Using Deep Learning for Emotion Analysis of 18th and 19th Century German Plays

Thomas Schmidt
Katrin Dennerlein
Christian Wolff

Fabrikation von Erkenntnis: Experimente in den Digital Humanities
Edited by Manuel Burghardt, Lisa Dieckmann, Timo Steyer, Peer Trilcke, Niels-Oliver Walkowski, Joëlle Weis, Ulrike Wuttke

Melusina Press 2021

Published in 2021 by **Melusina Press**
11, Porte des Sciences
L-4366 Esch-sur-Alzette
https://www.melusinapress.lu
Melusina Press is an initiative of the University of Luxembourg.

Concept: Niels-Oliver Walkowski, Johannes Pause
Copyediting: Carolyn Knaup, Niels-Oliver Walkowski
Cover: Valentin Henning, Erik Seitz

The digital version of this publication is freely available at https://www.melusina-press.lu.

ISBN (Online): 978-2-919815-25-8
DOI (Publication): 10.26298/melusina.8f8w-y749-udlf
DOI (Volume): 10.26298/melusina.8f8w-y749

Abstract

We present first results of the project "Emotions in Drama" in which we explore the annotation of emotions and the application of computational emotion analysis, predominantly deep learning-based methods, in the context of historical German plays of the time around 1800. We performed a pilot annotation study with five plays generating over 6,500 annotations for up to 13 sub-emotions structured in a hierarchical scheme. This emotion scheme includes common types like *joy*, *anger* or *hate* but also concepts that are specifically important for German literary criticism of this period like *friendship*, *compassion* or *Schadenfreude*. We evaluate the performance of various methods of emotion-based text sequence classification including lexicon-based methods, traditional machine learning, fastText as static word embedding, various transformer models based on *BERT*- or *ELECTRA*-architectures and pretrained with contemporary language, transformer-based methods pretrained or finetuned for historical and/or poetic language as well as the finetuning of BERT models via our own corpora and plays. We do achieve state-of-the-art results with hierarchical levels with two or three classes, i. e. the classification of valence (positive/negative). The best models are the transformer-based models *gbert-large* and *gelectra-large* by deepset pretrained on large corpora of contemporary German, which achieve accuracy values of up to 83%. Lexicon-based methods, traditional machine learning as well as static word embeddings are consistently outperformed by transformer-based models. Models trained on historical texts show small and inconsistent improvements. The performance becomes significantly smaller for settings with multiple sub-emotions like 6 or 13 due to the general challenge and class imbalances in which the models achieve 57% and 47% respectively. We discuss how we intend to continue our annotations and how to improve the prediction results via various optimization techniques in future work.

1. Introduction

Emotions are a central element in dramatic texts and serve the dramaturgy, character description, and the propagation of anthropological ideas. In the 18th century and the beginning of the 19th century, numerous important genres of German drama came into existence and some of the most important German playwrights worked in this timeframe. Thus, the role of emotions in plays of this period has been the subject of research of German Literary Studies for decades. Literary scholars explored these genres specifically concerning the intended emotional effect[1] and few scholars also investigated individual emotions in the context of character communication.[2] However, most of this research is focused on a limited set of canonical plays and the research methods are primarily qualitative and hermeneutical. Due to the large amount of dramatic texts written during this period, valid large-scale analysis is challenging. Therefore, little is

1 Pikulik 1965; Eibl 1971; Mog 1976; Sauder 1974-1980; Schings 1980; Nolting 1986; Wiegmann 1987.
2 Anz 2011; Schulz 1988; Mönch 1993, 344–350; Schonlau 2017.

known about large-scale research questions considering emotions, for example, how emotions are distributed in plays of that time in general, how they change with the plot or how they are linked to character attributes like gender and class. We present first results of the research project *Emotions in Drama,* [3] in which we explore computational emotion prediction methods on German plays around 1800 to support research based on these and similar research questions.

Computational sentiment and emotion analysis have gained a lot of interest in Digital Humanities (DH) and Computational Literary Studies (CLS) in recent years.[4] We use the term sentiment analysis to describe the task of predicting the expressed valence (positive/negative) in a text and emotion analysis to predict more varied and complex categories like anger, surprise, or happiness. Sentiment and emotion analysis is used in CLS and DH to explore genres like fairy tales[5], novels[6], fan fictions[7], lyrics[8], online forums[9], political texts[10], movie subtitles[11], and historical plays[12]. Methodologically, the application of lexicon-based sentiment and emotion analysis (see chapter 3.2.1) is dominant, although the method is regarded as outdated in Natural Language Processing (NLP) and has been shown to have difficulties dealing with the historical language.[13] Annotated corpora which are necessary for more modern machine learning (ML)-based approaches are rare and have to deal with the challenge of literature as a subjective and complex annotation object. Low annotation agreements among annotators and the difficulty of the annotation as such are a common subject concerning literary and historical texts.[14] Crowd-sourcing annotations is rare due to the lack of necessary expertise concerning the language and content of the annotated works. Furthermore, annotation schemes are mostly inspired by psychological concepts[15] that might diverge from the interests of literary scholars. In *Emotions in Drama*, we intend to develop annotation schemes that are focused on literary perspectives.

Computational emotion prediction is usually defined as a single- or multi-label text sequence classification task with multiple classes derived from categorial schemes.[16] Emotion prediction is still regarded as a challenging task even in contemporary text

3 The work presented here is supported by a grant of the *German Research Foundation* (*Deutsche Forschungsgemeinschaft*, DFG. It is part of DFG priority programme *Computational Literary Studies* (CLS; SPP 2207, for more information see: https://dfg-spp-cls.github.io).
4 Cf. Schmidt et al. 2018b.
5 Alm / Sproat 2005; Mohammad 2011.
6 Kakkonen / Kakkonen 2011; Mohammad et al. 2011; Samothrakis / Fasli 2015; Reagan et al. 2016; Jacobs 2019.
7 Kim / Klinger 2019; Pianzola et al. 2020.
8 Schmidt et al. 2020a.
9 Schmidt et al. 2020b; Moßburger et al. 2020.
10 Sprugnoli et al. 2016.
11 Öhman / Kajava 2018; Kajava et al. 2020; Schmidt et al. 2021c.
12 Mohammad 2011; Nalisnick / Baird 2013; Schmidt / Burghardt 2018a; Schmidt / Burghardt 2018b; Schmidt et al. 2018a; Schmidt 2019; Schmidt et al. 2019b; Schmidt et al. 2019c; Yavuz 2021.
13 Schmidt / Burghardt 2018a.
14 Alm / Sproat 2005; Sprugnoli et al. 2016; Schmidt et al. 2018a; Schmidt et al. 2019a; Schmidt et al. 2019b.
15 Cf. Wood et al. 2018a; Wood et al. 2018b.
16 Cf. Cao et al. 2020.

types but recent developments in deep neural networks and large word embeddings have led to significant improvements compared to lexicon-based approaches and traditional ML.[17] One of the most important developments are contextual word embeddings and large pretrained language models like *BERT*[18] or *ELECTRA*[19] that currently achieve the best performances in sentiment and emotion prediction evaluations in NLP and are considered state-of-the-art.[20] We argue that using semantic representations of language that do include context like these transformer-based models do, is a more promising approach for literary studies than previous methods since the overall interpretation of a text unit or a work is increasingly important. Furthermore, transformer-based architectures offer novel possibilities for domain adaptation by pretraining language models on texts of the target domain or further pretraining models trained on contemporary language.[21] Indeed, these approaches have shown success for NLP tasks on German historical and poetic text types.[22]

In the following paper, we report results of our project *Emotions in Drama* that are mostly work-in-progress. They include the following contributions to the research area of emotion analysis of historical German plays:

- We describe our current emotion annotation scheme and process that is more directed towards literary annotation than previously established schemes.
- We annotate a sub corpus of our target corpus using our annotation schemes.
- We use our annotations as gold standard for implementing and evaluating various types and models of the following methods for four different classification tasks ranging from binary valence classification to 13 sub-emotion classes.
  - lexicon-based sentiment analysis (baseline)
  - "traditional" machine learning methods via bag-of-words and Multinomial Naïve Bayes and support vector machines (SVM) (baseline)
  - A static word embedding (*fastText*[23]) with a subsequent neural network.
  - Various transformer-based language models pretrained on contemporary language.
  - Various transformer-based language models pretrained from scratch or further pretrained (fine-tuned) on historical and/or poetic language.
  - Various transformer-based language models further pretrained (fine-tuned) of our "main/target" corpus.

These contributions enable us to (1) develop and discuss fitting annotation schemes for the analysis of emotions in our setting, (2) explore in a first study how state-of-the-art

17 Shmueli / Ku 2019; Acheampong et al. 2020; Cao et al. 2020.
18 Devlin et al. 2018.
19 Clark et al. 2020.
20 Shmueli / Ku 2019; Munikar et al. 2019 ; Cao et al. 2020; Dang et al. 2020; Gonzáles-Carvajal et al. 2021; Cortiz 2021.
21 Sarma et al. 2018; Beltagy et al. 2019; Rietzler et al. 2020; Gururangan et al. 2020.
22 Labusch et al. 2019; Schweter / Baiter 2019; Brunner et al. 2020a; Schweter / März 2020.
23 Bojanowski et al. 2017.

approaches in computational emotion prediction perform against baselines, and (3) investigate how techniques of domain adaptation help to improve performance.

2. Annotation

In this chapter we detail the developed annotation scheme and process and present the results of the annotation of a representative set of plays that we use to implement and evaluate computational emotion prediction techniques. Please note that the scheme and process were developed in an iterative process of pilot annotations and is still work-in-progress anticipating further changes.

2.1 Annotation Scheme

We define emotions as characters' states of mind that are expressed, among other channels, through written language. Therefore, we annotate the intended emotions experienced by characters and/or attributed to them in the text taking context and interpretation into account. While most emotion annotation schemes in NLP are inspired from categorial systems in psychology,[24] we deviate from this in order to take into account the interest of literary scholars for this specific period. Our scheme consists of 13 *sub-emotions* as the lowest level that are classified by 6 higher-level categories that we refer to as *main emotion classes* (4 categories and 2 special cases). They can further be distinguished concerning their *binary valence* (positive/negative; marked as + and – in the following list). The set is as follows (we include the original German terms that we actually use for annotation in brackets):

- Emotions of affection (*Emotionen der Zuneigung*)
  - Desire (*Lust*) (-)
  - Love (*Liebe*) (+)
  - Friendship (*Freundschaft*) (+)
  - Adoration (*Verehrung*) (+)
- Emotions of joy (*Emotionen der Freude*)
  - Joy (*Freude*) (+)
  - Schadenfreude (+)
- Emotions of fear (*Emotionen der Furcht*)
  - Fear (*Angst*) (-)
  - Despair (*Verzweiflung*) (-)
- Emotions of suffering (*Emotionen des Leids*)
  - Suffering (*Leid*) (-)
  - Compassion (*Mitleid*) (-)

24 Wood et al. 2018a; Wood et al. 2018b.

    - Anger (*Ärger*) (-)
    - Hate (*Abscheu*) (-)
- Emotional movement (*Emotionale Bewegtheit*)

Emotional movement is a special type of annotation used to annotate unspecific emotional arousal as well as astonishment and has therefore no valence assignment. In conjunction with positive and negative, emotional movement forms the structural unit we will refer to as *triple valence*. Emotions of affection, joy, fear, suffering as well as the classes hate and emotional movement are referred to as *main emotion class* in the following. Annotators annotate speeches and stage directions of the plays. While the annotation of fixed structural units is recommended and common for NLP tasks, we experienced difficulties during the pilot annotations and decided for the annotation of varied text sequences which represent the emotional expressions of these texts the most. Therefore, annotators can annotate single words, parts of sentences, or multiple sentences.

2.2 Main Corpus and Annotated Sub-Corpus

We refer to the term *main corpus* to describe the plays we are planning to investigate in our project. For a representative sub-part of these plays, annotations are necessary for the ML-based emotion prediction techniques. Our main corpus currently consists of selected German canonical and non-canonical plays for the time period 1770–1815 from the platform *TextGrid*, [25] the corpus *GerDraCor* [26] and a selection of so called "Kasperl"-plays[27]. We plan to extend this corpus in the future. For our first annotation study we selected five representative plays of this corpus. We refer to this corpus in the following chapters as sub-corpus:

- *Minna von Barnhelm* (1767) by Lessing (comedy)
- *Kabale und Liebe* (1784) by Schiller (tragedy)
- *Kasperl' der Mandolettikrämer* (1789) by Eberl (comedy)
- *Menschenhass und Reue* (1790) by Kotzebue (comedy)
- *Faust. Ein Tragödie* (1807) by Goethe (tragedy)

2.3 Annotation Process

We follow annotation procedures similar to those of other projects[28] with the same kind of text types – annotations were performed by trained students and experts.[29] Plays were fully annotated from beginning to end since we perform context and content-

25 https://textgrid.de/digitale-bibliothek#search
26 Fischer et al. 2019.
27 For more information see: http://lithes.uni-graz.at/maezene/maezene_startseite.html
28 Alm / Sproat 2005; Sprugnoli et al. 2016; Schmidt et al. 2018a; Schmidt et al. 2019b; Brunner et al. 2020b.

aware annotations that need a deeper understanding of the plot of the play. Each play was annotated independently from each other by two student annotators, which are employed in the project and thus compensated monetarily for their work. The training of the annotators included multiple sessions with an expert annotator performing pilot annotations. Furthermore, they had access to an annotation guidelines document throughout the process. The annotations were performed with the tool *CATMA* [30] and each annotator had around 1-2 weeks to perform the annotations, which took in total around 8-12 hours to complete.

2.4 Annotation Results

So far, we have collected 6,596 annotations by two annotators per play. Table 1 illustrates the distribution of emotion annotations for all sub-emotions and higher classes.[31]

Table 1: Distribution of sub-emotions and main emotion class categories. The sub-emotions are listed followed by the summed results of the main categories in bold. Percentages are rounded.

EMOTION	ABSOLUTE	%	AVG. TOKENS	MIN TOKENS	MAX TOKENS	STD. TOKENS
Desire	50	1	23.22	4	83	16.49
Love	783	12	26.16	1	326	33.67
Friendship	127	2	22	1	120	18.66
Adoration	306	5	19.63	1	96	16.36
Emotions of affection	1,266	19	24.05	1	326	28.61
Joy	850	13	22.78	1	223	24.3
Schadenfreude	201	3	25.02	1	121	21.89
Emotions of joy	1,051	16	23.21	1	223	23.86
Fear	424	6	16.87	1	173	17.45

29 We want to thank the following student assistants who worked as annotators for their contributions to this project: Viola Hipler, Julia Jäger, Emma Ruß, and Leon Sautter.
30 Gius et al. 2020.
31 More information about the annotation results can also be found in the paper by Schmidt et al. 2021b.

Despair	282	4	30.78	1	206	30.15
Emotions of fear	706	11	22.42	1	206	24.32
Suffering	998	15	26.12	1	302	28.91
Compassion	318	5	21.61	1	156	21.87
Anger	880	13	22.14	1	261	24.35
Emotions of suffering	2,196	33	23.87	1	302	26.27
Hate	614	9	25.05	1	167	26.19
Emotional movement	763	12	24.4	1	313	32.74

We have identified a dominance of negative emotions (54%), which is in line with previous research concerning sentiment or emotion annotations of literary texts.[32] In accordance, the most frequent annotations are suffering (15%), joy (13%) and anger (13%). We have multiple categories which are rarely annotated like desire (1%) or friendship (2%). This and the class imbalance for most hierarchies poses problems for ML which will be discussed in the upcoming chapters. The average length of an annotation is around 25 tokens; however, annotation lengths vary widely from one-word-annotation to longer passages (table 1).

To investigate the validity of annotation concepts and limitations for computational approaches, agreement among annotators is an important metric. Due to the varied and possibly partially overlapping annotations, we decided to use the following heuristic to calculate traditional annotation metrics: We only regard the structural unit of speech and assign per annotator the emotion that is annotated the most for a single speech. If a speech shows no annotations, we mark it with an extra class, *non-annotation*. Applying this system, we calculated Cohen's κ (K in table 2) and percentage-wise agreement (% in table 2) among annotators per play and overall for all emotion hierarchies (table 2).

32 Alm / Sproat 2005; Sprugnoli et al. 2016; Schmidt et al. 2018a; Schmidt et al. 2019a; Schmidt et al. 2019b.

Table 2: Speech-based agreement statistics per play for the binary valence, the main emotion class and sub-emotions respectively.

DRAMA	VALENCE (K)	VALENCE (%)	CLASS (K)	CLASS (%)	EMOTION (K)	EMOTION (%)
Faust	0.44	67.853	0.345	59.399	0.342	58.064
Kabale und Liebe	0.382	58.908	0.325	50.313	0.312	47.992
Menschen-hass und Reue	0.402	75.28	0.347	72.331	0.347	71.91
Minna von Barnhelm	0.406	74.619	0.377	72.752	0.356	71.23
Kasperl' der Man-dolet-tikrämer	0.42	70.83	0.344	65.34	0.312	62.72
Overall	0.41	69.498	0.3476	64.027	0.333	62.383

The results indicate mostly moderate to fair agreements according to Landis and Koch.[33] While low compared to sentiment and emotion annotations of other text types,[34] this result is in line with previous research,[35] thus proving again the difficulties in annotation and the subjective nature of the material. For the ML algorithms of the computational emotion recognition we regard all annotations by the annotators as "gold standard". We motivate and further describe this decision in chapter 3.1.

3. Emotion Prediction

In the following chapter, we introduce the computational emotion prediction methods we have applied and evaluated. We regard the emotion prediction as single-label classifi-

33 Landis / Koch 1977.
34 Wood et al. 2018a; Wood et al. 2018b.
35 Alm / Sproat 2005; Sprugnoli et al. 2016; Schmidt et al. 2018a; Schmidt et al. 2019a; Schmidt et al. 2019b.

cation task on text sequences with varied lengths (as measured with the number of words). Depending on the emotion level, this results in various numbers of classes. For the binary valence, we predict two classes (positive vs. negative), for triple valence three classes (positive vs. negative vs. emotional movement), for the main emotion category six classes and the for the prediction of sub-emotions 13 classes (please refer to chapter 2 for the different classes). All approaches are implemented in *Python*.

3.1 Training and Evaluation Material

For the training of machine learning approaches as well as the general evaluation we use the entire annotated material of the annotation study described above. We use all text sequences that are annotated with emotions for the five plays by the two annotators. We do not resolve disagreements or adjust the text sequences. Due to the free and varied annotation process, it is difficult to identify and resolve disagreements in our setting, since clear word-by-word disagreements are rare while disagreements based on overlapping or in comparison to non-annotations are more frequent. While we do plan to investigate possibilities of dealing with this problem in future work (see chapter 5), we neglect it for our first studies to gain fast impressions of possible approaches. We also do not add unannotated material as sort of a neutral class to the training and evaluation corpus. Thus, this corpus consists of 6,596 annotated sequences of varied lengths. Please refer to table 1 for an overview of class and text size distributions. It is important to keep in mind that the missing resolving of disagreements and the general class imbalances pose challenges for every class prediction approach that must be kept in mind when interpreting the performance. Please note that if not stated otherwise in the following chapters, we used the annotations as they are without any preprocessing except the stripping of whitespace at the beginning and end of every annotation unit.

3.2 Approaches

We include the following methods in our experimental setting: Lexicon-based sentiment analysis (only for binary valence prediction) and traditional machine learning as baselines, one implementation of static word embeddings and a subsequent neural network, various pretrained transformer-based models, transformer-based models pretrained or further pretrained with historical or poetic language as well as models further pretrained with texts of our main corpus and the annotated sub-corpus.

3.2.1 Lexicon-based Sentiment Analysis

Lexicon-based sentiment analysis is a term used to describe rule-based methods that work via predefined lists of words that are annotated concerning the sentiment orientation with a numeric value representing the "a priori" valence of the specific word. By summing up the values for positive and negative words and deducting the sum of negative by the sum of positive one receives an overall value for the polarity or valence ex-

pression of a text unit.[36] In a similar way, one can perform emotion prediction with a fitting set of emotion-annotated lexicons. While these methods are regarded as outdated for sentiment and emotion classification, they are still popular in areas that lack annotated training corpora like CLS[37] also leading to the development of tools specifically for this task and community.[38] Some of the most popular sentiment and emotion lexicons in English are the *NRC Emotion Lexicon* [39] or *VADER*.[40] One of the most popular and largest German sentiment lexicons is *SentiWS*.[41] We implemented two lexicon-based sentiment classification approaches via *SentiWS*: (1) using *SentiWS* without any preprocessing in the way described above and (2) optimizing the usage of *SentiWS* via processes like lemmatization and the extension of the lexicon with historical variants for the seed words. Method (2) is further detailed in Schmidt and Burghardt[42] who evaluated this approach as the most successful compared to other optimizations of lexicon-based approaches in a very similar setting of historical German plays. Both methods can only be used for the binary valence prediction as the lexicon only consists of assignments for positivity and negativity. Thus, we use both methods as baseline only in this specific case.

We refer to (1) as *lb-sentiws* and to (2) as *lb-sentiws-optimized*. There is a possibility that the calculation process results in 0, meaning a text sequence is neither negative nor positive (which is especially frequent for very short text sequences). In this case, we do count the classification as false.

3.2.2 Traditional Machine Learning

Other baseline methods next to lexicon-based sentiment analysis include machine learning techniques often referred to as "traditional" ML (or "statistical" methods) in contrast to deep learning- or neural network-based ML. For these methods, texts are usually represented in a *bag-of-words* (BOW) model in which every text unit is represented in a multidimensional vector space with a numeric value like the *term-frequency* (TF) or the *term-frequency-inverse-document-frequency* (TF-IDF) for every word.[43] It is a standard representation format in computational text analysis for supervised machine learning. We apply TF-based bag-of-words modeling on our annotations and explore the learning algorithms *Multinomial Naïve Bayes* (MNB) and *Support Vector Machine*, both established algorithms in similar text classification settings.[44] For many NLP-tasks like sentiment analysis or text classification, traditional machine learning approaches are outper-

36 Taboada et al. 2011.
37 Schmidt et al. 2018b.
38 Schmidt et al. 2021a.
39 Mohammad / Turney 2013.
40 Hutto / Gilbert 2014.
41 Remus et al. 2010.
42 Schmidt / Burghardt 2018a.
43 Gonzáles-Carvajal et al. 2021.
44 Firmino et al. 2014. More details about the representation format as well as the chosen algorithms can be found in Gonzáles-Carvajal et al. 2021.

formed by more modern approaches like transformer-based language models,[45] however they are still widely used as baseline comparisons.

We implement both approaches via the scikit-learn machine learning library.[46] We refer to the Multinomial Naïve Bayes approach in the following as *bow-mnb* and for the SVM approach as *bow-svm*.[47] As with all machine learning approaches, we train and evaluate the algorithms in a stratified 5 x 5 setting which we describe more in chapter 4.1.

3.2.3 Static Language Models

The idea of static language models is to create vector representations for words based on their surrounding words in a corpus. Geometric differences or similarities of these word vectors represent semantic differences or similarities. The assumption for this representation idea is that words with similar meanings tend to appear in similar contexts. These models are referred to as static as any word has the same vector representation no matter the sentence or surrounding of a word, thus ambiguities of words are not represented in such models.[48] Some of the most popular algorithms to create such models are *Word2Vec*,[49] *GloVe* [50] and *fastText*.[51] These algorithms are trained on large corpora to create static word embeddings. The returned word embeddings can be used as features in various machine learning settings, e.g. as input for neural networks, and have been proven to outperform traditional machine learning in various settings.[52] We included one of the more recent static embedding approaches in our experiment: fastText developed by Facebook.[53] Several studies show that fastText achieves higher accuracies for text classification tasks compared to other static embeddings.[54] Furthermore, *fastText* has been shown to achieve higher accuracies in a sentiment analysis evaluation on German texts compared to other static embeddings.[55] fastText improves on other static word embeddings like *Word2Vec* by using sub-words and characters instead of words as representation vocabulary, thus each word is represented as a bag of character n-grams.

To implement the model, we use the *FLAIR* NLP Framework.[56] We load German fastText embeddings via this library. These embeddings resulted from training on the German Wikipedia. We employ the recommended default implementation for text se-

45 Cf. Shmueli / Ku, 2019; Mishev et al. 2020; Dang et al. 2020; Cortiz 2021; Gonzáles-Carvajal et al. 2021.
46 Pedregoasa et al. 2011.
47 The MNB implementation is based on Schütze et al. 2008, the SVM approach on Chang / Lin 2011 and we use the default settings of scikit-learn.
48 Yu et al. 2019.
49 Mikolov et al. 2013.
50 Pennington et al. 2014.
51 Bojanowski et al. 2017.
52 Cf. Bamler / Mandt 2017. For an in-depth review of the history of word embeddings please see Yu et al. 2019.
53 Bojanowski et al. 2017.
54 Mikolov et al. 2017; Goularas / Kamis 2019.
55 Schmitt et al. 2018.
56 Akbik et al. 2019.

quence classification by the *FLAIR* Framework: A *Gated Recurrent Unit* (GRU)-type *Recurrent Neural Network* (RNN)[57] is used to create document embeddings for the n-grams of every input text unit (in our case the annotation units) as last output state. This single embedding vector is of size 300 and represents the whole input. This text representation is put into a linear layer for the classification task.[58] We use the default hyperparameters of *FLAIR*, which are a learning rate of 0.1, a batch size of 32 to train the network for 12 epochs. We train 5 models with our data in a stratified 5 x 5 setting with 80% as train and 10% as validation and test data respectively. We refer to this model in the following as *fastText*.

3.2.4 Transformer-Based Language Models Pretrained on Contemporary Language

Transformer-based language models like *BERT*, *GLP* and *ELECTRA* have gained a lot of popularity in the NLP community in recent years and have achieved state-of-the-art results for various tasks like sentiment analysis, named entity recognition and question answering.[59] Like most static language models, these models are first pretrained in an unsupervised manner on large text corpora (e.g. the entire Wikipedia) to create word representations. In contrast to static language models these representations are contextualized, meaning the surrounding words of a word change the representation. Thus, these models are also oftentimes referred to as contextualized or dynamic. While there are contextualized embeddings that do not rely on transformer architecture (e.g. *FLAIR* [60] or *ELMo* [61]), we will focus for our analysis of dynamic word embeddings solely on transformer-based language models since they are currently considered state-of-the-art for classification tasks. Some of the most well-known architectures are *BERT*,[62] *distilBERT*,[63] *RoBERTa*,[64] *GPT-2*,[65] and *ELECTRA*.[66] Pretrained models for various languages as well as multilingual models exist. Since we solely perform emotion prediction on German texts, we will however focus on models pretrained on German texts. We distinguish between models that are pretrained or optimized for contemporary language and for rather historical language, which we introduce in the next sub chapter. The models construct contextualized embeddings for a fixed set of tokens: the vocabulary of a model. Specific tokenizers deconstruct text sequences into tokens of the vocabulary which boils down to the character level if no full token exists. By this, the problem of unknown words, which occurs with previous language models, is addressed.

57 Cho et al. 2014.
58 See Joulin et al. 2016 for more details.
59 Cf. Qiu et al. 2020.
60 Akbik et al. 2018.
61 Peters et al. 2018.
62 Devlin et al. 2018.
63 Sanh et al. 2019.
64 Liu et al. 2019.
65 Radford et al. 2019.
66 Clark et al. 2020.

Table 3 summarizes the chosen transformer models pretrained on contemporary language in our evaluation settings as well as the abbreviation we use below. We focused on models that are freely available via the *Hugging Face*-platform,[67] which does include models trained on large German datasets like Wikipedia or book corpora and we do include all German models that are considered as state-of-the-art.[68]

Table 3: Overview of the evaluated transformer-based models pretrained on contemporary language.

MODEL-IDENTIFIER	ARCHITECTURE	HUGGING FACE-LINK	PRETRAINED TEXT	RELATED PAPER (IF AVAILABLE) AND PROVIDER
bert-base-german-cased	*BERT*	Link	Wikipedia, legal texts, news (~ 12 GB)	Deepset
dbmdz-bert-base-german-cased	*BERT*	Link	Wikipedia, books, subtitles, crawled web data, news texts (~ 16 GB)	MDZ Digital Library
electra-base-german-uncased	*ELECTRA*	Link	Wikipedia, Subtitles, News (~ 73 GB)	German-NLP-Group
gbert-large	*BERT*	Link	Crawled web data, Wikipedia, subtitles, book, legal texts (~ 161 GB)	Deepset (Chan et al., 2020)
gelectra-large	*ELECTRA*	Link	Crawled web data, Wikipedia, subtitles, book, legal texts (~ 161 GB)	Deepset (Chan et al., 2020)

67 Wolf et al. 2020.
68 Chan et al. 2020.

To perform text sequence classification with these models, they need to be fine-tuned with labeled data for a downstream task, in our case text sequence classification or, more precisely emotion class prediction. The exact process is dependent on the chosen architecture, for example, in the case of *BERT* the final hidden state of the first token is used as representation of the input sequence and a softmax-classifier is added to the top of *BERT* to predict the probability of labels.[69]

For the fine tuning we select the same hyperparameters for every language model. The hyperparameters are chosen according to the standard recommendations by Devlin et al. in the case of *BERT*,[70] Clark et al. in the case of *ELECTRA* models,[71] and the *Hugging Face* library itself,[72] which we use to perform the correct tokenization and implement the models. Each model is fine-tuned for 4 epochs with a batch size of 32, a learning rate of $4e^{-5}$ and the *Adam* optimizer[73] for stochastic gradient descent. As maximum sequence length for the input text sequences, we chose 128 tokens (instead of the possible 512). The majority of annotation units are sufficiently represented with this length as the corpus statistics show (table 1), longer sequences get truncated. All models are loaded via the hugging-face library and trained using the Google Colab Framework on a *Nvidia Tesla P100* GPU.

3.2.5 Transformer-Based Language Models Pretrained and further Pretrained on Historical or Poetic Language

The performance of downstream tasks of transformer-based models can be improved by pretraining the model on language and text corpora that are linguistically closer to the texts of the specific downstream task.[74] The texts of the downstream task are historical German plays from around 1800, however the models of chapter 3.2.4 are derived from contemporary language, primarily the Wikipedia. Training a model from scratch, if enough text of the specific domain is available, has been proven to be successful for classification tasks.[75] Another way to improve the language model to the task at hand is the method to take a standard pretrained model and further pretrain it for several epochs, meaning the language modeling task of the transformer-based model is fine-tuned. Gururangan et al. differentiate between fine-tuning with texts of the same domain and texts of the same task[76] and were able to show improvements for various tasks and domains with further pretraining. We focus in the following on domain-based pretraining and look at models that are either trained from scratch on texts closer to our language of 1800 German or pretrained on contemporary language and then fine-tuned on texts closer to our language (we refer to the latter as "further pretraining" in the fol-

69 Devlin et al. 2018; For more information about the pretraining and fine-tuning of the specific transformer models please refer to the papers above introducing the specific models.
70 Devlin et al. 2018.
71 Clark et al. 2020.
72 Wolf et al. 2020.
73 Kingma / Ba 2014.
74 Beltagy et al. 2019; Ma et al. 2019; Rietzler et al. 2020; Gururangan et al. 2020; Wada et al. 2021.
75 Beltagy et al. 2019.
76 Gururangan et al. 2020.

lowing chapters). Both, training models from scratch and further pretraining models have been successfully applied for historical German for named entity recognition (NER)[77] and speech type recognition.[78]

Considering models trained from scratch we evaluate several transformer-based architectures that were pretrained on a large German language corpus of newspapers, the *Europeana* newspapers which are provided by the European Digital Library *Europeana*. This corpus has a size of 51 GB and consists of over 8 billion tokens. The texts range from 18th to 20th century, thus including a significant size of historical language compared to contemporary models based on Wikipedia dumps. We evaluate a *BERT* and an *ELECTRA* model trained on the Europeana-corpus, that are offered via the *Hugging Face* platform.[79] Another model that is trained on various literary and historical texts that we evaluate for our task is by Brunner et al.[80]. It has been successfully applied in the area of speech type recognition. As a model that employs domain adaptation by using a general model and further pretraining it, we evaluate *German BERT for literary texts*. It is based on *bert-base-german-dbmdz-cased* (see table 3) and has been further pretrained with the *Corpus of German Language* Fiction,[81] a corpus consisting of over 3,100 German literary texts (mainly novels and short stories) from predominantly 1840–1930. Due to architectural reasons the model, in its default state, can solely be used to predict maximum three classes. Thus, this model is only evaluated for binary valence prediction and triple valence prediction.

Table 4: Overview of the evaluated transformer-based models pretrained or further pretrained on historical German language.

MODEL-IDENTIFIER	ARCHITECTURE	HUGGING FACE-LINK	PRETRAINED TEXT	RELATED PAPER (IF AVAILABLE) AND PROVIDER
bert-base-german-europeana-cased	*BERT* (trained from scratch)	Link	*Europeana* newspaper (51 GB)	MDZ Digital Library (Schweter 2020)

77 Labusch et al. 2019; Schweter / Baiter 2019; Schweter / März 2020.
78 Brunner et al. 2020a.
79 See the GitHub-repository https://github.com/stefan-it/europeana-bert for more information; Schweter 2020.
80 Brunner et al. 2020a.
81 Fischer / Strötgen 2017.

electra-base-german-europeana-cased-discriminator	*ELECTRA* (trained from scratch)	Link	Europeana newspaper (51 GB)	MDZ Digital Library (Schweter 2020)
literary-german-bert	*BERT* (further pretrained)	Link	based on bert-basegerman-dbmdz-cased further pretrained with the Corpus of German-Language-Fiction (mostly prose texts) (~ 1 GB)	Severin Simmler
bert-base-historical-german-rw-cased	*BERT* (trained from scratch)	Link	fairy tales, historical newspapers, magazine articles, narrative texts, texts of Projekt Guutenberg	Brunner et al. 2020a

The hyperparameter settings of the models are the same as with the general transformer-based models in that we rely on the default recommendations of fine tuning the models on the classification task for 4 epochs with a batch size of 32, a learning rate of 4e-5 and the *Adam*-optimizer for stochastic gradient descent. Table 4 illustrates the metadata of the models and introduces the abbreviated names we will use in the following evaluation.

3.2.6 Transformer-Based Language Models Further Pretrained with Texts of the Main Corpus

Similar to the model *german-literary-bert*, we also explored possibilities of small-scale domain adaptation via further pretraining of contemporary models with texts of our main corpus we intend to analyze and use for annotation. We refer to this technique as *further pretraining* with the main corpus texts. However, there is no specific technical difference compared to the generation of models like *german-literary-bert* besides the chosen texts for the continued unsupervised pretraining. Research suggests improvements for classification tasks[82] even with rather small corpora consisting of one million or less sentences[83] of the specific domain. We selected the model *bert-base-german-cased*

82 Rietzler et al. 2020; Gururangan et al 2020.
83 Ma et al. 2019; Rietzler et al. 2020.

(German BERT model pretrained on Wikipedia) and continued pretraining for four epochs. We limit ourselves to just this model to explore the general influence of this domain adaptation in a first experiment. We used the "simpletransformer-library" in its default setting (the library uses the default settings of the base model and settings) to perform the domain adaptive fine-tuning of the language model.

We used two different corpora for further pretraining: (1) our current main corpus consisting of the *GerDracor* corpus (950 plays)[84] and 30 "Kasperl"[85] plays, (2) only the five plays we used for annotation. We prepared the corpora into a format consisting of each sentence of the original texts per line. The sentences were acquired via the *NLTK PunktSentence* segmentation of the *NLTK* package. These text lines are used for the continued pretraining as samples. The tokenization of these samples is done by the default *Hugging Face*-tokenizers. Training material (1) consists of 1,195,018 lines (and thus sentences) with a size of 56 MB and (2) consists of 19,356 lines and has a size of close to 1 MB. We refer to (1) as *bert-base-german-cased-main-corpus* and to (2) as *bert-base-german-cased-annotated-texts*. The fine-tuning for the classification task is implemented in the same way and with the same hyperparameters as for the general transformer-based models, so please refer to chapter 3.2.4 for more information.

4. Results

4.1 Evaluation Concept

The evaluation concept follows current ML-based algorithm evaluation approaches:[86] For all ML-based methods we perform the evaluation in a 5 x 5 stratified cross evaluation. We first split the sub-corpus into 5 shuffled sets of the size of 20% of the entire corpus in a stratified way, meaning all classes are equally distributed in these sets. Four of these sets are used as training data for the algorithms, one for the evaluation/test which is switched through for five runs. Each model of these five runs is saved and tested on the test set. The models are trained according to the settings outlined in chapter 3. For all evaluation metrics we calculate the average values of these runs. The evaluation process differs only (1) for the static embedding for which we separate the test set by 50% in a similar stratified fashion and use one part for the validation of the neural network and one part as test set and (2) for the lexicon-based approaches (see chapter 3.2.1) which need no specific setting due to the rule-based calculation.

As evaluation metrics we report the most common metrics in ML evaluation for single-label classification. We report accuracy (the proportion of correct predictions among all predictions) as major evaluation metric, macro and weighted F1 score (f1-m and f1-w), precision (p-m and p-w), and recall (r-m and r-w). The respective macro

84 Fischer et al. 2019.
85 See for more information: http://lithes.uni-graz.at/maezene/maezene_startseite.html
86 Raschka 2020.

scores are (absolute) averages of the sum value of all classes, thus penalizing low performance in minority classes much stronger while weighted values take the proportion of each class into account. We do not specifically report micro values since the micro values of these metrics are the same as the accuracy in a single label classification setting. In general, all values are better the closer they are to 1.0.[87] We limit the results to the most important ones in the following.[88]

4.2 Binary Valence Prediction

We refer to binary valence prediction as the prediction of an annotation as either positive or negative. Thus, we remove all annotation with emotional movement for this task which leaves us with 5,835 annotations. Most annotations are negative (3,572); the majority baseline is 0.612. Table 5 summarizes the results. For this and the upcoming result tables, we first report random and majority baselines, followed by the lexicon-based methods, traditional machine learning, *fastText*, general transformer-based models, transformer-based models including historical language and models trained with our main corpus.

Table 5: Results of the binary valence prediction. Best three models according to the accuracy value are in bold.

METHOD	ACC	F1-W	F1-M	P-W	P-M	R-W	R-M
random baseline	.500	-	-	-	-	-	-
majority baseline	.612	-	-	-	-	-	-
lb-sentiws	.445	.448	.445	.488	.464	.445	.464
lb-sentiws-optimized	.588	.592	.578	.602	.579	.587	.582
bow-mnb	.742	.740	.724	.739	.730	.742	.721
bow-svm	.685	.635	.591	.713	.725	.685	.608
fasttext	.714	.703	.681	.712	.707	.714	.676
bert-base-german-cased	.804	.792	.803	.803	.796	.804	.789

87 For more information about ML metrics please refer to Jeni et al. 2013; Chicco / Jurman 2020.
88 For a detailed review of all results including further ML-metrics and class-based metrics please refer to this repository: https://files.mi.ur.de/d/8cd40c4193454f99a5b1/

dbmdz-bert-base-german-cased	.804	.791	.802	.803	.795	.804	.788
electra-base-german-uncased	.776	.762	.775	.775	.765	.776	.760
gbert-large	**.821**	**.820**	**.810**	**.820**	**.813**	**.821**	**.808**
gelectra-large	**.825**	**.824**	**.814**	**.824**	**.818**	**.825**	**.812**
bert-base-german-europeana-cased	.798	.797	.786	.797	.788	.798	.785
electra-base-german-europeana-cased-discriminator	.808	.808	.798	.809	.799	.808	.799
literary-german-bert	.799	.798	.787	.798	.789	.799	.786
bert-base-historical-german-rw-cased	**.813**	**.813**	**.803**	**.813**	**.804**	**.813**	**.801**
bert-base-german-cased-main-corpus	.796	.794	.781	.795	.790	.796	.776
bert-base-german-cased-annotated-texts	.809	.809	.798	.808	.801	.809	.795

The lexicon-based methods do not reach the majority baseline according to the accuracy, however the optimization to this text sorts as recommended by Schmidt and Burghardt[89] shows improvement. Traditional ML as well as fastText perform slightly above the majority baseline. Transformer-based models achieve the highest accuracies with the largest models *gbert-large* and *gelectra-large* performing best (up to 83%). We cannot identify a significant performance boost using models based on historical language except for *bert-base-historical-german-rw-cased*, which performs slightly above smaller general transformer-based models. Training with texts of our main corpus does not result in improvements with the approaches applied here.

4.3 Triple Valence Prediction

Triple valence prediction is a three-class classification using the entire annotated corpus with the classes positive (2,267 annotations; 34%), negative (3,566; 54%) and emotional movement (763; 12%). Table 6 illustrates the result for this task. Lexicon-based methods could not be applied in this setting (see chapter 3.2.1).

89 Schmidt / Burghardt 2018.

Table 6: Results of the triple valence prediction.

METHOD	ACC	F1-W	F1-M	P-W	P-M	R-W	R-M
random baseline	.333	-	-	-	-	-	-
majority baseline	.541	-	-	-	-	-	-
bow-mnb	.659	.633	.505	.632	.560	.659	.504
bow-svm	.603	.524	.378	.647	.643	.603	.407
fasttext	.647	.616	.495	.645	.631	.647	.491
bert-base-german-cased	.711	.707	.625	.705	.635	.711	.617
dbmdz-bert-base-german-cased	.716	.714	.636	.712	.644	.716	.629
electra-base-german-uncased	.690	.682	.593	.681	.618	.690	.582
gbert-large	**.740**	**.735**	**.654**	**.733**	**.670**	**.740**	**.645**
gelectra-large	**.748**	**.746**	**.670**	**.745**	**.678**	**.748**	**.664**
bert-base-german-europeana-cased	.718	.714	.632	.712	.644	.718	.624
electra-base-german-europeana-cased-discriminator	.722	.717	.636	.715	.649	.722	.629
literary-german-bert	.718	.716	.638	.716	.644	.718	.634
bert-base-historical-german-rw-cased	**.723**	**.719**	**.637**	**.717**	**.648**	**.723**	**.630**
bert-base-german-cased-main-corpus	.714	.695	.590	.705	.673	.714	.577
bert-base-german-cased-annotated-texts	.709	.705	.626	.703	.638	.709	.618

While all methods perform above the random and majority baseline, accuracies decrease around 5-10% compared to the binary task, which is to be expected. The overall results, however, stay the same with the best models being large BERT and ELECTRA implementations achieving up to 75% accuracy followed, again, by bert-base-historical-german-rw-cased with 72% accuracy. Regarding the macro-values, the problem of imbalanced class distribution becomes apparent, as it also will in the following chapters.

Most models have problems in the prediction of classes with very few annotations (in this case emotional movement). Thus, the macro-values show a significant decrease, which is explained by the low performance on the prediction and recall of emotional movement with a class-precision of 0.484 and a class recall of 0.374. We did not identify a difference between the models concerning this specific problem.

4.4 Main Emotion Class Prediction

The main class prediction is a single-label classification task with six classes. Please refer to table 1 for an illustration of the distribution. The majority baseline is determined by the emotions of suffering with 33% among all annotations.

Table 7: Results of the emotion main class prediction.

METHOD	ACC	F1-W	F1-M	P-W	P-M	R-W	R-M
random baseline	.167	-	-	-	-	-	-
majority baseline	.333	-	-	-	-	-	-
bow-mnb	.451	.409	.342	.460	.453	.451	.333
bow-svm	.392	.304	.229	.479	.475	.392	.245
fasttext	.404	.343	.292	.379	.346	.404	.281
bert-base-german-cased	.512	.508	.471	.509	.485	.512	.462
dbmdz-bert-base-german-cased	.517	.511	.472	.511	.485	.517	.465
electra-base-german-uncased	.474	.449	.391	.456	.431	.474	.391
gbert-large	**.545**	**.539**	**.500**	**.540**	**.517**	**.545**	**.492**
gelectra-large	**.564**	**.558**	**.517**	**.560**	**.537**	**.564**	**.508**
bert-base-german-europeana-cased	**.528**	**.518**	**.477**	**.522**	**.502**	**.528**	**.469**
electra-base-german-europeana-cased-discriminator	.525	.509	.459	.517	.498	.525	.452
bert-base-historical-german-rw-cased	.524	.519	.479	.519	.490	.524	.473

bert-base-german-cased-main-corpus	.492	.458	.392	.490	.483	.492	.395
bert-base-german-cased-annotated-texts	.505	.500	.464	.501	.478	.505	.457

The overall results stay the same with transformer-based models outperforming traditional ML and the static embedding approach (table 7). Due to the higher number of classes the overall accuracies settle in areas of around 50%. While *gbert-large* and *gelectra-large* stay the best models, the historical models do perform slightly better (1-2%) than smaller contemporary counterparts like *bert-base-german-cased*. Training with the main corpus does not yield improvements. Looking at the class-based performance for the best model *gelectra*-large, F1-scores range from as low as 35% (for hate represented within the corpus 9%) to up to 67% (for emotions of affection represented in the corpus with 19%) signifying again the imbalance problem.

4.5 Sub-emotion Prediction

The sub-emotion prediction is a classification task with 13 classes. Thus, the random and majority baseline are very low. See table 1 for an illustration of the emotion distributions among the annotations. We summarize the results of the sub-emotion prediction in table 8.

Table 8: Results of the sub-emotion prediction.

METHOD	ACC	F1-W	F1-M	P-W	P-M	R-W	R-M
random baseline	.077	-	-	-	-	-	-
majority baseline	.151	-	-	-	-	-	-
bow-mnb	.348	.298	.191	.348	.294	.348	.213
bow-svm	.284	.248	.178	.362	.410	.284	.181
fasttext	.289	.241	.151	.280	.213	.289	.175
bert-base-german-cased	.428	.417	.346	.418	.383	.428	.340
dbmdz-bert-base-german-cased	.430	.417	.335	.415	.361	.430	.332
electra-base-german-uncased	.358	.320	.215	.336	.270	.358	.231

gbert-large	**.467**	**.461**	**.409**	**.463**	**.451**	**.467**	**.398**
gelectra-large	**.460**	**.436**	**.325**	**.448**	**.382**	**.460**	**.330**
bert-base-german-europeana-cased	.420	.400	.310	.407	.365	.420	.307
electra-base-german-europeana-cased-discriminator	.416	.373	.251	.371	.277	.416	.269
bert-base-historical-german-rw-cased	**.444**	**.436**	**.368**	**.440**	**.422**	**.444**	**.358**
bert-base-german-cased-main-corpus	.379	.326	.207	.347	.260	.379	.231
bert-base-german-cased-annotated-texts	.425	.415	.342	.414	.375	.425	.335

Although the increase of classes is significant, the drop-off in accuracies is not that strong compared with the main emotion class prediction. The overall results stay equivalent to previous classification tasks with *gbert-large*, *gelectra-large* and *bert-base-historical-german-rw-cased* as best performing models. The difference between macro and weighted values become stronger since some models tend to not predict classes with very low representation in the corpus at all in the evaluation (e.g. desire, friendship, Schadenfreude).

5. Discussion

While our project is still in an early phase for the annotation part as well as the computational side, we were able to gain some first insights based on the preliminary results of a first annotation and evaluation study.

First, focusing on the annotation scheme and process towards the literary scholar perspective has led to positive feedback by the annotators since the flexible annotation structure is more in line with (1) the perceived emotional expressions in the plays and (2) is closer to the usual annotation process of literary scholars. The low agreements, although in line with previous research[90], and the varied annotation segments pose additional challenges for the NLP-approach. We plan to improve upon the agreement with further training for the annotators and sophistication of the annotation guidelines. Furthermore, we plan to explore other fuzzy agreement metrics that fit more our varied annotation.[91] The varied annotation poses also problems in the creation of valid training material for the ML-approaches since it is difficult to dissolve segment-based dis-

90 Alm / Sproat 2005; Sprugnoli et al. 2016; Schmidt et al. 2018a; Schmidt et al. 2019a; Schmidt et al. 2019b.

91 Kirilenko / Stepchenkova 2016.

agreements among annotations. We plan to deal with this problem by implementing a post-annotation process in which annotators create a *consensus* annotation guided by a literary scholar by discussing their disagreements. These annotation results can be used for the ML approaches and will also likely increase accuracy since the models do not have to deal with annotation conflicts.

While the number of annotations is still limited for thorough evaluation studies of the computational emotion prediction, there are some general results giving us directions for the future work. Our best evaluated models achieve accuracies up to 83% for a binary valence classification and up to 75% with an additional class. Indeed, this is up to pair with state-of-the-art results in similar sentiment analysis settings with contemporary language[92] which is rather promising regarding the challenging material, the currently few and eventually disagreeing annotations and the fact that we mostly used models in their default setting. The overall results are consistent over all classification tasks. The best models are transformer-based models in general and specifically the large German models *gbert-large* and *gelectra-large*.[93] Traditional ML as well as the prediction pipeline based on *fastText* performed similarly below transformer-based models with a Multinomial Naïve Bayes with term frequency features performing best among the methods. We can validate previous findings that ML methods in general outperform lexicon-based methods. It is striking, however, that the improved method by Schmidt and Burghardt,[94] which does include the extension of historical variants into the lexicon does yield better results pointing again to the historical language being the major problem of this material.

The usage of transformer-based models trained on historical language or further trained on historical and poetic language led to no significant improvements in our setting. Looking closer, however, we find that while these models are trained on historical and sometimes poetic language, a large part of these corpora is made not just of language the 18th or 19th century make but also comprise large chunks of language of the 20th century e.g. the *Corpus of German Language Fiction* [95] in the case of *literary-german-bert*. The same holds true for the various *Europeana* models.[96] Further pretraining with our main corpus did also show no significant changes. In the case of *GerDracor*, the problem might be similar to the other historical models in that the majority of the texts are actually from the 20th century. However, the main problem is likely the limited amount of training texts which is far too low compared to other fine-tuning research. Furthermore, the fine-tuning is also limited on a resource perspective since we performed the fine-tuning solely for 4 epochs which is rather low compared to usual recommendations.[97] Summing up, we argue that the language of these models is still too "modern" for our material. We want to further pursue the language model fine tuning strategy by collecting and filtering corpora towards language around 1800 to explore if

92 Cf. Yang et al. 2019; Munikar et al. 2019; Cao et al. 2020; Dang et al. 2020.
93 Chan et al. 2020.
94 Schmidt / Burghardt 2018a.
95 Fischer / Strötgen 2017.
96 Schweter 2020.
97 Beltagy et al. 2019; Ma et al. 2019; Rietzler et al. 2020; Gururangan et al. 2020; Wada et al. 2021.

we can replicate improvements found in similar settings with German historical language.[98]

One general limitation of our models is the usage of default settings. While this was deemed fitting to perform first explorations of these models, we are confident that we can boost accuracies with hyperparameter optimization via grid search, which might be sufficient to perform binary and triple valence detection on a larger scale. The classification tasks with 6 to 13 classes, however, need more sophisticated adjustments. Classification tasks with that many classes are rare and challenging by design with one problem being the class imbalance and several of the classes being annotated too few times. We plan to explore recommendations to deal with this problem like training with minority oversampling instead of stratified samples in future work.[99] The continued acquisition of more annotations of current minority classes will certainly help . Indeed, emotions like desire and friendship are very much dependent on the selected plays and we anticipate more annotations in the upcoming project phases. Lastly, we also intend to investigate the adjustment of our conceptual approach on several limits. Integrating a neutral non-annotation class in our approaches or interpreting the prediction as a multi-label classification task could represent our annotation process better than the current classification approach. Furthermore, we plan to explore the influence of the classification of various structural text units like token-, n-gram- or speech-based classification. We will continue our annotations and the exploration of emotion prediction methods to further advance towards the possibility of large-scale emotion analysis on our dataset.

References

Francisca Adoma Acheampong / Chen Wenyu / Henry Nunoo-Mensah: Text-Based Emotion Detection: Advances, Challenges, and Opportunities. In: Engineering Reports 2 (2020), No. 7. [online].

Alan Akbik / Tanja Bergmann / Duncan Blythe / Kashif Rasul / Stefan Schweter / Roland Vollgraf: FLAIR: An easy-to-use framework for state-of-the-art NLP. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations). Minneapolis, MN 2019, pp. 54–59.

Cecilia Alm Ovesdotter / Richard Sproat: Emotional Sequencing and Development in Fairy Tales. In: ACII, 2005. [online]

Thomas Anz: Todesszenarien: literarische Techniken zur Evokation von Angst, Trauer und anderen Gefühlen. In: Emotionale Grenzgänge. Konzeptualisierungen von Liebe, Trauer und Angst in Sprache und Literatur. Ed. by Lisanne Ebert / Carola Gruber / Benjamin Meisnitzer / Sabine Rettinger. Würzburg 2011, pp. 54–59.

98 Labusch et al. 2019; Schweter / Baiter, 2019; Brunner et al. 2020; Schweter / März 2020.
99 Buda et al. 2018.

Robert Bamler / Stephan Mandt: Dynamic Word Embeddings. In: Proceedings of the 34th International Conference on Machine Learning. PMLR 70. Sydney 2017, pp. 380–389). [online]
Iz Beltagy / Kyle Lo / Arman Cohan: SciBERT: A Pretrained Language Model for Scientific Text. arXiv:1903.10676 [cs], 10 September 2019. [online]
Piotr Bojanowski / Edouard Grave / Armand Joulin / Tomas Mikolov: Enriching Word Vectors with Subword Information. arXiv:1607.04606 [cs], 19 June 2017. [online]
Annelen Brunner / Ngoc Duyen / Tanja Tu / Lukas Weimer / Fotis Jannidis (2020a): To BERT or Not to BERT – Comparing Contextual Embeddings in a Deep Learning Architecture for the Automatic Recognition of Four Types of Speech, Thought and Writing Representation. In: Proceedings of the 5th Swiss Text Analytics Conference and the 16th Conference on Natural Language Processing. SwissText/KONVENS 2020. Zurich 2020. [online]
Annelen Brunner / Stefan Engelberg / Fotis Jannidis / Ngoc Duyen / Tanja Tu / Lukas Weimer (2020b): Corpus REDEWIEDERGABE. In: Proceedings of the 12th Language Resources and Evaluation Conference. Marseille 2020, pp. 803–812. [online]
Mateusz Buda / Atsuto Maki / Maciej A. Mazurowski: A systematic study of the class imbalance problem in convolutional neural networks. In: Neural Networks 106 (2018), pp. 249–259.
Lihong Cao / Sancheng Peng / Pengfei Yin / Yongmei Zhou / Aimin Yang / Xinguang Li: A Survey of Emotion Analysis in Text Based on Deep Learning. In: 2020 IEEE 8th International Conference on Smart City and Informatization (ISCI). Guangzhou 2020, pp. 81–88. [online]
Branden Chan / Stefan Schweter / Timo Möller: German's Next Language Model. arXiv:2010.10906 [cs], 3. Dezember 2020. [online]
Chih-Chung Chang / Chih-Jen Lin: LIBSVM: A Library for Support Vector Machines. ACM Transactions on Intelligent Systems and Technology 2 (April 2011), No. 3, pp. 1–27. [online]
Davide Chicco / Giuseppe Jurman: The Advantages of the Matthews Correlation Coefficient (MCC) over F1 Score and Accuracy in Binary Classification Evaluation. In: BMC Genomics 21 (2. Januar 2020). [online]
Kyunghyun Cho / Bart van Merriënboer / Dzmitry Bahdanau / Yoshua Bengio: On the Properties of Neural Machine Translation: Encoder–Decoder Approaches. In: Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation. Doha 2014, pp. 103–111. [online]
Kevin Clark / Minh-Thang Luong / Quoc V. Le / Christopher D. Manning: ELECTRA: Pre-Training Text Encoders as Discriminators Rather Than Generators, 2020. arXiv:2003.10555 [cs], 23 March 2020. [online]
Diogo Cortiz: Exploring Transformers in Emotion Recognition: a comparison of BERT, DistillBERT, RoBERTa, XLNet and ELECTRA. arXiv:2104.02041 [cs], 5 April 2021. [online]

Nhan Dang Cach / María N. Moreno-García / Fernando De la Prieta: Sentiment Analysis Based on Deep Learning: A Comparative Study. In: Electronics 9 (2020), 483. [online]
Jacob Devlin / Ming-Wei Chang / Kenton Lee / Kristina Toutanova: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805 [cs], 24 May 2018. [online]
Karl Eibl: Gotthold Ephraim Lessing: Miss Sara Sampson: ein bürgerliches Trauerspiel. Frankfurt / Main 1971.
Gius Evelyn / Jan Christoph Meister / Marco Petris / Malte Meister / Christian Bruck / Janina Jacke / Mareike Schuhmacher / Marie Flüh / Jan Horstmann: CATMA [Computer software] 2020. [online]
Alves Firmino / André Luiz / Cláudio de Souza Baptista / Anderson Almeida Firmino / Maxwell Guimarães de Oliveira / Anselmo Cardoso de Paiva: A Comparison of SVM Versus Naive-Bayes Techniques for Sentiment Analysis in Tweets: A Case Study with the 2013 FIFA Confederations Cup. In: Proceedings of the 20th Brazilian Symposium on Multimedia and the Web. WebMedia '14. New York, NY 2014, pp. 123–130. [online]
Frank Fischer / Ingo Börner / Mathias Göbel / Angelika Hechtl / Christopher Kittel / Carsten Milling / Peer Trilcke: Programmable Corpora: Introducing DraCor, an Infrastructure for the Research on European Drama. In: DH2019. Conference Abstracts. Utrecht 2019. [online]
Frank Fischer / Jannik Strötgen: Corpus of German-Language Fiction (txt). 2017. [online]
Santiago González-Carvajal / Eduardo C. Garrido-Merchán: Comparing BERT against traditional machine learning text classification. In: arXiv:2005.13012 [cs, stat], 12 January 2021. [online]
Dionysis Goularas / Sani Kamis: Evaluation of Deep Learning Techniques in Sentiment Analysis from Twitter Data. In: 2019 International Conference on Deep Learning and Machine Learning in Emerging Applications (Deep-ML). Istanbul 2019, pp. 12–17. [online]
Suchin Gururangan / Ana Marasović / Swabha Swayamdipta / Kyle Lo / Iz Beltagy / Doug Downey / Noah A. Smith: Don't Stop Pretraining: Adapt Language Models to Domains and Tasks. arXiv:2004.10964 [cs], 5 May 2020. [online]
C. J. Hutto / Eric Gilbert: VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. In: Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media, 2014. [online]
Arthur M. Jacobs: Sentiment Analysis for Words and Fiction Characters From the Perspective of Computational (Neuro-)Poetics. In: Frontiers in Robotics and AI 6 (2019). [online]
László A. Jeni / Jeffrey F. Cohn / Fernando De La Torre: Facing Imbalanced Data Recommendations for the Use of Performance Metrics. In: International Conference on Affective Computing and Intelligent Interaction and Workshops. Proceedings. 2013, pp. 245–251. [online]

Armand Joulin / Edouard Grave / Piotr Bojanowski / Tomas Mikolov: Bag of Tricks for Efficient Text Classification. arXiv:1607.01759 [cs], 9 August 2016. [online]
Kaisla Kajava / Emily Öhman / Piao Hui / Jörg Tiedemann: Emotion Preservation in Translation: Evaluating Datasets for Annotation Projection. In: Proceedings of Digital Humanities in Nordic Countries (DHN 2020). Aachen 2020. pp. 38–50. [online]
Tuomo Kakkonen / Gordana Galić Kakkonen: SentiProfiler: Creating Comparable Visual Profiles of Sentimental Content in Texts. In: Proceedings of the Workshop on Language Technologies for Digital Humanities and Cultural Heritage. Hissar 2011, pp. 62–69. [online]
Sarma Kameswara Prathusha / Yingyu Liang / Bill Sethares: Domain Adapted Word Embeddings for Improved Sentiment Classification. In: Proceedings of the Workshop on Deep Learning Approaches for Low-Resource NLP. Melbourne: Association for Computational Linguistics. Melbourne 2018, pp. 51–59. [online]
Evgeny Kim / Roman Klinger: Frowning Frodo, Wincing Leia, and a Seriously Great Friendship: Learning to Classify Emotional Relationships of Fictional Characters. In: arXiv:1903.12453 [cs], 1. April 2019. [online]
Diederik P. Kingma / Jimmy Ba: Adam: A Method for Stochastic Optimization. In: arXiv:1412.6980 [cs], 29. January 2017. [online]
Andrei P. Kirilenko / Svetlana Stepchenkova: Inter-Coder Agreement in One-to-Many Classification: Fuzzy Kappa. In: PLOS ONE 11 (2 March 2016), 3, e0149787. [online]
Kai Labusch / Clemens Neudecker / David Zellhofer: BERT for Named Entity Recognition in Contemporary and Historical German. In Proceedings of the 15th Conference on Natural Language Processing. Erlangen 2019, pp. 8–11. [online]
J. Richard Landis / Gary G. Koch: The Measurement of Observer Agreement for Categorical Data. In: Biometrics 33 (1977), No. 1, pp. 159–174. [online]
Yinhan Liu / Myle Ott / Naman Goyal / Jingfei Du / Mandar Joshi / Danqi Chen / Omer Levy / Mike Lewis / Luke Zettlemoyer / Veselin Stoyanov: RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv:1907.11692 [cs], 26 July 2019. [online]
Xiaofei Ma / Peng Xu / Zhiguo Wang / Ramesh Nallapati / Bing Xiang: Domain Adaptation with BERT-based Domain Classification and Data Selection. In: Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019). Hong Kong 2019, pp. 76–83. [online]
Tomas Mikolov / Ilya Sutskever / Kai Chen / Greg Corrado / Jeffrey Dean: Distributed Representations of Words and Phrases and their Compositionality. arXiv:1310.4546 [cs, stat], 16 October 2013. [online]
Tomas Mikolov et al.: Advances in Pre-Training Distributed Word Representations. In: arXiv:1712.09405 [cs], 26 December 2017). [online]
Kostadin Mishev / Ana Gjorgjevikj / Irena Vodenska / Lubomir T. Chitkushev / Dimitar Trajanov: Evaluation of Sentiment Analysis in Finance: From Lexicons to Transformers. In: IEEE Access 8 (2020), pp. 131662–131682. [online]

Paul Mog: Ratio und Gefühlskultur: Studien zu Psychogenese und Literatur im 18. Jahrhundert. Tübingen 1976.
Robert Remus / Uwe Quasthoff / Gerhard Heyer: SentiWS - A Publicly Available German-language Resource for Sentiment Analysis. In: Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10). Valletta 2010. [online]
Saif Mohammad: From Once Upon a Time to Happily Ever After: Tracking Emotions in Novels and Fairy Tales. In: Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities. Portland, OR 2011, pp. 105–114. [online]
Saif M. Mohammad / Peter D. Turney: Nrc emotion lexicon. In: National Research Council, Canada 2 (2013).
Cornelia Mönch: Abschrecken oder Mitleiden: das deutsche bürgerliche Trauerspiel im 18. Jahrhundert: Versuch einer Typologie. Tübingen 1993.
Luis Moßburger / Felix Wende / Kay Brinkmann / Thomas Schmidt: Exploring Online Depression Forums via Text Mining: A Comparison of Reddit and a Curated Online Forum. In: Proceedings of the Fifth Social Media Mining for Health Applications Workshop & Shared Task. Barcelona 2020, pp. 70–81. [online]
Manish Munikar / Sushil Shakya / Aakash Shrestha: Fine-grained Sentiment Classification using BERT. arXiv:1910.03474 [cs, stat], 4 October 2019. [online]
Eric T. Nalisnick / Henry S. Baird: Character-to-Character Sentiment Analysis in Shakespeare's Plays. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). Sofia 2013, pp. 479–483. [online]
Winfried Nolting: Studien zu einer Geschichte der literarischen Empfindung. Bd. 1: Die Dialektik der Empfindung: Lessings Trauerspiele „Miss Sara Sampson" und „Emilia Galotti". Stuttgart 1986.
Emily Sofi Öhman / Kaisla S. A. Kajava: Sentimentator: Gamifying Fine-grained Sentiment Annotation. In: Proceedings of Digital Humanities in the Nordic Countries 2018. Aachen 2018, pp. 98–110. [online]
Fabian Pedregosa et al.: Scikit-learn: Machine learning in Python. In: Journal of machine learning research 12 (2011), pp. 2825–2830.
Jeffrey Pennington / Richard Socher / Christopher Manning: GloVe: Global Vectors for Word Representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Doha 2014, pp. 1532–1543. [online]
Matthew E. Peters / Mark Neumann / Mohit Iyyer / Matt Gardner / Christopher Clark / Kenton Lee / Luke Zettlemoyer: Deep contextualized word representations. arXiv:1802.05365 [cs], 22 March 2018. [online]
Federico Pianzola / Simone Rebora / Gerhard Lauer: Wattpad as a Resource for Literary Studies. Quantitative and Qualitative Examples of the Importance of Digital Social Reading and Readers' Comments in the Margins. In: PLOS ONE 15 (15 January 2020), 1, e0226708. [online]
Lothar Pikulik: „Bürgerliches Trauerspiel" und Empfindsamkeit. Köln et al. 1966.

Xipeng Qiu / Tianxiang Sun / Yige Xu / Yunfan Shao / Ning Dai / Xuanjing Huang: Pre-trained Models for Natural Language Processing: A Survey. arXiv:2003.08271 [cs], 24 April 2020. [online]
Alec Radford / Jeffrey Wu / Rewon Child / David Luan / Dario Amodei / Ilya Sutskever: Language Models Are Unsupervised Multitask Learners. 2019. [online]
Sebastian Raschka: Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning. arXiv:1811.12808 [cs, stat], 10 November 2020. [online]
Andrew J. Reagan / Lewis Mitchell / Dilan Kiley / Christopher M. Danforth / Peter Sheridan Dodds: The emotional arcs of stories are dominated by six basic shapes. In: EPJ Data Science 5 (2016), 31. [online]
Alexander Rietzler / Sebastian Stabinger / Paul Opitz / Stefan Engl: Adapt or Get Left Behind: Domain Adaptation through BERT Language Model Finetuning for Aspect-Target Sentiment Classification. In: Proceedings of the 12th Language Resources and Evaluation Conference. Marseille 2020, pp. 4933–4941. [online]
Spyridon Samothrakis / Maria Fasli: Emotional Sentence Annotation Helps Predict Fiction Genre. In: PLOS ONE 10, (2 November 2015), 11, e0141922. [online]
Victor Sanh / Lysandre Debut / Julien Chaumond / Thomas Wolf: DistilBERT, a Distilled Version of BERT: Smaller, Faster, Cheaper and Lighter. In: 5th Workshop on Energy Efficient Machine Learning and Cognitive Computing. 2020. [online]
Gerhard Sauder: Empfindsamkeit. Stuttgart 1974.
Hans-Jürgen Schings: Der mitleidigste Mensch ist der beste Mensch: Poetik des Mitleids von Lessing bis Büchner. München 1980.
Thomas Schmidt / Manuel Burghardt / Katrin Dennerlein (2018a): Sentiment Annotation of Historic German Plays: An Empirical Study on Annotation Behavior. In: Proceedings of the Workshop on Annotation in Digital Humanities (annDH 2018). Sofia 2018, pp. 47–52. [online]
Thomas Schmidt / Manuel Burghardt / Christian Wolff (2018b): Herausforderungen für Sentiment Analysis-Verfahren bei literarischen Texten. In: INF-DH-Workshop 2018 – Im Spannungsfeld zwischen Tool-Building und Forschung auf Augenhöhe. Bonn 2018. [online]
Thomas Schmidt / Manuel Burghardt (2018a): An Evaluation of Lexicon-based Sentiment Analysis Techniques for the Plays of Gotthold Ephraim Lessing. In: Proceedings of the Second Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature. Santa Fe, NM 2018, pp. 139–149. [online]
Thomas Schmidt / Manuel Burghardt (2018b): Toward a Tool for Sentiment Analysis for German Historic Plays. In: COMHUM 2018b: Book of Abstracts for the Workshop on Computational Methods in the Humanities 2018. Lausanne 2018, pp. 46–48. [online]
Thomas Schmidt et al. (2019a): Inter-Rater Agreement and Usability: A Comparative Evaluation of Annotation Tools for Sentiment Annotation. In: Informatik 2019: 50 Jahre Gesellschaft für Informatik – Informatik für Gesellschaft (Workshop-Beiträge). Bonn 2019, pp. 121–133. [online]

Thomas Schmidt / Manuel Burghardt / Katrin Dennerlein / Christian Wolff (2019b): Sentiment Annotation for Lessing's Plays: Towards a Language Resource for Sentiment Analysis on German Literary Texts. In: LDK. 2nd Conference on Language, Data and Knowledge 2019. Leipzig 2019. [online]
Thomas Schmidt / Manuel Burghardt / Christian Wolff (2019c): Toward Multimodal Sentiment Analysis of Historic Plays: A Case Study with Text and Audio for Lessing's Emilia Galotti. In: Proceedings of the Digital Humanities in the Nordic Countries 4th Conference. Workshop Proceedings. Copenhagen 2019, pp. 405–414. [online]
Thomas Schmidt: Distant Reading Sentiments and Emotions in Historic German Plays. In: DH_Budapest_2019. Abstract Booklet. Budapest 2019, pp. 57–60. [online]
Thomas Schmidt et al. (2020a): Der Einsatz von Distant Reading auf einem Korpus deutschsprachiger Songtexte. In: DHd 2020. Conference Abstracts. Paderborn 2020, pp. 296–300. [online]
Thomas Schmidt / Florian Kaindl / Christian Wolff (2020b): Distant Reading of Religious Online Communities: A Case Study for Three Religious Forums on Reddit. In: Proceedings of the Digital Humanities in the Nordic Countries 5th Conference (DHN 2020). Riga 2020. [online]
Thomas Schmidt / Johanna Dangel / Christian Wolff (2021a): SentText: A Tool for Lexicon-based Sentiment Analysis in Digital Humanities. In Information between Data and Knowledge. Ed. by. Werner Hülsbusch. Glückstadt 2021, pp. 156-172. [online]
Thomas Schmidt / Katrin Dennerlein / Christian Wolff (2021b): Towards a Corpus of Historical German Plays with Emotion Annotations. In: LDK 2021. 3rd Conference on Language, Data and Knowledge LDK. Zaragoza 2021 [accepted].
Thomas Schmidt / Isabella Engl / David Halbhuber / Christian Wolff (2021c): Comparing Live Sentiment Annotation of Movies via Arduino and a Slider with Textual Annotation of Subtitles. In: Post-Proceedings of the 5th Conference Digital Humanities in the Nordic Countries. Riga 2021. [online]
Martin Schmitt et al.: Joint Aspect and Polarity Classification for Aspect-based Sentiment Analysis with End-to-End Neural Networks. In: arXiv:1808.09238 [cs], 28 August 2018. [online]
Anja Schonlau: Emotionen im Dramentext: eine methodische Grundlegung mit exemplarischer Analyse zu Neid und Intrige 1750–1800. Berlin et al. 2017.
Georg-Michael Schulz: Tugend, Gewalt und Tod: das Trauerspiel der Aufklärung und die Dramaturgie des Pathetischen und des Erhabenen. Tübingen 1988.
Hinrich Schütze / Christopher D Manning / Prabhakar Raghavan: Introduction to information retrieval. Cambridge 2008.
Stefan Schweter: Europeana BERT and ELECTRA models. 2020. [online]
Stefan Schweter / Johannes Baiter: Towards Robust Named Entity Recognition for Historic German. In: Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019). Florence 2019, pp. 96–103. [online]

Stefan Schweter / Luisa März: Triple E – Effective Ensembling of Embeddings and Language Models for NER of Historical German. In: CLEF 2020. Thessaloniki 2020. [online]
Boaz Shmueli / Lun-Wei Ku: SocialNLP EmotionX 2019 Challenge Overview: Predicting Emotions in Spoken Dialogues and Chats. arXiv:1909.07734 [cs], 18 September 2019. [online]
Rachele Sprugnoli / Sara Tonelli / Alessandro Marchetti / Giovanni Moretti: Towards Sentiment Analysis for Historical Texts. In: Digital Scholarship in the Humanities 31 (2016): pp. 762–72. [online]
Maite Taboada / Julian Brooke / Milan Tofiloski / Kimberly Voll / Manfred Stede: Lexicon-Based Methods for Sentiment Analysis. In: Computational Linguistics 37 (Juni 2011), 2, pp. 267–307. [online]
Shoya Wada / Toshihiro Takeda / Shiro Manabe / Shozo Konishi / Jun Kamohara / Yasushi Matsumura: Pre-training technique to localize medical BERT and enhance biomedical BERT. arXiv:2005.07202 [cs], 25 February 2021. [online]
Hermann Wiegmann (Ed): Die ästhetische Leidenschaft: Texte zur Affektenlehre im 17. und 18. Jahrhundert. Hildesheim 1987.
Thomas Wolf et al.: HuggingFace's Transformers: State-of-the-art Natural Language Processing. In: arXiv:1910.03771 [cs], 13 July 2020. [online]
Ian Wood / John P. McCrae / Vladimir Andryushechkin / Paul Buitelaar (2018a): A Comparison Of Emotion Annotation Schemes And A New Annotated Data Set. In: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018). Miyazaki 2018. [online]
Ian Wood et al. (2018b): A Comparison of Emotion Annotation Approaches for Text. In: Information 9 (2018), 117. [online]
Kisu Yang et al.: EmotionX-KU: BERT-Max based Contextual Emotion Classifier. In: arXiv:1906.11565 [cs], 22 August 2019. [online]
Mehmet Yavuz Can: Analyses of Character Emotions in Dramatic Works by Using EmoLex Unigrams. In: Proceedings of the Seventh Italian Conference on Computational Linguistics. Bologna 2021. [online]
Shanshan Yu / Jindian Su / Da Luo: Improving BERT-Based Text Classification With Auxiliary Sentence and Domain Knowledge. In: IEEE Access 7 (2019): pp. 176600–176612. [online]