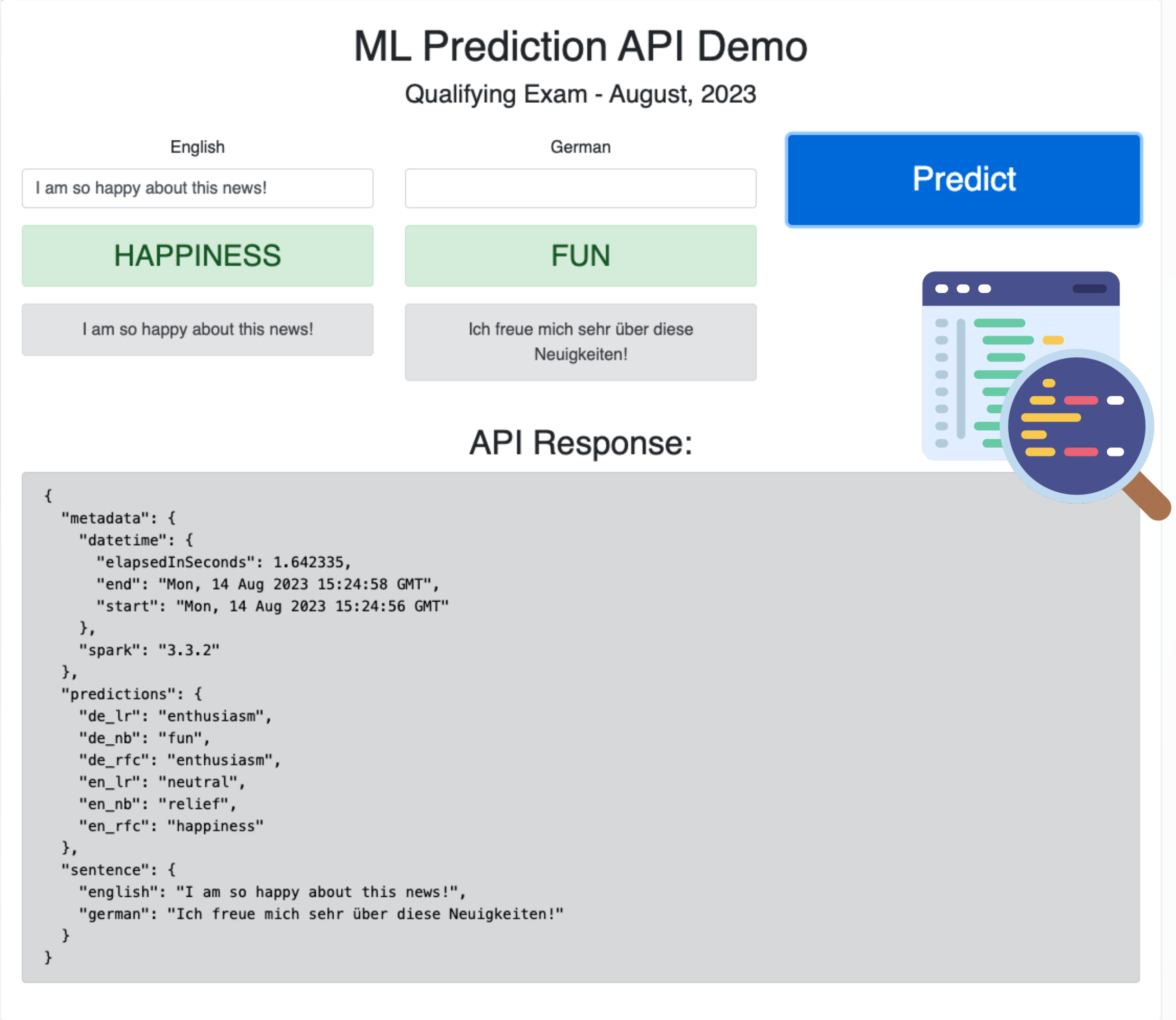
Results & Analysis

Real-Time API Translation and Prediction

Considering the disappointing outcomes of data extension via translation, our approach of utilizing real-time translation through the creation of an API and translating text in real-time for both German and English appears to be unsatisfactory in terms of achieving any substantial improvement as well.

Webpage API Test

	English		German		
Test	Emotion	Sentence	Emotion	Sentence	Time
A	Worry	This is bad idea and we need to stop right now	Anger	Dies ist eine schlechte Idee und wir müssen jetzt aufhören	1.49s
B	Happiness	What a wonderful world this is!	Relief	Was fur eine wundervolle Welt das ist!	1.65s
C	Happiness	I am so happy about this news!	Fun	Ich freue mich sehr uber diese Nauigkeiten!	1.64s
D	Sadness	Gosh I hate doing this today.	Relief	Meine Güte, ich hasse es heute.	2.56s



Conclusions & Future Work

Concluding

No significant improvements through translation by extending the dataset improved prediction results. We believe one of the main culprits is that the data for learning based on eight classes was insufficient.

$$\frac{Features}{Labels} = \frac{Sentences}{Classes} = \frac{1,500}{8} = 185$$
$$or = \frac{2,700}{8} = 337$$

The models seemingly resorted to memorizing[10] the sentence rather than learning the intricacies of the labeled classes.

Areas of Future Work

- Firstly, conducting further tests involving the translation of different models using diverse translation services. While Google Cloud's translation service sufficed for this research, superior paid subscription-based translation services are available.
- Secondly, to overcome the lack of data we could use AI to generated sentences that are similar yet distinct from the original. This could be achieved by providing a sentence and its associated labeled class for data creation. This approach aims to produce additional sentences imbued with similar emotional attributes, potentially expanding the labeled dataset.

Thank you

Citations

1. Andry Chowanda et al. “Exploring Text-based Emotions Recognition Machine Learning Techniques on Social Media Conversation”. In: Procedia Computer Science 179 (2021). 5th International Conference on Computer Science and Computational Intelligence 2020, pp. 821–828. issn: 1877-0509. doi: https : / / doi . org / 10 . 1016 / j . procs . 2021 . 01 . 099. url: https://www.sciencedirect.com/science/article/pii/S1877050921001320.

2. *Jay Stanley. ”THE DAWN OF ROBOT SURVEILLANCE”. 2019. url: https://www.aclu. org/report/dawn-robot-surveillance*

3. Sajani Ranasinghe et al. “An Artificial Intelligence Framework for the Detection of Emotion Transitions in Telehealth Services”. In: July 2022, pp. 1–5. doi: 10.1109/HSI55341.2022.9869503.

4. Laura Mascarell et al. “Stance Detection in German News Articles”. In: Proceedings of the Fourth Workshop on Fact Extraction and VERification (FEVER). Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 66–77. doi: 10.18653/v1/2021.fever-1.8. url: https://aclanthology.org/2021.fever-1.8.

5. Valentina Colonnello, Katia Mattarozzi, and Paolo M Russo. “Emotion recognition in medical students: effects of facial appearance and care schema activation”. In: Medical Education 53.2 (2019), pp. 195–205. doi: https://doi.org/10.1111/medu.13760. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/medu.13760. url: https://onlinelibrary.wiley.com/doi/abs/10.1111/medu.13760.

6. Shalini Kapoor and Tarun Kumar. “Detecting emotion change instant in speech signal using spectral patterns in pitch coherent single frequency filtering spectrogram”. In: Expert Systems with Applications 232 (2023), p. 120882. issn: 0957-4174. doi: https://doi.org/10.1016/j.eswa.2023.120882. url: https://www.sciencedirect.com/science/article/pii/S0957417423013842.

7. Sheetal Kusal et al. “AI Based Emotion Detection for Textual Big Data: Techniques and Contribution”. In: Big Data and Cognitive Computing 5.3 (2021). issn: 2504-2289. doi: 10.3390/bdcc5030043 . url: https://www.mdpi.com/2504-2289/5/3/43.

8. Kaggle. Url: https://www.kaggle.com/datasets/pashupatigupta/emotion-detection-from-text

9. Victor Sanh et al. “DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter”. In: ArXiv abs/1910.01108 (2019).

10. Xue Ying. “An Overview of Overfitting and its Solutions”. In: Journal of Physics: Conference Series 1168 (Feb. 2019), p. 022022. doi: 10.1088/1742-6596/1168/2/022022.