POZNAN UNIVERSITY OF TECHNOLOGY

FACULTY OF COMPUTING AND TELECOMMUNICATION
Institute of Computing Science

Master's thesis

EMOTION DETECTION CHALLENGES IN ENGLISH TO POLISH PARALLEL CORPUS

inż. Igor Czudy, 145198

Supervisor
dr inż. Dawid Wiśniewski

POZNAŃ 2024

Tutaj będzie karta pracy dyplomowej;
oryginał wstawiamy do wersji dla archiwum PP, w pozostałych kopiach wstawiamy ksero.

Contents

Abstract

English

The present study attempts to create a new Polish-language dataset used to predict a wide range of emotions. Automatic translation tools were used to translate the already existing qualitative English data set with is *GoEmotions: A Dataset of Fine-Grained Emotions* [7]. This work seeks to determine the quality of the newly created data and provides baseline models for the prediction of 27 emotions. Furthermore, three distinct automatic translation tools were evaluated in this study. The tool that demonstrated the most optimal performance, *DeepL*, was subsequently employed.

Polish

W niniejszej pracy podjęto próbę stworzenia nowego polskojęzycznego zbioru danych wykorzystywanego do przewidywania szerokiego zakresu emocji. Narzędzia do automatycznego tłumaczenia zostały wykorzystane do przetłumaczenia już istniejącego jakościowego zbioru danych w języku angielskim, którym jest *GoEmotions: A Dataset of Fine-Grained Emotions* [7]. Niniejsza praca ma na celu określenie jakości nowo utworzonych danych i zapewnia bazowe modele do przewidywania 27 emocji. Ponadto w badaniu oceniono trzy różne narzędzia do automatycznego tłumaczenia. Następnie użyto narzędzia, które wykazało najlepszą jakość tłumaczeń, którym jest *DeepL*.

Chapter 1

Introduction

The ability to detect emotions is a crucial aspect of human-computer communication. Feelings are key in human communication, influencing our decisions, relationships, and overall well-being. Understanding and automatically classifying them can significantly contribute to developing many applications. Examples of this are recommendation systems, intelligent user support systems, customer service bots that respond empathetically to users' feelings, or social robots that interact more naturally with humans.

In recent years, there have been significant advances in emotion classification methods through the use of advanced machine learning techniques, and much bigger datasets [4]. In Polish, however, the progress has not been as great. As Polish is not widely spoken, there is a significant problem with the lack of data in this language. It is believed that nowadays a key role in improving the performance of deep learning models is not the model itself, but the size and quality of the dataset [8]. It is therefore of the utmost importance to develop large and high-quality Polish language datasets for the continued development of natural language processing in Poland. What is more, in the problem of emotion classification, most of the datasets contain only a few classes defined. However, being more specific about emotions and extending the number of them could have a beneficial effect. The extension could facilitate a more comprehensive understanding of customers, as well as enhance the efficiency of systems that rely on human-computer communication.

Therefore, this work aims to create a Polish dataset that contains a wide range of emotions. This is achieved by translating with *DeepL* translator [5] a well-known, and one of the biggest English datasets, described in paper *GoEmotions: A Dataset of Fine-Grained Emotions* [7]. This work also aimed to describe and assess the quality of newly established data. Furthermore, robust baseline models have been developed based on this data. This has enabled a comparison of the results of models trained on Polish and English corpus.

The structure of the work is as follows. Chapter 2 provides an overview of the fundamental concepts used in this work, including neural network architectures, training methodologies, and measurements used. Chapter 3 introduces new data in depth, with analysis. Chapter 4 presents the experimental results from the models trained to classify emotions. Finally, chapter 5 presents conclusions, findings, potential limitations, and directions for future research.

Chapter 2

Theoretical background

2.1 Introduction

Supervised learning in the book *The Elements of Statistical Learning* by Hastie, Tibshirani, & Friedman [11] is defined as a task whereby a set of *independent variables* is attempted to predict *dependent variables*. The learning algorithm attempts to generate a function that optimally predicts dependent variables. Supervised learning can be divided into two groups: *regression problems* and *classification problems*. A regression occurs when the dependent variables are continuous. Conversely, if the attribute to be determined is a discrete variable, it is a classification problem. Commonly used algorithms used for classification problems are logistic regression, SVM, neural networks, and random forest. In the present work, the dependent variables are discrete, and thus the problem of emotion detection has been applied as a classification problem. More specifically, it is a **multi-label classification** problem [26]. This implies that each example may be categorized into multiple predefined classes, because more than one emotion may occur together for each comment.

2.2 Historical background

Emotion classification, or sentiment analysis, has been an active area of research within Natural Language Processing for several decades. The approach to this problem has evolved significantly over time, moving from rule-based systems to machine learning and, more recently, deep learning techniques.

In the early stages, emotion classification was primarily handled using **rule-based systems**. These systems relied on handcrafted rules and lexicons to identify and classify emotions [16]. Linguists or domain experts designed these rules, which often involve if-then statements. An example of such a statement is the following: *If a sentence contains the word "happy" then classify it as a joy.* The principal challenge was that creating comprehensive and accurate rules necessitates a profound understanding of language and is time-consuming. Commonly and well-known lexicons used for emotion classification or sentiment analysis are: *General Inquirer* [29], *LIWC (Linguistic Inquiry and Word Count)* [24] and *NRC Emotion Lexicon* [21].

In the early 2000s, researchers began utilizing **machine learning** techniques for sentiment analysis [23] [32]. These methods involved training classifiers such as Naive Bayes, Support Vector Machines (SVM), and Decision Trees [3]. The main difficulty in this concept was transforming text data into numerical features, known as feature extraction. At this times, the commonly used method for this was *Bag of Words (BoW)*, which represented text as a set of words and their frequencies, disregarding grammar and word order [13]. Furthermore, a common extension of the BoW method is N-grams, which provide some context by considering a sequence of tokens of a given length (e.g., pairs, named bigrams,

or triplets, named trigrams). The token represents meaningful elements in a text, e.g., a single word, number, or punctuation mark. [13]

In the 2010s, there was a notable improvement in the accuracy and robustness of sentiment analysis and emotion classification due to the utilization of **deep learning** methods. Using **Word Embedings** helped to resolve many problems that BoW representation of text has [13]. Word Embeddings are dense vectors that represent words. Their main advantage over Bag of Words is that they capture words' semantic meanings. Vectors that represent similar words are located in a similar position in vector space, which allows the model to understand relationships between words. Furthermore, BoW creates a very high-dimensional space, where each dimension corresponds to a unique word in the vocabulary. On the other hand, Word Embeddings can represent words in fewer dimensions, making the data more computationally efficient. In those years researchers started to use more complex neural networks, like **convolutional neural networks (CNNs)** and **recurrent neural networks (RNNs)**. Examples of papers using these models are: *Convolutional Neural Networks for Sentence Classification.* [14] and *Document Modeling with Gated Recurrent Neural Network for Sentiment Classification* [30].

2.3 State-of-the-art approach

The breakthrough in natural language processing and emotion classification came in 2017 with the release of the *Attention is all you need* paper [33]. In this work, a new deep learning model called **Transformer** has been proposed, which gives very good results in many NLP tasks such as translation, text generation, and classification problems. This model is a state-of-the-art approach in emotion classification problem until the day of writing this work [20]. Due to this fact in the following work this architecture was used.

The Transformer architecture relies on a mechanism called self-attention, which allows the model to weigh tokens against each other by considering their representations within a context window of a given size. It creates a new representation of each token based on the information from all the other tokens in this window. Unlike recurrent neural networks that process data sequentially, transformers process entire sentences simultaneously, enabling them to capture long-range dependencies more efficiently and faster. The architecture consists of an encoder and a decoder, each comprising self-attention, feed-forward layers, residual connections to bypass the attention or feedforward layers, and normalization layers to improve the learning process. The encoder reads and encodes the input text into a series of vectors, while the decoder generates the output text by interpreting these vectors and the text generated so far. Key innovations like positional encoding, which provides the model with information about the order of words, and multi-head attention, which allows the model to focus on different parts of the sentence simultaneously, make transformers highly effective and versatile for a wide range of language-related tasks, including emotion classification in text.

2.4 BERT

To be more precise, in this work, the BERT (Bidirectional Encoder Representations from Transformers) [8] model was employed, constituting only the transformer's encoder component. The objective of training a BERT model has changed in comparison to the original transformer. BERT was trained on two objectives, which are the Masked Language Model (MLM) and next-sentence prediction. MLM consists of predicting randomly masked words in a sentence by using the surrounding context, which enables the model to learn **bidirectional** representations of text. Unlike previous approaches, e.g. recurrent networks, which created representations based only on the left or right context, BERT sees both contexts

at the same time, which results in much better quality embeddings. On the other hand, next-sentence prediction predicts if true is that the sentence follows a specific different sentence.

Since the original BERT was trained on an English dataset, other models had to be used in this work. A solid baseline has been provided by fine-tuning the Polish HerBERT model [22]. This model uses the same idea as the original BERT, but it was trained for Polish language understanding. To fine-tune for emotion classification problem, one fully connected classification layer, with x neurons, was added on top of the model. It maps the embeddings from the [CLS] token (representing the whole input text) to a given number of neurons, each representing a different emotion. Each output neuron's value represents the probability of a given emotion.

Analogously, the mDeBERTa V3 model (multilingual Disentangled attention Bidirectional Encoder Representations from Transformers) was used in this work [12]. The main difference between this model and HerBERT is that mDeBERTa is multilingual and it has been trained on texts in multiple languages simultaneously. The model is capable of processing texts in different languages including Polish and English. In this work, the model was used separately for the English and Polish datasets and the combined bilingual dataset.

Moreover, the architecture of the mDeBERTa model has also been changed compared to the BERT. mDeBERTa uses *Disentangled Attention Mechanism*, which instead of treating words as a single entity, separates content and position information. This allows the model to differentiate between the syntactic roles of words and their semantic meanings. In the mDeBERTa model, the purpose of training was also changed. Instead of the *Next Sentence Prediction (NSP)* known from the BERT model, *replaced token detection (RTD)* was used. RTD introduces a binary classification problem: the model must predict whether a given token in the input sequence has been replaced with a random token or is the original token. This procedure reduces complexity and improves generalization.

2.5 Hyperparameters

Hyperparameters play a crucial role in the performance of machine learning models, including transformer models. In transformer models, hyperparameters are configurations set before the training process and include elements such as the **number of epochs, batch size, learning rate** and **weight decay**. The number of epochs is the number of transitions through the entire dataset when learning the model. The batch size determines the number of training samples processed before the model's internal parameters are updated, impacting the training process's speed and stability. A smaller batch size can provide more frequent updates but may introduce noise, while a larger batch size leads to more stable updates but requires more memory. The learning rate determines the step size during gradient descent, influencing how quickly or slowly the model converges. Weight decay, also known as L2 regularization, helps prevent overfitting by adding a penalty proportional to the squared magnitude of the model parameters to the loss function. This discourages the model from becoming too complex and helps it generalize to unseen data. Balancing these hyperparameters is essential for achieving optimal performance and ensuring the model's robustness and generalization ability.

2.6 Loss function

As stated in the introductory section of this chapter, supervised learning aims to identify a function that provides the most accurate prediction of the dependent variable. In this task the concept of the **loss function** is relevant. The loss function quantifies the difference between the predicted output of a model and the actual target values, guiding the optimization process to improve model performance [3].

For multi-label classification problems **Sigmoid Cross Entropy Loss Function** is a widely used loss function. This function consists of a sigmoid followed by a Binary Cross Entropy (BCE) loss function [25]. Sigmoid maps the input logits (which can be any real numbers) to output values between 0 and 1, which can be interpreted as probabilities of belonging to a particular class. Equation 2.1 shows the formula of the Sigmoid function.

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{2.1}$$

Then Cross Entropy equation, shown in 2.2, is applied. It calculates the cross-entropy between the true labels and the predicted probabilities, effectively penalizing predictions that are far from the actual labels. The loss is averaged over all instances in the dataset. By minimizing this loss, the model's predicted probabilities become closer to the true labels, enhancing its accuracy in distinguishing between classes.

$$\text{BCE}_{\text{multi-label}}(\mathbf{y}, \hat{\mathbf{y}}) = -\frac{1}{K} \sum_{k=1}^{K} (y_k \log(\hat{y}_k) + (1 - y_k) \log(1 - \hat{y}_k)) \tag{2.2}$$

where:

K - the number of classes,

y - a variable indicating whether a class is involved (0 or 1),

$\hat{y}$ - probability of this class

2.7 Metrics

A variety of techniques are employed to validate the quality of the outputs. These are typically more adequate than the previously described loss functions employed during the learning process. The purpose of this section is to describe the validation metrics used in this work.

2.7.1 Accuracy

Accuracy is the simplest and the most intuitive metric used for model evaluation. As equation 2.3 shows, accuracy is calculated as the ratio of the number of correctly predicted labels to the total number of labels [35]. Nevertheless, these metrics do not provide a comprehensive account. It is not known what class model has a problem with. What is more, accuracy should not be used for unbalanced data, due to its tendency to favor the majority class and mask the poor performance of minority classes.

$$\text{Accuracy} = \frac{\text{Number of correctly predicted instances}}{\text{Total number of instances}} \times 100\% \tag{2.3}$$

2.7.2 Confusion Matrix

Significantly more information than the accuracy can be shown by the confusion matrix. In multilabel classification, each instance can belong to one or more classes simultaneously. The evaluation is based on the principle of *one vs rest*, which means that for each class, one confusion matrix is counted. In contrast to the situation in binary classification, in this case, it is not possible to ascertain with which other classes a particular class is confused. However, it is possible to determine whether a class is predicted too often or too rarely. Every generated confusion matrix consists of four following matrices [28].

- True Positives (TP): Correctly predicting that text does express a certain emotion

- True Negatives (TN): Correctly predicting that text does not express a certain emotion
- False Positives (FP): Incorrectly predicting that text does express a certain emotion
- False Negatives (FN): Incorrectly predicting that text does not express a certain emotion

2.7.3 Precision, recall, f1-score

The measures listed above are the starting point for calculating **precision, recall**, and **f1-score** [36]. In the context of multi-label classification, all of these metrics have three distinct aggregation methods. They are **micro-average**, **macro-average** and **weighted-avarge**.

Precision, intuitively is the ability of the classifier not to label as positive a sample that is negative. The equation for Precision is provided in Formula 2.4.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{2.4}$$

Micro-average precision is calculated by considering each true positive and false positive across all classes. The formula 2.5 shows the equation for it.

$$\text{Micro-average Precision} = \frac{\sum \text{True Positives}}{\sum(\text{True Positives} + \text{False Positives})} \tag{2.5}$$

Macro-average precision calculates the precision for each class individually and then takes the average of these precisions. Each class is treated equally regardless of its size. The formula 2.6 shows the equation for it.

$$\text{Macro-average Precision} = \frac{1}{K}\sum_{i=1}^{K}\text{Precision}_i \tag{2.6}$$

Weighted-average precision calculates the precision for each class individually like macro-average but then takes a weighted average of these precisions, where the weights are the number of true instances for each class. The formula 2.7 shows the equation for it.

$$\text{Weighted-average Precision} = \frac{\sum_{i=1}^{K}(\text{Precision}_i \cdot \text{Support}_i)}{\sum_{i=1}^{K}\text{Support}_i} \tag{2.7}$$

Each of these metrics provides a different perspective and is valuable for different scenarios. Weighted-average is useful for imbalanced datasets because it gives more importance to frequent classes. On the other hand, Macro-Average treats all classes equally. Micro-avarge is good for the general view of performance.

Recall is the ability of the classifier to find all the positive samples. Recall has the same aggregation methods as was described for precision. What is more, all of them are analogous to them. Due to that, in order not to duplicate information, it was decided not to describe this aggregation method once again. The formula 2.8 shows all three equations.

$$\begin{aligned}
\text{Micro-average Recall} &= \frac{\sum \text{True Positives}}{\sum(\text{True Positives} + \text{False Negatives})} \\
\text{Macro-average Recall} &= \frac{1}{K}\sum_{i=1}^{K}\text{Recall}_i \\
\text{Weighted-average Recall} &= \frac{\sum_{i=1}^{K}(\text{Recall}_i \cdot \text{Support}_i)}{\sum_{i=1}^{K}\text{Support}_i}
\end{aligned} \tag{2.8}$$

The **F1-score** is a metric that combines precision and recall. To be more specific, the F1-score is a weighted harmonic mean of precision and recall. In this work, this metric is of crucial importance and is the main measure evaluating the final result of the classifier. Similarly to the matrices previously described, this matrix also has three aggregation methods. The formula 2.9 shows equations for every certain aggregation method, without redundant explanation.

$$\begin{aligned}
\text{Micro-average F1-Score} &= \frac{2 \cdot \text{Micro-average Precision} \cdot \text{Micro-average Recall}}{\text{Micro-average Precision} + \text{Micro-average Recall}} \\
\text{Macro-average F1-Score} &= \frac{1}{N} \sum_{i=1}^{N} \text{F1-Score}_i \\
\text{Weighted-average F1-Score} &= \frac{\sum_{i=1}^{N} (\text{F1-Score}_i \cdot \text{Support}_i)}{\sum_{i=1}^{N} \text{Support}_i}
\end{aligned} \tag{2.9}$$

2.7.4 Dataset split

The dataset was partitioned randomly into three subsets to facilitate model training, validation, and evaluation. The used division ratios are the same as in the paper *GoEmotions: A Dataset of Fine-Grained Emotions* [7]. Specifically, 80% of the dataset was allocated to the training set, 10% to the validation set, and 10% to the test set. This division ensures a balanced approach to model development, minimizing the risk of overfitting and providing an unbiased estimate of performance.

Chapter 3

Dataset

The dataset on which the experiments were conducted is a translated version of the original well-known and researched dataset: *GoEmotion* [7]. The original dataset was translated from English to Polish by DeepL Translator [1]. For purposes of this work, other translation methods have been tested, which have been further described in chapter 3.1, but the best results were obtained by DeepL.

The original dataset was created from Reddit comments from the years 2005 to 2019. The most popular comments were chosen and all non-English comments were removed. The assignees were native English speakers from India. For every example, at least three assignees labeled it independently. For example, if all three assignees disagree, two additional reviewers have been assigned. Assignees could choose more than one emotion from 27 emotions. If the raters were unsure about any emotion being expressed, they were instructed to select Neutral. This means that the dataset was labeled as a multi-label classification problem [3] with 27 classes + Neutral.

3.1 Translation

In recent years, there has been a significant advancement in the field of automatic translation. The advancements have been driven by innovation in deep learning techniques and greater computing power [33].

The study tested three popular APIs for translating the dataset from the English language to the Polish language. They are: *LibreTranslate* [17], *Google Translate* [9] and *DeepL* [5]. Exact information on the models used under these APIs is not easy to find, as they are companies' secrets.

However, the task of translating *GoEmotion* dataset is challenging. As mentioned before, this dataset consists of comments from the Reddit website. This can lead to problems. Translators may not be able to cope with slang, acronyms, and linguistic or typographical errors, frequently occurring on the site. What is more, comments in the dataset, are usually only a short response for other comments. This deprives them of a context that would facilitate translation.

The table 3.1 shows selected examples of translation. The first two translations are very similar to each other for all three APIs. However, *LibreTranslate* failed with a third translation. Probably, since word *DRINK* is written in capital letters, *LibreTranslate* did not translate this world. Similarly, the fourth example contains acronyms, which both *Google Translate* and *DeepL* have coped well, but *LibreTranslate* failed. These cases appear to be generalized to a whole dataset, not only these few, discussed examples. The author who is a Polish native speaker has drawn a conclusion that the quality of translation for *Google Translate* and *DeepL* are very similar, but *LibreTranslate* deviates from their quality.

Due to the above conclusions and that *DeepL* API is more accessible than *Google translate*, the author

Orginal	DeepL	Google Translator	libretranslate
I'm not even sure what it is, why do people hate it	Nie jestem nawet pewien, co to jest, dlaczego ludzie tego nienawidzą	Nie jestem nawet pewien, co to jest, dlaczego ludzie tego nienawidzą	Nawet nie wiem, co to jest. Dlaczego ludzie tego nienawidzą
OMG, yep!!! That is the final answer. Thank you so much!	OMG, tak!!! To jest ostateczna odpowiedź. Dziękuję bardzo!	O mój Boże, tak!!! To jest ostateczna odpowiedź. Bardzo dziękuję!	OMG, tak! To ostateczna odpowiedź. Bardzo dziękuję!
Yup, not anymore. Keep your blood sugar up! It really helps and DRINK water...	Tak, już nie. Utrzymuj poziom cukru we krwi na wysokim poziomie! To naprawdę pomaga i PICIE wody...	Tak, już nie. Utrzymuj wysoki poziom cukru we krwi! To naprawdę pomaga i PIJ wodę...	Już nie. Podnieście poziom cukru! To naprawdę pomaga i woda Drink...
>sexuality shouldn't be a grouping category It makes you different from othet ppl so imo it fits the definition of "grouping"	>Seksualność nie powinna być kategorią grupującą. Sprawia, że różnisz się od innych ludzi, więc imo pasuje do definicji „grupowania".	>seksualność nie powinna być kategorią grupującą. To odróżnia cię od innych ludzi, więc imo pasuje do definicji „grupowania"	> seksualność nie powinna być kategorią grupującą To sprawia, że różni się od othet ppl tak imo pasuje do definicji "grupowania"

Table 3.1: Selected examples of translation

decided to use *DeepL* model for the translation of *GoEmotion* dataset. The following subsection goes into detail of this translation.

3.1.1 DeepL translation

The *DeepL* website explains how their translation model works in a very superficial way [6]. It is demonstrated that they use stay-of-the-art Transformer architecture, with an attention mechanism [33]. However, *DeepL* company claims that they obtain very good translation results by changing a standard Transformer architecture. This key change is not publicly known, because it is the company's know-how. Public information on the trained dataset is also inferior. It is only known that a dataset comes from web crawlers.

DeepL translation results

The following results were obtained using the *DeepL*.

The four examples, shown in table 3.2, were translated to an empty string by *DeepL*. For this commends API returns value *Nan*. Due to this fact, the author of this work translated these phrases by himself. The result of this, are shown in the last column. Looking at these examples, it is possible to confirm the hypothesis that, due to their specificity, comments from Reddit are not easy to translate automatically. Repeated sequences of the same letters prevent DeepL from working correctly. In addition, *deepL* can not translate word *transvestite*. In the third sentence shown in the table 3.2, this word occurs in the whole context, but even when only a single word *transvestite* is attempted to be translated, the *DeepL* returns value *Nan*.

Orginal	DeepL	Author's translation
sssssssssssssshhhhhhhhhhhh just give this one to me it's been a weird day	Nan	sssssssssssssshhhhhhhhhhhh Daj mi to, to był dziwny dzień
WHY IS PULJUJARVI ON THE 4TH LINE?!?!?!?!?!'	Nan	DLACZEGO PULJUJARVI JEST W CZWARTEJ LINII?!?!?!?!?!
I'm scared to do these cuz the S.A.G.E. told me I was a transvestite	Nan	Boję się to zrobić, ponieważ S.A.G.E. powiedział mi, że jestem transwestyta
Oh fuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuu uuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuck RIGHT OFF!!!!!'	Nan	O kurwaaaaaaaaaaaaaaaaaaaaaaaaaaaa, PRAWIE OFF!!!!!

Table 3.2: Examples which were not translated by *DeepL*

Further translation inaccuracies also relate to the lack of translations. 1232 examples were not translated. It means that 1232 examples have the same string of characters before and after translation. These comments are only 0.5% of the whole dataset. The author has decided not to change or delete them in any way. Table 3.3 shows some of these comments. The following conclusions can be drawn:

Orginal	DeepL translation
y i k e s	y i k e s
oh, okay	oh, okay
#IDidNotVoteForThisClown!	#IDidNotVoteForThisClown!
#ERROR!	#ERROR!
wow fkn nice :)	wow fkn nice :)
Eat my fuck- [NAME]	Eat my fuck- [NAME]
Pew pew pew.. gotcha!	Pew pew pew.. gotcha!
Ay my brotha I'm peepin that squad car A$AP 👌	Ay my brotha I'm peepin that squad car A$AP 👌
LOL SO RANDOM	LOL SO RANDOM
BleachedAssholePink. 😂	BleachedAssholePink. 😂
Lol! 🕉🕉🕉	Lol! 🕉🕉🕉

TABLE 3.3: Examples of not-translated commends

		tokenized text	number of tokens
commend with hashtag:	#IDidNotVoteForThisClown!	#IDidNotVoteForThisClown!	10
linguistically corrected sentences made from this hashtag:	I did not vote for this clown!	I did not vote for this clown!	8

TABLE 3.4: Example of hashtag tokenization

DeepL has a problem with translating a slang. The word *yikes* is not commonly used in the official English language; what is more, here in the dataset it is written with space between every character, which makes a translation even more difficult.

The word *Ok* is commonly used in both English and Polish language. However, the translation of *okay* is *okej*, many native Polish people write it as it would be written in English. That can be a reason why *DeepL* did not translate *oh, okay* commend.

Hashtags are words or phrases preceded by a pound sign # used on social media platforms to categorize content and make it easily discoverable. They are very specific to the Internet language and many of them occur in this dataset. However, they are also not easy to translate automatically. The words in them are not separated by spaces but by capital letters. This can lead to problems with the tokenizer. The table 3.4 shows tokens for the hashtag and linguistically corrected sentences, made from this hashtag. Number of tokens for hashtag example is bigger than for corrected sentences. This can lead to unwanted token embeddings. For example, here instead of tokens *I* and token *did*, there are tokens *ID* and *id* that have a different meaning. The tokenizer used for the purpose of this example is *gpt-4o*[2] tokenizer, but this issue can be generalized to all modern tokenizers.

DeepL also has a problem with acronyms. In untranslated examples are many of them. Examples of them are: *fkn* with is an acronym of *fucking*, or *gotcha* with is *got you*. It is noteworthy that some of these terms are frequently employed in Polish informal language without any translation. Examples of them are shown in the next rows of the table 3.3. They are words: *LOL* or *ASAP*.

A significant proportion of the examples presented include emojis. Emojis are of considerable importance in the context of the classification of emotion, that why it is worth checking if they are translated correctly. For purposes of this work sufficient would be if any emoji code would be translated without any change into Polish. The cultures of Poland and England are so similar that all emojis have the same meaning in both languages. *DeepL* manages to translate a single emoji without any changes. In addition, it has not been observed that their occurrence in the context of a sentence has a negative impact on the translation. The reason why so many emojis appear in untranslated data is because in general, there are many of them in the dataset.

3.2 Data analysis

Description of the analysis of the original English dataset *GoEmotion* is presented in paper named: *GoEmotions: A Dataset of Fine-Grained Emotions*[7]. In order not to duplicate information this section only describes the original dataset shortly, and will focus on a description of translated data with a comparison to the original.

Both translated and original dataset contain 211 225 examples. The translated dataset, the same as the original one, has been randomly divided into training, validation, and testing sets in the following proportion: 80%-training, 10%-validation, 10%-test. The following is a list of possible 27 classes + Neutral.

- **admiration**
- **amusement**
- **anger**
- **annoyance**
- **approval**
- **caring**
- **confusion**
- **curiosity**
- **desire**
- **disappointment**
- **disapproval**
- **disgust**
- **embarrassment**
- **excitement**
- **fear**
- **gratitude**
- **grief**
- **joy**
- **love**
- **nervousness**
- **optimism**
- **pride**
- **realization**
- **relief**
- **remorse**
- **sadness**
- **surprise**
- **neutral**

Multiple labels may be assigned to each instance which makes this problem a **multi label classification** problem [3].

The dataset contains columns that are not names of emotions or text of comments and for this work, they were dropped. These columns are:

- **id** - Unique identifier of the comment
- **author** - The Reddit username of the comment's author
- **subreddit** - Subreddit that the comment belongs to
- **link_id** - The link id of the comment
- **parent_id** - The parent id of the comment
- **created_utc** - The timestamp of the comment
- **example_very_unclear** - Whether the annotator marked the example as being very unclear or difficult to label
- **rater_id** - The unique id of the annotator

Information about if the example was very unclear for the annotator is also relevant to this work. Such examples can also be difficult to classify for further models. However, all examples from the dataset are set as examples not very unclear.

3.2.1 Emotion distibution

Plot 3.1 shows the distribution of emotion in the whole dataset. This reveals a major problem of an unbalanced dataset. Further models might have a problem with the classification of minority classes as *grief*, *relief*, *pride*, *nervousness*, *embarrassment*, and *remorse*. Models might predict these classes very rarely. On the other hand majority classes as *neutral*, *approval*, *admiaration*, *annoyance*, *gratitude* or *disapproval*, might be predicted too often, as false positive.

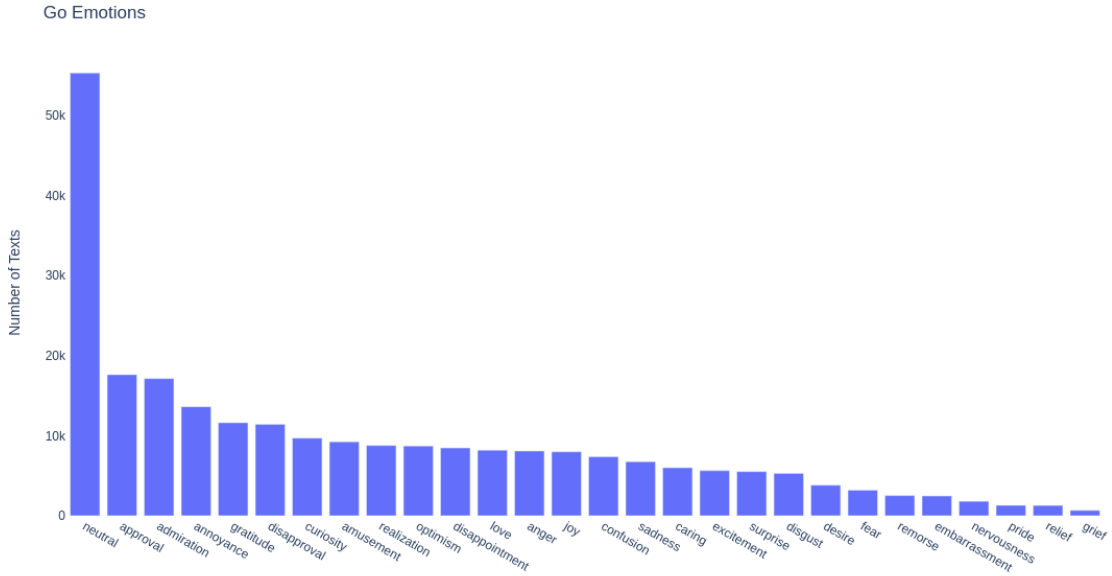


FIGURE 3.1: Emotion Distribution

3.2.2 Emotion correlation

Correlation between classes is relevant in multi-label classification problems. It helps to check with classes occur together. In particular, in the case of this work, which concerns as many as 27 classes, many emotions are very similar to each other. This is reflected in the plot 3.2. This plot shows *Pearson correlation*, which measures the linear relationship between two emotions. A high level of correlation can be remarked between *fear* and *nervousness*, *excitement* and *joy*, *anger* and *annoyance*, *sadness* and *disapointment*. All these pairs of emotions have very similar semantic meanings.

Neutral emotion draws attention. It has a very low correlation with the majority of other emotions. Due to the annotation rule, *Neutral* could be chosen only individually, without choosing other labels. This principle can be identified in the dataset, as all records labeled as *Neutral* do not exhibit any other emotional label.

3.2.3 Comments text analysis

Fig 3.3 visualizes the distribution of the length of texts measured by the number of characters in comments and the length of texts measured by the number of words in comments for both translated and original datasets. Outliers are not shown to improve readability. Both lengths had not changed much. Comments translated into Polish are slightly longer in the length of the number of characters than in English. Most of the comments have character lengths between 2 and 200. The longest and shortest comments were not translated and are the same for both datasets.

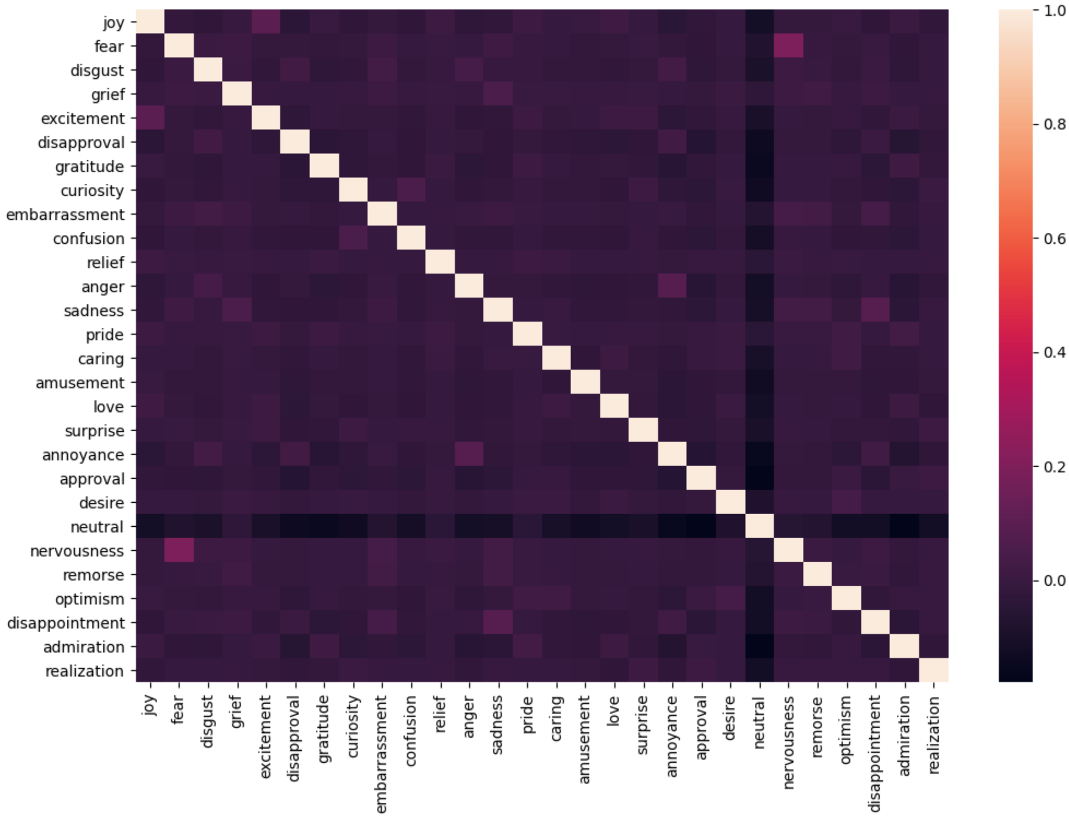


FIGURE 3.2: Correlation of emotions

Longest both Polish and English comment is: *This person is the smartest person to play town of salem literally 9999999999999999999999999999999999999999999999999999999999999999999999999999999999 9999910000000000000000000000000000000000000000000000000000000000000000000000000000000000000000 00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000 00000000000000000000000000000000000000000000000000000000000000000000000000000000000000012345 67898765432345676543345678987654345678909876543234567898765432345678909876543234567898765432 34567898765432345678765434567654345654345643454343434343432345676543456765432345432345654334 567898765432345678987656565656565656565656565656565454545654565454323456765432345678765456 IQ*

The shortest both Polish and English text is: *ok*

For word length, the situation is reversed. For Polish dataset lengths are shorter than for English.

However, since models used for emotion prediction in further work mostly rely on tokens, it is worth checking their length distribution. Plot 3.4 shows the token length distribution for both datasets. The tokenizer used for this plot, for the Polish dataset is *Herbert* Tokenizer [27], and for English *BERT* [8] Tokenizer. These tokenizers have the same architecture, and the only thing that makes them different is the dataset they were trained on. The length of Polish comments is somewhat longer than that of English. Due to these observations, in the future, during training, it might be worth extending a context length for *HerBERT* compared to *BERT*.

3.2.4 Repetitions

0.27% of comments are those that are repeated in the dataset. Table 3.5 shows the most repeated examples. They are mostly phrases commonly used in language, especially internet language. In addition,

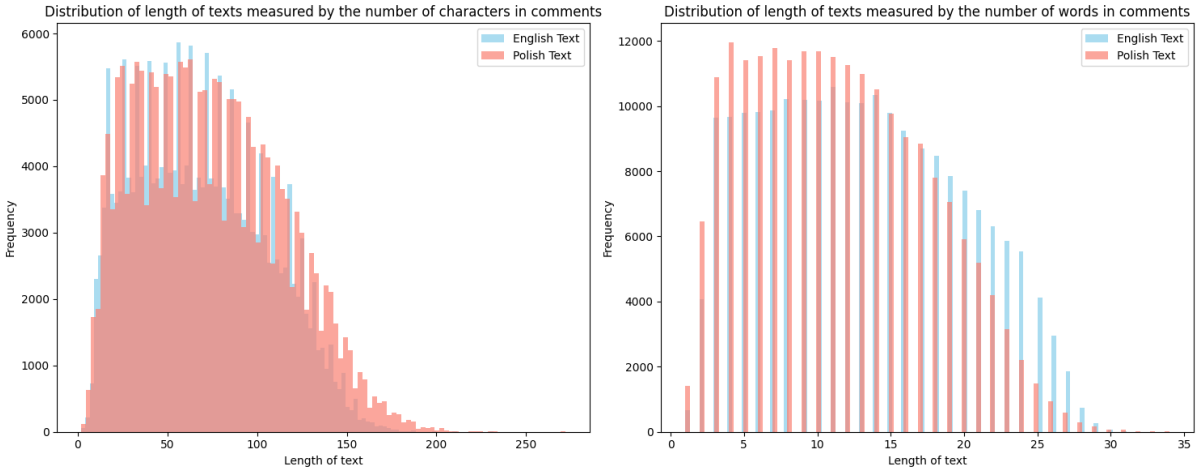


FIGURE 3.3: Distribution of length of texts measured by the number of words or characters in comments

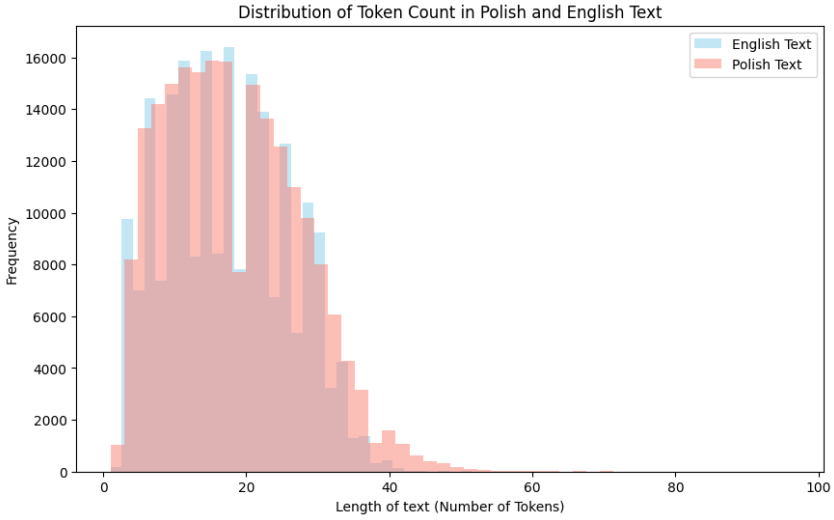


FIGURE 3.4: Token length of commends distribution

many phrases are repeated in the table, with the difference only in punctuation marks. For example the most frequently used word *Dziękuję (ang. Thank you)* appears in the text with a dot or exclamation mark.

Readers may be surprised that commend *Dziwny flex, ale ok* is often used. This phrase is not commonly used in the Polish language. However, the original text of this comment is *Weird flex, but ok*. It seems that the use of such a phrase in an informal internet conversation can be common in English.

Similarly comment *Szczęśliwego dnia ciasta!* with original meaning *Happy cake day !*. In neither Polish nor English-speaking countries *cake day* is a popular holiday. However, *Cake day* on the Reddit portal means account birthday, and it is common to give best wishes to other users on that day. This example illustrates the close relationship between our Polish dataset and the English language, as well as the informal language used on internet forums. This can lead to bias if we try to use a model trained on this dataset to classify emotion in very formal text.

The words *[NAZWA]* and *[NAZWA].* are also among the most repeated examples in the dataset. These tokens are the result of a translation process from the original dataset, where the token *[NAME]* was used to mask proper names referring to individuals. This masking was carried out using a BERT-based Named Entity Tagger [31] to ensure anonymity and privacy. Similarly, terms related to religion were masked using the *[RELIGION]* token. The frequent appearance of *[NAZWA]* and *[NAZWA].* in

the dataset is a direct consequence of this masking process, which replaces identifiable names and certain sensitive terms with standardized tokens.

Comment	Number of occurrences
Dziękuję.	56
Dziękuję!	47
Szczęśliwego dnia ciasta!	39
Dziwny flex, ale ok	33
Wszystkiego najlepszego!	31
Powodzenia!	28
Uwielbiam to	28
O tak	27
Uwielbiam to.	26
[NAZWA].	25
Nie ma za co	24
Podoba mi się	22
Szczęśliwego Nowego Roku!	22
[NAZWA]	22
To niesamowite!	22

Table 3.5: Most often commets in dataset

3.2.5 Keywords extraction

A significant amount of information about the created dataset can be derived through keyword extraction. This process is a technique that automatically extracts the most used and important words and expressions from a text. This chapter presents two keyword extraction methods for Polish and English datasets.

KeyBERT

KeyBERT[10] technique is based on *BERT* embeddings. Firstly a whole document, from which the objective is to extract the keywords, is represented by *BERT* embedding. Then every word or n-gram, from this text, is represented by *BERT* embeddings. Cosine similarity between these words and embeddings of the whole text is counted. The words with the highest similarity are chosen as keywords.

Table 3.6 shows Polish and English datasets keyword results for every emotion. A default model from KeyBERT: *all-MiniLM-L6-v2* was used for the English dataset, and for Polish, a multi-language model: *paraphrase-multilingual-MiniLM-L12-v2* was used.

The first observation is that the results are slightly different for languages. It could be expected that for the same, only translated datasets, keywords would be very similar and partially overlap. Here this is not the case. It appears that there are discrepancies between the translated and original datasets. The observed discrepancy may be caused by using two distinct models from KeyBERT.

In general, extracted keywords are not very related to emotion classes. A bit better results were obtained for English than for Polish. To illustrate, the words *anxiety, anxious, pray, and nervousness* are examples of words that have undergone senescence in the context of the class *nervousness*. However, for Polish, the results are many variations of the word *may (pl. móc)*, which are not related to emotion *nervousness*. A comparable situation can be observed in the case of the classes *fear or joy* when also keywords are much better for English than Polish. The best result for the Polish dataset was obtained for class *optimism* or *anger*, where specific words like *poszczęściło (eng. luck)* or *pieprzyłem (eng. fucked)* are meaningful.

Emotion	Language	Keyword_1	Keyword_2	Keyword_3	Keyword_4	Keyword_5
admiration	English	punjabi	suprise	intrigued	terrifyingly	hahahaha
	Polish	**wskazywałem**	**zespoły**	**wyjechał**	**zaciekawiło**	**zgłosi**
amusement	English	squeamish	hysterical	comical	sponge	craziest
	Polish	**zespołowo**	**żółty**	**zobaczyłem**	**złoty**	**zespoły**
anger	English	moderation	filth	abusive	abuse	abusing
	Polish	**złodziejem**	**pieprzyłem**	**złomiarza**	**zgłoszony**	**zatrzymałbym**
annoyance	English	defamation	coulter	slander	accusation	accused
	Polish	**zgłoszenia**	**zatłoczyli**	**zirytowało**	**zirytowałoby**	**zespoły**
approval	English	europhilic	tories	republicanism	brexit	ukpolitics
	Polish	**współdzielony**	**wykazało**	**zakończyłoby**	**wygłosić**	**wyłączanie**
caring	English	hug	breakup	heal	embrace	hugging
	Polish	**mogły**	**mogłoby**	**wysiłkom**	**wyszło**	**zdecydowały**
confusion	English	deadline	clemson	deadlines	headlines	scumbag
	Polish	**zwiększyłoby**	**mogłoby**	**zobaczyłem**	**zgłosić**	**zjebałem**
curiosity	English	runescape	supremacists	supremacist	4chan	militant
	Polish	**zakończyły**	**zdarzały**	**zdarzyło**	**zniedołężniały**	**zakończyło**
desire	English	4chan	ammo	survival	survive	scare
	Polish	**zapytałbym**	**zapytałem**	**zdarzyło**	**zniknęło**	**zamieniłbym**
disappointment	English	upset	despair	dumped	grief	cope
	Polish	**zgłoszenie**	**zakończyło**	**zdarzyło**	**zwiększyło**	**zdarzały**
disapproval	English	judge	wade	bail	lonzo	harden
	Polish	**nieokiełznana**	**zdarzały**	**zdarzyło**	**żółty**	**zgłoszony**
disgust	English	hypocrite	hypocrisy	homophobe	hating	hypocritical
	Polish	**zdarzyło**	**zwięzły**	**zapytałem**	**zdarzały**	**złoty**
embarrassment	English	dui	flashbacks	prank	reminded	forgetful
	Polish	**zapytał**	**zapytałem**	**zły**	**kłopoty**	**kłody**
excitement	English	stadium	winnipeg	hockey	montreal	nhl19
	Polish	**zdarzyło**	**zatrzymało**	**żółwia**	**zaryzykowała**	**zdarzało**
fear	English	horrific	horrifically	curse	cursed	vicious
	Polish	**mogłoby**	**mógłbym**	**współistniejącymi**	**mogłem**	**zażywałem**
gratitude	English	rejected	accepting	attempted	reconsidering	willing
	Polish	**zapytałem**	**zapytał**	**zgadzał**	**zdziczałe**	**zauważyło**
grief	English	murderous	grief	drowned	griefing	killer
	Polish	**zobaczyłem**	**przyjaciółmi**	**zgłoszenie**	**przytrafiło**	**przyszłości**
joy	English	joyous	joyful	thrilled	upset	delighted
	Polish	**żyło**	**złoty**	**zdawało**	**żółwia**	**zespoły**
love	English	nba	pistons	reddit	basketball	games
	Polish	**żałosnym**	**zespołu**	**głodu**	**ogłosili**	**zagłębiają**
nervousness	English	anxiety	anxious	pray	nervousness	meditating
	Polish	**mogłoby**	**przeczytałem**	**mógł**	**mogłeś**	**mogłem**
optimism	English	permit	enforce	laws	borrow	enforces
	Polish	**poszczęściło**	**przyszło**	**przyszłość**	**przeżyły**	**mogły**
pride	English	barrel	shoot	18	gun	jealous
	Polish	**przypomniało**	**przypominałoby**	**płci**	**przypomniał**	**przyjaciółkom**
realization	English	dealing	runescape	coping	overcome	pray
	Polish	**przyszłość**	**przyszło**	**zwiększyło**	**przyszłości**	**przyszedłem**
relief	English	margaritas	recovering	sober	relieving	unnerved
	Polish	**mogłoby**	**mogło**	**zeszły**	**przeszłości**	**siły**
remorse	English	apology	apologize	apologized	worries	apologise
	Polish	**przeczytał**	**przeczytałem**	**przyczyniło**	**przestał**	**przykuło**
sadness	English	upset	sadness	cries	saddened	heartbroken
	Polish	**zatruły**	**zdarzały**	**zapytałem**	**zdawałem**	**zaniepokoiły**
surprise	English	conspiracy	corrupt	paranoia	corruption	mysteriously
	Polish	**zgłosić**	**złożyć**	**zagłosować**	**zapytał**	**zatrzymał**

Table 3.6: Key words from keyBERT

TF-IDF

Another method used for keyword extraction is *TF-IDF (term frequency-inverse document frequency)* [18]. This method is much simpler than the previous one. For purposes of using it, for every emotion two documents were created. One contained all comments classified as a specific emotion, and another randomly chose all comments not classified as this particular emotion. It has been taken care to ensure that both documents are the same size. This method aims to find words that frequently exist in the first document and are rarely in the second document. For this purpose, two matrices are measured.

- The **term frequency** of a word in a document, which is the raw count of instances a word appears in a document. It is defined as:

$$\mathrm{TF}(t,d) = \frac{\text{Number of occurrences of the term } t \text{ in document } d}{\text{Total number of terms in the document } d}$$

- The **inverse document frequency** is defined as:

$$\mathrm{IDF}(t,D) = \log\left(\frac{N}{|\{d \in D : t \in d\}|}\right) \tag{3.1}$$

  where:

$$\begin{aligned} N &\quad \text{- is the total number of documents in the corpus} \\ |\{d \in D : t \in d\}| &\quad \text{- is the number of documents containing the term } t \end{aligned}$$

The multiplication of these two numbers yields the *TF-IDF score* of a word. The greater the score, the more pertinent the word is to the document.

$$\text{TF-IDF}(t,d,D) = \mathrm{TF}(t,d) \times \mathrm{IDF}(t,D) \tag{3.2}$$

Table 3.7 shows results from this method. The general conclusion that can be drawn from it is that keywords for the Polish and English languages are decent. In contrast to the previous method, the outcomes of this approach are more favorable for the Polish dataset. This demonstrates that, despite the Polish dataset being a non-original, automatically translated version, it is still high quality.

Examples of keywords that seem more matched for Polish than English are keywords for *love* emotion. For Polish all five words like: *ulubionymi (ang. favorite), zakochany (ang. in love), kochałem (ang. loved), zapach (ang. scent), kochany (ang. lovely)* are very much related to *love*. However, for the English dataset, some words like *rice* or *goat* are not related.

Examples when *TF-IDF* did not find the best words for both languages are for emotion *approval, disapproval* or *desire*. This may be because these emotions are more multidimensional and can relate to many different statements.

Classes of a high standard in terms of the quality of their keywords in both languages include: *embarrassment, admiration, confusion* or *fear*.

As mentioned in subsection 3.2.2, emotion *fear* and *nervousness* are correlated. Therefore, It is interesting to observe that one of the keywords is nearly repeated in both classes. Words *terrified* is a keyword for class *fear* and *terrifying* is a keyword for class *nervousness*. The main difference between these two keywords is that *terrifying* describes something that causes fear, while *terrified* describes someone who feels that fear. Similarly, for the correlated pair *sadness-disappointment*. We can observe a similar keyword: *boleśnie (eng. painfully)* for *sadness* class, and for the *disappointment* class, keyword: *boleć (eng. pain)*. The difference here is that one word is a verb, and another is an adverb.

In Polish, the keyword occurrence of the same word but with a different lemmatization form is common. For example for emotion *pride*, keywords are *dumna (eng. proud), dumni (eng. proud), dumą*

Emotion	Language	Keywords 1	Keywords 2	Keywords 3	Keywords 4	Keywords 5
admiration	English	lit	stadium	sweetest	generous	pup
	Polish	**pięknie**	**szanuję**	**niesamowici**	**fantastycznie**	**seksowne**
amusement	English	hahahaha	funnier	hahah	funniest	hah
	Polish	**rozśmieszyło**	**przezabawny**	**hahahaha**	**uśmiałem**	**zaśmiałem'**
anger	English	infuriating	motherfucking	nike	cunts	frustrated
	Polish	**śmiesz**	**suka**	**dranie**	**nike**	**draniu**
annoyance	English	irritating	ego	sloppy	spit	fools
	Polish	**irytujący**	**irytuje**	**irytująca**	**dupkami**	**dupie**
approval	English	oblivion	pockets	kicks	burned	agreeing
	Polish	**rozsądne**	**mogłyby**	**młodo**	**seksualny**	**98**
caring	English	worrying	cared	dealing	aim	visit
	Polish	**modlę**	**ostrożny**	**pomogę**	**błogosławię**	**jedz**
confusion	English	confusing	doubtful	unsure	clueless	code
	Polish	**mylące**	**rozumiałem**	**zdezorientowana**	**zrozumiem**	**idk**
curiosity	English	curiosity	familiar	study	intrigued	examples
	Polish	**ciekaw**	**ciekawi**	**sądzisz**	**dostałeś**	**twierdzisz**
desire	English	pizza	desperately	praying	desire	christmas
	Polish	**modlę**	**zagłosować**	**znów**	**zrobią**	**pozbyć**
disappointment	English	disappointment	disaster	upsetting	welp	election
	Polish	**rozczarowująca**	**rozczarowaniem**	**śmieciem**	**brakowało**	**boleć**
disapproval	English	criticism	unacceptable	ads	intentionally	seeking
	Polish	**nazwałbym**	**odmawiam**	**niedopuszczalne**	**firm**	**niepopularne**
disgust	English	shower	obnoxious	horrendous	website	disgust
	Polish	**obrzydliwy**	**paskudne**	**najgorszych**	**złą**	**niedorzeczne**
embarrassment	English	embarrassed	embarrassment	cringy	oops	ugh
	Polish	**żenujące**	**niezręczne**	**niezręczny**	**wstydzić**	**niezręcznie**
excitement	English	excitement	merrily	holiday	outstanding	ding
	Polish	**czekam**	**cakeday**	**urodzin**	**ekscytujące**	**podekscytowana'**
fear	English	scared	afraid	scary	terrified	scares
	Polish	**boję**	**obawiam**	**przerażający**	**przeraża**	**przerażony**
gratitude	English	clarification	thankfully	recommendation	clarifying	kindly'
	Polish	**dziękujemy**	**udostępnienie**	**opublikowanie**	**przyjrzę**	**rekomendację'**
grief	English	died	rip	loss	die	killed
	Polish	**twojej**	**straty**	**rip**	**zmarł**	**kondolencje**
joy	English	enjoyable	gladly	luckily	constructive	rams
	Polish	**szczęśliwa**	**radości**	**uszczęśliwia**	**szczęśliwszy**	**zadowoleni'**
love	English	cutest	rice	scratch	goat	scents
	Polish	**ulubionymi**	**zakochany**	**kochałem**	**zapach**	**kochany**
nervousness	English	nervous	worrying	depression	terrifying	trouble
	Polish	**niepokój**	**denerwuję**	**martwiłem**	**niespokojny**	**martwi**
optimism	English	hopeful	hopes	confident	optimism	handled
	Polish	**nadziei**	**miej**	**poprawi**	**pozostanie**	**poprawić**
pride	English	pride	congratulations	accomplishment	em	hero
	Polish	**dumna**	**dumni**	**wygrał**	**dumą**	**duma**
realization	English	realised	realization	messed	typed	forgotten
	Polish	**zdałem**	**uświadomienie**	**resztę**	**24**	**zdali**
relief	English	relieved	oof	words	thankfully	saved
	Polish	**bogu**	**ulga**	**jedyny**	**problemu**	**wreszcie**
remorse	English	apologies	guilty	guilt	fault	forgive
	Polish	**sorry**	**przepraszamy**	**winy**	**żal**	**radzić**
sadness	English	lonely	heartbreaking	heavy	tear	severe
	Polish	**bolesne**	**smutno**	**płaczu**	**smuci**	**boleśnie**
surprise	English	shocking	surprisingly	unbelievable	woah	surprises
	Polish	**niespodzianka**	**zdumiony**	**szok**	**zaskakująco**	**whoa**

Table 3.7: Keywords from TF-IDF

(eng. proud), duma (end. pride), and for annoyance: *irytujący (eng. irritating), irytuje (end. irritates), irytująca (end. irritating)*. This example shows problems of inflectional languages, where more keywords do not provide new information.

Chapter 4

Experiments

Strong baseline models were provided for emotion classification trained on a newly created Polish dataset. This chapter goes into detail on the training process and the final results of these models.

4.1 HerBERT model

The work entitled *GoEmotions: A Dataset of Fine-Grained Emotions* [7] provides a classification model with hypermeters and details of the classification results. The original English dataset is used in conjunction with this model. Nevertheless, the outcomes of this research may serve as a good starting point when evaluating a model trained on the Polish dataset.

Demszky gets the best result by fine-tuning the BERT-base model [8]. Most of the parameters used for training stay the same as in the paper *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding* [8], apart from batch size and learning rate. The final result, as presented in this paper, is an F1 macro-average score of 0.46.

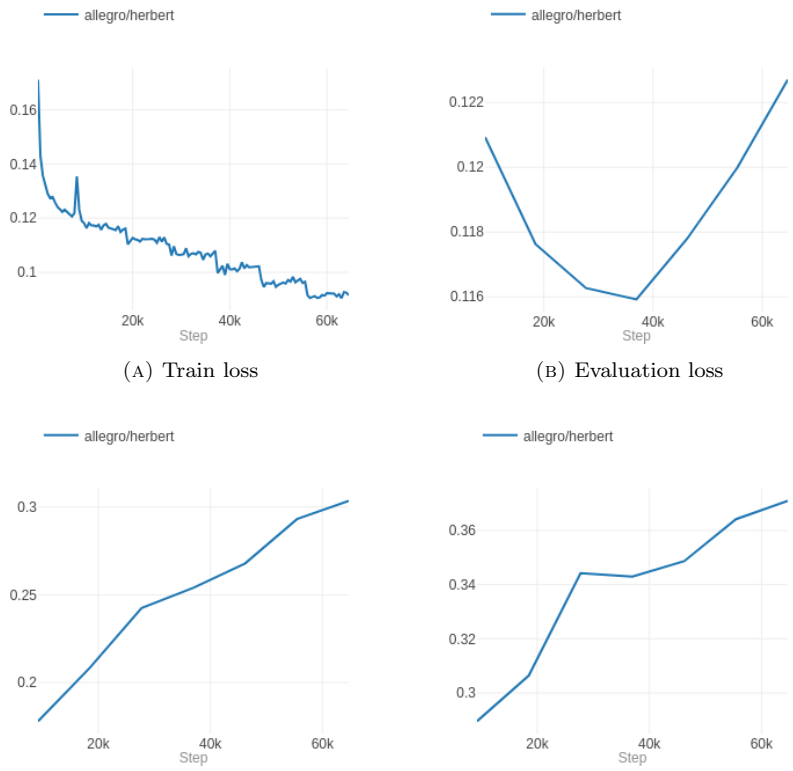
The Polish equivalent of the BERT-base model model is HerBERT-base-cased [22]. Due to this fact, the author of this work decided to research this model. The exact parameters used are shown in Table 4.1 and they are the same as parameters used in paper *GoEmotions: A Dataset of Fine-Grained Emotions*. This allows a better comparison between the two models and an assessment of the quality of the newly developed Polish dataset.

Model	**Number of Epochs**	**Batch Size**	**Learning Rate (lr)**	**Weight Decay (wd)**
herbert-base-cased-finetuned	7	16	5e-05	0.01

Table 4.1: Model Parameters

Figure 4.1 shows the process of learning of HerBERT model. Training loss decreases in every step of training. Validation loss converges to it until the fourth epoch. After four epochs, validation loss starts to increase which suggests overfitting. Nevertheless, as plots 4.1c and 4.1d show F1 scores still increase, providing better results in subsequent iterations of learning. This divergence is because the final probability of being assigned to classes changes, which affects the loss, but not necessarily the outcome of the F1 measure. This implies that the model's quality has improved, although its reliability has diminished. It is worth noting that the authors of the *GoEmotion* paper also noted a similar situation, where learning the model on an English dataset after the fifth epoch led to overfitting.

A final result is shown in table 4.2a. **F1 macro-avarge is 0.304**. Comparing it to F1 obtained on the English dataset in *GoEmotion* paper, the F1 macro-average on the Polish dataset is 0.16 lower. This drop in performance may be due to the fact of the inherent limitations of automated translation. As



(A) Train loss (B) Evaluation loss

(C) F1 macro (D) F1 micro

FIGURE 4.1: Training process of HerBERT

mentioned in Chapter 3, the quality of the data is diminished following the translation process, where some data were translated incorrectly, while others were not translated at all. Although automatic translation tools generally work very well, they still make mistakes. This is particularly evident in this work, where the dataset is difficult to translate because of its origin. As a reminder, the dataset consists of online comments with many linguistic errors, typos, acronyms, and slang. This may be the reason for the deterioration of the result compared to the result for the English collection.

A comparison of the F1 results of the individual emotions for both models shows some similarities between them. In general, for all the emotions with which the BERT model had a problem and obtained poor results, the HerBERT model also had a problem. It is worth noting that emotions with low f1 values are minority classes, which is most likely the reason for their poor performance. An exception to this is the least numerous class *Grief*, which received a better score for the Polish model than the English one. However, an f1 score of 0.18 is still a poor result. The emotion that has experienced the least deterioration in quality is *gratitude*, with a reduction of only 8% in the F1 score. This suggests that *gratitude* emotion is relatively easy to classify and even the translation process has not devalued its meaning. On the other hand, the *neutral* class result has deteriorated. This class is not a name of emotion and it implies the absence of any emotion. A reduction in the quality of this class may be indicative of a general decline in the quality of the data.

4.1.1 Prediction examples

Table 4.3 shows some examples from the test set of the HerBERT model prediction. The first two examples were predicted correctly. The first example intuitively appears easy to classify. There is only one label - *admiration* and the whole commant has a positive sentiment, matching this emotion. The second example is less straightforward to categorize. Certainly, the second part of the comment- *Ten dzieciak*

Emotion	F1 Score
Admiration	0.5576
Amusement	0.5717
Anger	0.3131
Annoyance	0.1020
Approval	0.1937
Caring	0.2725
Confusion	0.2294
Curiosity	0.3037
Desire	0.2742
Disappointment	0.1357
Disapproval	0.2080
Disgust	0.2009
Embarrassment	0.2133
Excitement	0.2142
Fear	0.4131
Gratitude	0.7948
Grief	0.1849
Joy	0.3085
love	0.605
nervousness	0.114
Neutral	0.4113
Optimism	0.3662
Pride	0.1993
Realization	0.1187
Relief	0.1026
Remorse	0.4499
Sadness	0.3246
Surprise	0.3136
macro-average	**0.3034**
micro-average	**0.3709**

(A) F1 Scores for Polish dataset for HerBERT model

Emotion	F1 Score
admiration	0.65
amusement	0.80
anger	0.47
annoyance	0.34
approval	0.36
caring	0.39
confusion	0.37
curiosity	0.54
desire	0.49
disappointment	0.28
disapproval	0.39
disgust	0.45
embarrassment	0.43
excitement	0.34
fear	0.60
gratitude	0.86
grief	0.00
joy	0.51
love	0.78
nervousness	0.35
neutral	0.68
optimism	0.51
pride	0.36
realization	0.21
relief	0.15
remorse	0.66
sadness	0.49
surprise	0.50
macro-average	0.46

(B) F1 Scores for English dataset for BERT model, source: *GoEmotions: A Dataset of Fine-Grained Emotions* [7]

TABLE 4.2: Comparison of F1 Scores

jest bohaterem (eng. This kid is a hero) is indicative of the presence of emotion *admiration*. However, the presence of words such as *kochać (eng. love)* and *nienawidzić (eng. hate)* in the first part of the sentence makes this example a little more difficult. These words are closely related to other emotions than labeled. A naive classifier could be distracted by the occurrences of these words. Nevertheless, the emotion in question has been correctly predicted. The last commend was mispredicted. Correctly labels are annoyance and optimism. The occurrence of the emotion *optimism* is indicated by the part of the sentence: *mam nadzieję, że nic im nie będzie... (eng. I hope they will be fine....)*. Nevertheless, the entire sentence does not have an optimistic sentiment. It is closer to the also labeled *annoyance* emotion. It is not unexpected that the model would make an erroneous prediction in this instance. The predicted emotion of *nervousness* is not a significant departure from the sentiment expressed in the sentence. This example illustrates the challenges inherent in the task of classifying as many as 27 discrete emotions.

Table 4.4 shows examples from outside the dataset. The first example is a quote from a well-known Polish threnody written by Jan Kochanowski [15]. Model classified the text as *sadness*, which is only partially correct. Certainly, this quote has a profound sense of sadness. Nevertheless, our model has been developed to predict more complex emotions. A more accurate representation of the text's sentiment

would be an expression of *grief*. However, as Table 4.2a shows, the model has difficulty with this emotion. The next example from the table is a quote from a popular Polish song [19]. The text was classified as *Love*, which is the correct label. This example is relatively straightforward. The text also contains the keyword *miłości (eng. love)*, and there are no words that present a challenge to the classification process.

Text	**Correct Labels**	**Predicted Labels**
Piękne miejsce, byłem tam we wrześniu!	admiration	['admiration']
Można go kochać lub nienawidzić. Ten dzieciak jest bohaterem	admiration	['admiration']
Naprawdę się martwię :(Wysłałem do nich DM, mam nadzieję, że nic im nie będzie...	annoyance, optimism	['nervousness']

Table 4.3: Example of HerBERT model prediction for examples from test set

Text	**Predicted Labels**
Wielkieś mi uczyniła pustki w domu moim, Moja droga Orszulo, tym zniknienim swoim. Pełno nas, a jakoby nikogo nie było: Jedną maluczką duszą tak wiele ubyło.	Sadness
Przez twe oczy, te oczy zielone oszalałem Gwiazdy chyba twym oczom oddały cały blask A ja serce miłości spragnione ci oddałem	Love

Table 4.4: Example of HerBERT model predictions for examples outside of the dataset

4.2 mDeBERTaV3 model

This study examines the multilingual mDeBERTaV3 model [12]. This model was trained on a multilingual dataset-*cc100* [34]. This dataset contains 12 GB of Polish data and 82 English data among other languages. Using a model trained on such a dataset enables a more precise and equitable comparison of the emotion classification outcomes for the English and recently created Polish datasets. Moreover, it also makes it possible to learn on both datasets simultaneously, where English and Polish are given alternately.

4.2.1 DeBERTaV3 models results

Table 4.5 shows parameters and evaluation metrics for mDeBERTaV3 model. The results are significantly worse than those presented in section 4.1. In the paper *GoEmotions: A Dataset of Fine-Grained Emotions*, the best model achieved a 0.46 F1-macro score, which is 0.17 superior to the current result. The performance of the Polish dataset is inferior when utilizing the mDeBERTaV3 model, yielding an F1-macro score of 0.253, a deterioration of 0.05, compared to the results presented in section 4.1. These significantly worse results when using the mDeBERTaV3 model may be due to poor selection of learning parameters. Due to computational limitations, the author of this paper could not elaborate further on this topic, leaving here room for further improvement.

Similarly as in section 4.1, there is an apparent lower score for the Polish-translated dataset than for the original English one. Nonetheless, using the same model for both the Polish and English corpus allows us to draw more confident conclusions than in the previous section. Choosing the same model for both corpuses removes the possibility of discrepancies in results caused by using two different models. However, it is important to note that the mDeBERTaV3 model was trained on a significantly larger corpus of English data than Polish data, which may contribute to the observed discrepancy. Further

reasons for this discrepancy are unchanged from the reasons outlined in section 4.1. These reasons are inherent limitations of automated translation and the high complexity of the Polish inflectional language.

Furthermore, it is of interest to examine the results of the combined Polish-English dataset. The final result of macro F1-score proved to be the most effective for the combined dataset than for the single English or Polish one. The improvement is 0.013 in comparison to the English language and 0.046 in comparison to the Polish language. This combination increased the dataset size by a factor of two. The expansion of the dataset may have had a considerable influence on the enhancement of the results. More data generally helps models to generalize better by capturing more diverse patterns. Also, the model trained on a bigger dataset is less likely to memorize noise or overfit to the training data. What is more, the combined dataset introduces a greater variety of linguistic patterns and vocabulary. In this case, such a linguistic variety was achieved by a certain type of trick. Finally, the newly added data is the same corpus, but in a different language. From the perspective of the multilingual model, this entails the processing of linguistic variety and improves the performance. It is important to note that the expansion of the dataset resulted in a significant increase in the time required for learning, which doubled in duration.

Language	**Duration**	**Nr. of Epochs**	**Batch Size**	**Learning Rate**	**Weight Decay**	**f1_macro**
English	5.2h	6	16	1×10^{-5}	0.01	0.286
Polish	5.2h	6	16	1×10^{-5}	0.01	0.253
Polish and English	10.5h	6	16	1×10^{-5}	0.01	0.299

Table 4.5: mDeBERTaV3 Models Training Parameters and Evaluation Metrics

DeBERTaV3 models results for individual classes

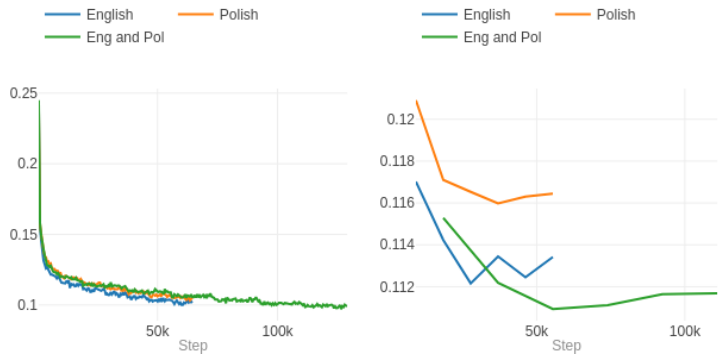
Table 4.6 shows the F1-score for every class for models trained only for the English or Polish dataset and for the combined dataset. In general, the least distinguishable classes are the minority classes such as Grief, Pride, Nervousness, and Embarrassment. Nevertheless, a model trained on a combined dataset appears to demonstrate superior performance in addressing these issues. Minority classes such as Relief, Pride, and Embarrassment have a significantly better F1-score score for a model trained on a connected dataset. This observation is another example confirming that more data helps models to generalize better. More specifically, as a consequence of the enlargement of the dataset, the number of minority classes also increased naturally. This approach enabled the resolution of the issue of imbalanced data, also improving the final result of the model.

DeBERTaV3 models learning process

Figure 4.2 shows the learning process of mDeBERTaV3 models. As a combined Polish and English dataset is bigger, without changing a number of epochs and batch size, the number of steps is also bigger compared with a single Polish or English dataset. That is the reason why the green plot is longer. What is more, figure 4.2a shows that a bigger dataset(combined Polish and English) needs more time to flatten out and achieve near minimum loss. Figure 4.2b presents evaluation losses. The model trained on a combined dataset (green line) learns much more stably. No major fluctuations are apparent, compared to the model trained on Enlis data (blue line). This agrees with the theory discussed in the paragraph above, where it is written that learning from larger data sets is less likely to be subject to learning noise and overfitting.

Metric	English model	English & Polish model	Polish model
F1 Score (Admiration)	0.572	0.578	0.548
F1 Score (Amusement)	0.627	0.614	0.563
F1 Score (Anger)	0.378	0.337	0.284
F1 Score (Annoyance)	0.022	0.051	0.003
F1 Score (Approval)	0.168	0.213	0.150
F1 Score (Caring)	0.301	0.278	0.254
F1 Score (Confusion)	0.245	0.240	0.218
F1 Score (Curiosity)	0.263	0.308	0.196
F1 Score (Desire)	0.259	0.273	0.241
F1 Score (Disappointment)	0.107	0.103	0.065
F1 Score (Disapproval)	0.185	0.190	0.131
F1 Score (Disgust)	0.215	0.212	0.156
F1 Score (Embarrassment)	0.183	0.219	0.148
F1 Score (Excitement)	0.217	0.214	0.141
F1 Score (Fear)	0.430	0.409	0.411
F1 Score (Gratitude)	0.816	0.818	0.787
F1 Score (Grief)	0.000	0.000	0.000
F1 Score (Joy)	0.358	0.303	0.274
F1 Score (Love)	0.658	0.651	0.628
F1 Score (Nervousness)	0.044	0.046	0.000
F1 Score (Neutral)	0.375	0.374	0.392
F1 Score (Optimism)	0.357	0.366	0.348
F1 Score (Pride)	0.011	0.251	0.000
F1 Score (Realization)	0.067	0.093	0.052
F1 Score (Relief)	0.000	0.055	0.000
F1 Score (Remorse)	0.445	0.445	0.406
F1 Score (Sadness)	0.352	0.369	0.358
F1 Score (Surprise)	0.358	0.363	0.321
F1 Macro	0.286	0.299	0.253
F1 Micro	0.376	0.375	0.348
ROC AUC	0.635	0.633	0.620

Table 4.6: Evaluation Metrics for mDeBERTaV3 Models



(a) Traning losees (b) Evaluation losses

Figure 4.2: Process of learning for mDeBERTaV3 models

Chapter 5

Conclusion

The present study attempts to create a new Polish-language dataset used to predict a wide range of emotions. Automatic translation tools were used to translate the already existing qualitative English data set. This work seeks to determine the quality of the newly created data and provides baseline models for the prediction of 27 emotions. Furthermore, three distinct automatic translation tools were evaluated in this study. The tool that demonstrated the most optimal performance, *DeepL*, was subsequently employed.

The generation of new datasets via this methodology has been demonstrated to be a viable approach; however, it is not without inherent limitations. It has been demonstrated that *DeepL* is unable to meet the required standards in some cases. This tool has problems with translating slang, acronyms, hashtags, and words with typos. For these reasons, the translated dataset certainly loses quality compared to the original dataset. This is also confirmed by results from trained models used to predict emotions. In general, the results of all experiments are more favorable for the English dataset than for the Polish one. One potential explanation for this discrepancy is the inferior quality of the translated data. Furthermore, the Polish language is an inflected language, which is considerably more complex than English. This undoubtedly contributes to the observed discrepancies in the results.

The main difficulty of the project was the limited computing resources. Training the models described in Chapter 4 requires a very large computing infrastructure, which was not possible to have in this project. For this reason, the work does not focus on the selection of appropriate parameters, but only provides a strong baseline model.

The obtained results leave much room for improvement. Researching other model parameters can lead to better results as well as interesting comparisons between different languages. In addition, it is worthwhile to engage in data preprocessing. As shown in chapter 3 the dataset has many recurring comments. Removing them can have a positive impact on results. Moreover, the use of methods for unbalanced data can lead to better results. The data used in this work are highly unbalanced and, as shown in chapter 4, this has a direct impact on the poorer performance of these classes.

Bibliography

[1] DeepL Translator. Library Catalog: www.deepl.com KerkoCite.ItemAlsoKnownAs: 2405685:Q7K3UGVE.

[2] GPT-4.0 tokenizer, 2024. OpenAI's GPT-4.0.

[3] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[4] Erik Cambria, Yang Li, Frank Xing, Soujanya Poria, and Kenneth Kwok. Senticnet 6: Ensemble application of symbolic and subsymbolic ai for sentiment analysis. pages 105–114, 10 2020.

[5] DeepL. Deepl language translation api libraries. `https://github.com/DeepLcom/deepl-node`, 2023. Accessed: 2024-05-30.

[6] DeepL. How does deepl work?, 2023. Accessed: 2024-05-30.

[7] Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. GoEmotions: A Dataset of Fine-Grained Emotions. In *58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020.

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[9] Google. Google translate. `https://translate.google.com/`, 2024. Accessed: 2024-05-30.

[10] Maarten Grootendorst. Keybert: Minimal keyword extraction with bert., 2020.

[11] T. Hastie, R. Tibshirani, and J.H. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer series in statistics. Springer, 2009.

[12] Pengcheng He, Jianfeng Gao, and Weizhu Chen. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing, 2021.

[13] Daniel Jurafsky and James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall PTR, USA, 1st edition, 2000.

[14] Yoon Kim. Convolutional neural networks for sentence classification. *CoRR*, abs/1408.5882, 2014.

[15] Jan Kochanowski. *Treny*. Wydawnictwo Ossolineum, Wrocław, 1997. Wydanie krytyczne z komentarzem, opracował Julian Krzyżanowski.

[16] Sophia Lee, Ying Chen, and Chu-Ren Huang. A text-driven rule-based system for emotion cause detection. pages 45–53, 06 2010.

[17] LibreTranslate. Libretranslate: Free and open source machine translation api. `https://github.com/LibreTranslate/LibreTranslate`, 2024. Accessed: 2024-05-30.

[18] Christopher D Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[19] Zenek Martyniuk. Przez twe oczy zielone, 2024. Piosenka.

[20] Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. Deep learning–based text classification: A comprehensive review. *ACM Comput. Surv.*, 54(3), apr 2021.

[21] Saif M Mohammad and Peter D Turney. Nrc emotion lexicon. *National Research Council, Canada*, 2:234, 2013.

[22] Robert Mroczkowski, Piotr Rybak, Alina Wróblewska, and Ireneusz Gawlik. HerBERT: Efficiently pretrained transformer-based language model for Polish. In *Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing*, pages 1–10, Kiyv, Ukraine, April 2021. Association for Computational Linguistics.

[23] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 79–86. Association for Computational Linguistics, July 2002.

[24] James W Pennebaker, Martha E Francis, and Roger J Booth. Linguistic inquiry and word count: Liwc 2001. *Mahway: Lawrence Erlbaum Associates*, 71(2001):2001, 2001.

[25] PyTorch. PyTorch Documentation. `https://pytorch.org/docs/stable/generated/torch.nn.BCEWithLogitsLoss.html`. Accessed: Insert Date Here.

[26] Jesse Read and Fernando Pérez-Cruz. Deep learning for multi-label classification. *CoRR*, abs/1502.05988, 2015.

[27] Piotr Rybak, Robert Mroczkowski, Janusz Tracz, and Ireneusz Gawlik. KLEJ: Comprehensive benchmark for polish language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1191–1201, Online, July 2020. Association for Computational Linguistics.

[28] scikit-learn developers. $sklearn.metrics.multilabel_confusion_matrix$, 2024. [$Online; accessed 28-June-2024$].

[29] Philip J Stone, Dexter C Dunphy, and Marshall S Smith. The general inquirer: A computer approach to content analysis. 1966.

[30] Duyu Tang, Bing Qin, and Ting Liu. Document modeling with gated recurrent neural network for sentiment classification. In Lluís Màrquez, Chris Callison-Burch, and Jian Su, editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1422–1432, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[31] Henry Tsai, Jason Riesa, Melvin Johnson, Naveen Arivazhagan, Xin Li, and Amelia Archer. Small and practical BERT models for sequence labeling. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3632–3636, Hong Kong, China, November 2019. Association for Computational Linguistics.

[32] Peter Turney. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In Pierre Isabelle, Eugene Charniak, and Dekang Lin, editors, *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 417–424, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics.

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[34] Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. CCNet: Extracting high quality monolingual datasets from web crawl data. In Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France, May 2020. European Language Resources Association.

[35] Wikipedia contributors. Accuracy and precision. `https://en.wikipedia.org/wiki/Accuracy_and_precision`, 2024. [Online; accessed 28-June-2024].

[36] Wikipedia contributors. Precision and recall. `https://en.wikipedia.org/wiki/Precision_and_recall`, 2024. [Online; accessed 28-June-2024].

Poznan University of Technology
Faculty of Computing and Telecommunication
Institute of Computing Science

Typeset using LaTeX.