Melnyk, Lidiia, and Feld, Linda (2022) Sentiment analysis and stance detection on German YouTube comments on gender diversity. Journal of Computer-Assisted Linguistic Research 6: 59-86. ISSN: 2530-9455.

Sentiment Analysis and Stance Detection on German YouTube Comments on Gender Diversity

Lidiia Melnyk*, Linda Feld*
Friedrich Schiller University Jena, Germany
*Corresponding author: lidiia.melnyk@uni-jena.de; linda.feld@uni-jena.de

Received: 26 July 2022 / Accepted: 9 October 2022 / Published: 25 November 2022

Abstract
This paper explores different options of detecting the stance of German YouTube comments regarding the topic of gender diversity and compares the respective results with those of sentiment analysis, showing that these are two very different NLP tasks focusing on distinct characteristics of the discourse. While an already existing model was used to analyze the comments' sentiment (BERT), the comments' stance was first annotated and then used to train different models – SVM with TF-IDF, DistilBERT, LSTM and CNN – for predicting the stance of unseen comments. The best results were achieved by the CNN, reaching 78.3% accuracy (92% after dataset normalization) on the test set. Whereas the most common stance identified in the comments is a neutral one (neither completely in favor nor completely against gender diversity), the overall sentiment of the discourse turns out to be negative. This shows that the discourse revolving around the topic of gender diversity in YouTube comments is filled with strong opinions, on the one hand, but also opens up a space for anonymously inquiring and learning about the topic and its implications, on the other. Our research thereby (1) contributes to the understanding and application of different NLP tasks used to predict the sentiment and stance of unstructured textual data, and (2) provides relevant insights into society's attitudes towards a changing system of values and beliefs.

Keywords: stance detection, sentiment analysis, BERT, neural networks, annotation, YouTube comments, gender diversity

1. Introduction

Sex and gender are two fundamental aspects of a person's identity. However, people define these two concepts very differently, and the question of the existence of diverse genders (and sexes) outside the binary system sparks emotionally loaded controversies driven by a wide

DOI: HTTPS://DOI.ORG/10.4995/JCLR.2022.18224
PUBLISHER: UNIVERSITAT POLITÈCNICA DE VALÈNCIA.

range of opinions and arguments. Whereas sexual minorities and their rights have officially been acknowledged by the German law, their existence is still being questioned and even denied by a large part of the society. In the comments sections below YouTube videos on the topic, people get involved in heated discussions to express their opinions in support of, questioning, or against gender diversity. In describing and evaluating a comment in an online environment like YouTube, beside the comment's content, one can focus on different qualities of that comment. While one major property of the comment is the author's stance towards the topic in question, the sentiment of the comment can play an important role in the way the comment contributes to the discussion. Hence, we conducted both Sentiment Analysis (SA) of the comments as well as Stance Detection (SD) to (1) get an idea of the overall sentiment of the discourse and (2) evaluate how the stance of a comment relates to its sentiment.

For our analysis, we created a corpus consisting of 383,000 comments that were scraped from YouTube and partly annotated for stance according to specific annotation guidelines. Based on a thorough review of previous studies regarding SA methods, we decided to carry out SA with the transformer-based classification model BERT, labeling the comments for positive, negative, and neutral sentiment. For determining the comments' stance (in favor, against, or neutral), we tested four different methods of SD using the annotated data. The first method we tested was vectorizing the comments with TF-IDF and classifying them using a Support Vector Machine. As a second method we classified the comments using three different neural network architectures: a distilled version of BERT, an LSTM, and a CNN and experimented with different activation functions and varying numbers of layers. Having identified the SD method with the highest accuracy as well as categorical precision and recall, we compared the results of SD with the results of SA in order to identify possible differences and the coreference points. The summary of our research methodology is shown in the picture below.

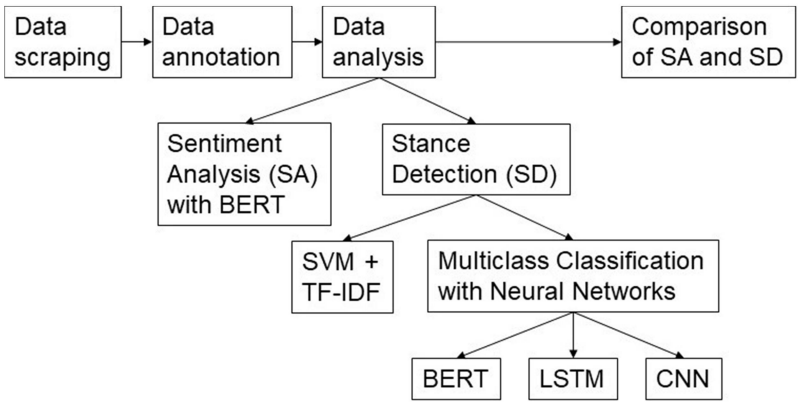

FIGURE 1. RESEARCH METHODOLOGY

SD on an unbalanced dataset scraped from real life YouTube comments is an ambitious task due to the nature of the data itself. The language and structure of the comments can hardly be standardized, which also influences the performance of the algorithms in stance identification. While the Convolutional Neural Networks (CNN) displayed the best results of accuracy,

precision, and F1 score for each of the classes, none of the different methods of SD outlined above was able to provide equally good performance for the underrepresented classes. Therefore, adding more training data for the underrepresented classes as well as continuous fine-tuning of the model is recommended for future research.

2. Corpus creation and description

2.1. Scraping preparation

For the purpose of creating our own corpus of German YouTube comments on the topic of gender diversity, comments were scraped from the comments sections under YouTube videos touching upon this very topic. For choosing these videos, we generated a list of keywords with (positive, negative, and neutral) terms referring to different genders (and sexes) as well as terms otherwise related to the topic of gender diversity. The keywords can be subdivided into the following groups: trans*, non-binary, and inter* gender (including terms related to these).[1]

The keywords were then used in a Python script that retrieved the video IDs of all the videos that were found on YouTube as a result of the keyword-based search. The total number of unique links to videos that had at least one comment under the video is 450. In our script, we applied a language filter to the videos to only include videos in German.

For scraping the comments, we used a JavaScript code in Google Apps Script. The JavaScript code made a request to the YouTube API to collect comments and responses for each video separately. We retrieved the text of each comment, its date, the link to the video the comment belongs to, the author's name, and the number of likes and replies a comment had received. Out of privacy considerations, only the text of the comment, its creation date, and the links to the videos will be provided in open access.[2]

2.2. Corpus description

The corpus we created contains a total number of roughly 383,000 unique comments. The number of comments and user activity, however, is unequally distributed between the videos. User engagement in the form of views, likes, and comments can be influenced by the following factors:

- pronouns: use of 'you' and 'they' decreases engagement;
- language style: subjective language style increases engagement;
- verbs: informativity, activity, and temporality decrease engagement;
- content: argumentativeness and informativity decrease engagement;
- emotionality: moderate or low-arousal emotion levels increase engagement;
- time of publication: non-business hours and weekdays increase engagement;
- length of the video: medium-length and long videos increase engagement. (Munaro et al. 2021)

[1] For the full list, see the appendix.
[2] For the full corpus, visit https://www.kaggle.com/datasets/lidiiamelnyk/youtube-comments-on-gender-diversity.

The following chart illustrates the distribution of the comments between the videos. As can be seen, for most of the videos, the number of comments ranges from 0 to 2,000, and only a couple of videos received more than 2,000 comments.

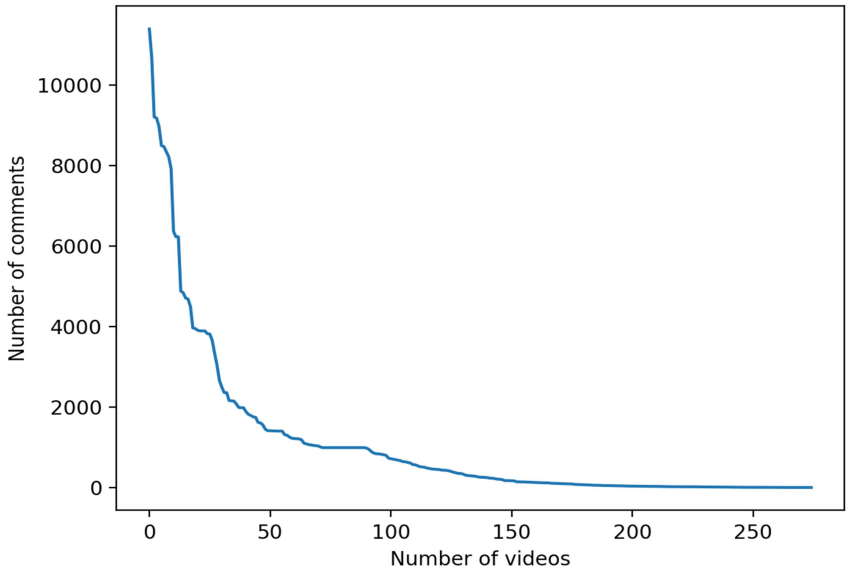

FIGURE 2. COMMENTS DISTRIBUTION

When creating the corpus, we did not set any time frame. The comments obtained were created between 2015, which is the date of the first video found on the topic including an active comments section, and the beginning of 2022, which marked the end of the scraping procedure for our project. However, comments are not equally distributed in the corpus time frame, with very little activity spotted up to the middle of 2017. Thus, we decided to exclude comments from before September 2017, deeming these comments to be statistically insignificant.

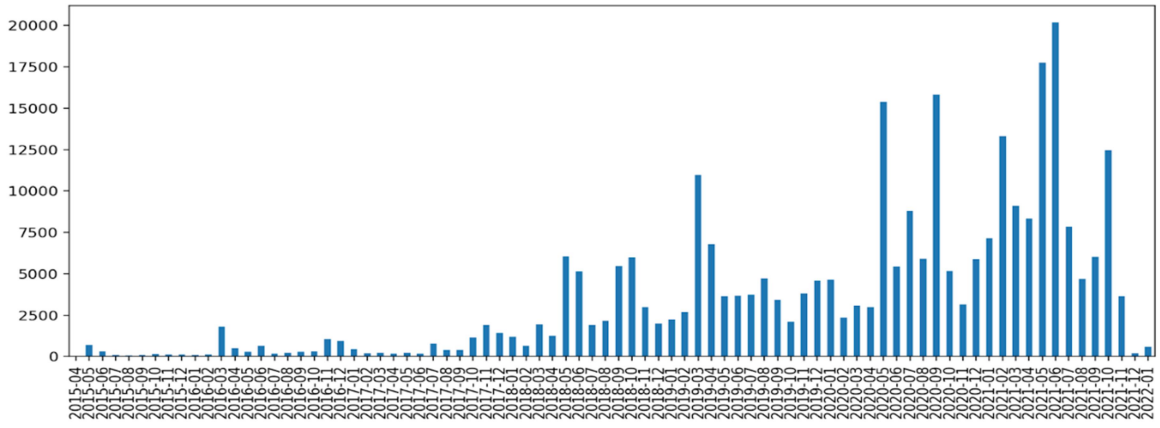

FIGURE 3. TEMPORAL DISTRIBUTION OF THE COMMENTS

Even though the language of the videos themselves is German, non-German comments were

rather frequent. To identify the language of the comments and only keep those in German, we used the LangID tool in Python. The total number of comments we ended up with is 350,000.

3. Sentiment Analysis with BERT

SA "is the computational treatment of opinions, sentiments and subjectivity of text" (Medhat et al. 2014, 1093) and a well-established technology in Natural Language Processing (NLP). It is frequently used for fast and statistically representative analyses of user reviews, for the analysis of repercussions of events on social media (Gonçalves et al. 2014), and for scholarly purposes. Academic research applying SA is mostly focused on analyzing the sentiment of Tweets (Go et al. 2009; Saif et al. 2012; Sarlan et al. 2014) due to the accessibility of Twitter data and the limited maximum length of the Tweets, which makes them easier to process computationally. The wide variety of SA applications has sparked the interest of scientists and software developers, which has resulted in the development of a range of approaches seeking to solve SA's main challenges: its inability to identify sarcasm, difficulties in evaluating the influence of negation, and word sense disambiguation.

Generally, machine learning methods of SA (as opposed to lexicon-based approaches) attempt to evaluate text polarity based on train and test datasets (Birjali et al. 2021). Such methods can be subdivided into supervised, semi-supervised, unsupervised, and reinforcement learning (Yusof et al. 2015). Supervised learning is preferred for tasks with a specific set of classes, while unsupervised learning is used in the opposite case, i.e., when the author does not have any labeled data. Semi-supervised approaches work well with unlabeled data, provided there are some labeled examples available. Reinforcement learning is a machine learning algorithm using trial and error methods to interact with the environment and obtain the maximum cumulative rewards (Birjali et al. 2021). More specifically, within the domain of supervised learning, there are classification algorithms such as SVMs and CRFs, which are rather traditional approaches representing data as engineered features, neural networks such as RNNs, (bi)LSTMs, or CNNs, treating the training data as sequences of vectors, and approaches combining neural networks with the more traditional classification algorithms (Wojatzki et al. 2017).

Since our research especially focuses on identifying the commenters' stance towards the given topic, we decided against collecting annotations for sentiment – for reasons of lacking personnel and temporal capacities as well as annotations being likely to be biased due to previous stance annotations. Having no labeled data ruled out using supervised learning algorithms such as Naïve Bayes, SVM, or neural networks. To overcome this inconvenience, we drew on the findings of Guhr et al. (2020), who (1) created a large publicly available corpus containing more than 5.3 million samples labeled for negative, neutral, and positive sentiment, and (2) trained and tested two different sentiment classification models on this dataset. The corpus comprises the following datasets collected from different sources:

- *PotTS* with 7,504 messages from Twitter (Sidarenka 2016);
- *SB10k* with 9,783 tweets (Cieliebak et al. 2017);
- *GermEval-2017* with 23,525 available documents covering texts related to the "Deutsche Bahn" (Wojatzki et al. 2017);

- *Scare* (sentiment corpus of app reviews) with 800,000 application reviews (Sänger et al. 2016);
- *Filmstarts* dataset with 71,229 user-generated movie reviews from *filmstarts.de*;
- *Holidaycheck* dataset with 3,524,193 text-rating hotel reviews;
- *Leipzig-wikipedia* with 1,000,000 Wikipedia documents (Goldhahn et al. 2012) annotated as neutral;
- *Emotions* dataset with 1,306 emotionally marked and frequently insulting utterances.

The two sentiment classification models Guhr et al. (2020) trained and tested on this corpus were FastText and BERT. BERT, Bidirectional Encoder Representations from Transformers, is a language representation model "designed to pre-train deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers" (Devlin et al. 2019, 4171). It can be fine-tuned by adding just one more output layer, and it can be applied to solving a variety of NLP tasks without task-specific modifications of the model architecture. BERT relies on two core processes: pretraining and fine-tuning. During the pretraining stage, the model is trained on unlabeled data with a variety of tasks, and after initializing the model with the pretrained parameters, these will be fine-tuned with the use of labeled data. The major advantage of BERT is the existence of a wide spectrum of large pretrained models, which minimizes the need for a big amount of training data for the model to provide representative results.

Guhr et al. (2020) used a model pretrained by the developers on the German BERT small model, making use of "bidirectional training of a deep transformer-based network architecture" (Guhr et al. 2020, 1630), and trained their German BERT model for SA using the implementation provided by the HuggingFace repository. While the authors recognize certain advantages of FastText, being a traditional word embedding model, and the model's performance was relatively high for both data from known domains as well as from an unknown domain, the BERT model scored better in all of these settings and in classifying unseen data, reaching an F1 score of 0.80 and outperforming FastText by 9.4%.

We used a Python script to apply the pretrained model to our data. We sought to predict the sentiment of each comment in our dataset separately and to receive a clear result as to whether the polarity is positive, negative, or neutral. Unfortunately, the model suggested is aspect-based, where aspect stands for "strongly correlated categories of an item (e.g. the price of a product)" (Göhring et al. 2021, 213), and one and the same sentence can have several aspects related to one and the same entity. Thus, instead of one aspect for each comment, the results we obtained provide a list of aspects for each comment, and some basic arithmetic calculations were needed to define the leading sentiment of each comment. Since we could not get the weights of each aspect in the comment, the calculation might lead to unreliable and slightly distorted results at the end.

We applied the SA model to the same dataset used for annotating the comments according to their stance (see Sections 4 to 5). The results of our SA are displayed in the figure below. It can be seen that the largest part of the annotated data classifies as having a negative sentiment.

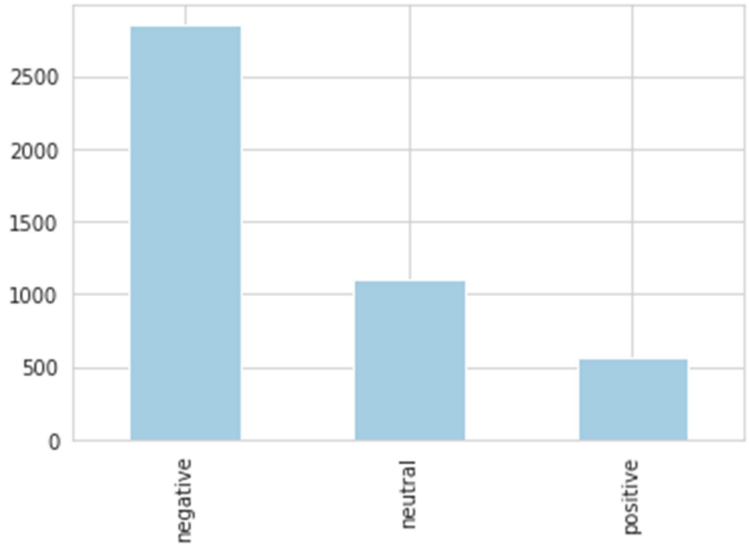

FIGURE 4. SENTIMENT DISTRIBUTION

Guhr et al. (2020) warn that for unbalanced data, the accuracy of negative sentiment is lower. The probability of a different accuracy for the sentiment categories should be taken into consideration for further research on SA.

4. Stance annotation

While the sentiment of the comments does play an important role in analyzing the discourse related to gender diversity, the main focus of our research was SD. Following Biber and Finegan (1988) and Du Bois (2007) in their works on stance and stance-taking, we understand stance as a person's subjective position and attitude towards a specific subject, and determining a comment's stance "focuses on identifying a person's standpoint or view toward an object of evaluation, either to be in favor of (supporting) or against (opposing) the topic" (ALDayel and Magdy 2021, 4). SD is an NLP task used to identify a text's attitude towards a specific topic (Augenstein et al. 2016), and ALDayel and Magdy (2021) define "[s]tance detection on social media [as] an emerging opinion mining paradigm for various social and political applications in which sentiment analysis may be sub-optimal" (ALDayel and Magdy 2021, 1). While the focus of SA is on analyzing whether the sentiment, or overall tonality, of the comment is positive, negative, or neutral, SD seeks to identify the author's favorability towards a certain target, which, moreover, does not necessarily have to be mentioned in text. The task of SD is complicated even more by the fact that the author of the text might express a positive opinion about the target, for example, but is doing so by quoting or restating facts representing a negative opinion (Krejzl 2017).

Deploying a threefold classification along a continuum, comments can either be completely in favor of (FAVOR) or totally against (AGAINST) gender diversity, or they might not have a clear stance towards either direction or be positioned somewhere in between the two poles, in which case they are considered to be neutral (NEUTRAL). More specifically, to be completely in favor of gender diversity means to recognize all genders and identities and to express one's support

of measures taken in order to improve the situation of gender (and sexual) minorities. To be totally against gender diversity, as the other end of the continuum, means to deny the existence of more than two genders (and sexes) and to oppose any societal or institutional developments acknowledging gender (and sexual) minorities. To be classified as FAVOR or AGAINST, 1) the opinion of the comment's author has to be either explicitly stated or easily inferable from the comment's semantics and pragmatics, and 2) it must be possible to understand from the comment's context alone that the opinion expressed is relevant to the topic of gender diversity. Comments that either do not clearly express the author's position or restate the facts without any personal evaluation as well as all the comments that lack context and relevance to our topic are to be annotated as NEUTRAL.

The following examples (taken from the corpus) illustrate the three stances:

FAVOR:

(1) Ja vor allem weil man ja sowohl biologisch als auch von der Identität her Intersexuell sein kann und es medizinische Situationen gibt in denen es wichtig sein kann das zu wissen

(Yes, especially since you can be both biologically intersexual or identifying as such, and there are medical situations in which this might be important to know.)

(2) Für mich wäre es kein Problem, nicht-binäre Menschen so anzusprechen, wie sie sich respektiert fühlen. Allerdings wäre es deutlich leichter, wenn wir sprachlich eine Alternative zu Herr/Frau, er/sie, sein/ihr usw. hätten. Es klingt in meinen Ohren sehr holprig, immer Vor- und Zunamen zu nennen. Gibt es da bereits Alternativen?

(It wouldn't be a problem for me to address non-binary people in a way they feel respected. However, it would be much easier if we had a linguistic alternative to Mr./Mrs., he/she, his/hers and so on. It sounds rather bumpy to my ears to use both the given and the family name every time. Are there any alternatives yet?)

In (1), the author recognizes both biological and non-biological aspects of intersexuality, and the relevance of the comment to the topic (gender diversity) is also clearly identifiable. In 1b, the person criticizes the limited vocabulary of the German language with respect to pronouns and different forms of address, which makes the comment incline towards negative sentiment. Nevertheless, the person expresses their wish to address non-binary people the way they want to be addressed and feel respected. Therefore, the stance of the comment is 'in favor'.

AGAINST:

(3) No hate: Man ist ein biologischer Mann wenn man einen Penis hat ... was nicht heißt dass man nicht so leben kann wie man möchte: Mode, Makeup, Nickname...😅😅

(No hate: You're a man biologically if you have a penis … which doesn't mean that you can't live the way you want to: fashion, makeup, nickname … 😅😅)

(4) Ja, wir werden immer diskriminiert. Bei den TRANSport Helikoptern ist es am schlimmsten.

(Yes, we're always discriminated against. It's worst for the TRANSport helicopters.)

In (3), the author denies the existence of diverse genders and sexes (compliant with the external sex organs) and mitigates the challenges gender diverse people have to face on a regular basis, reducing their concerns to external expressions of gender adherence (fashion, makeup, and names). It is clear that the commenter's stance towards the topic is negative and they oppose gender diversity. In (4), the author makes a sarcastic joke about being trans* and ridicules the discriminatory experiences trans* people have, which clearly shows the commenter's stance against gender diversity.

NEUTRAL:

(5) @Schwarz und so ...wäre aber korrekt!
(@Schwarz and so on …would be correct though!)

(6) Bei dem Wort binary wird eher die erste Silbe betont :)
(The stress is rather on the first syllable of the word binary :))

(7) Weil ich unter dieser ÖR Produktion einer aufklärenden Maßnahme sachliche Kritik äußern will.
(Because I want to voice objective criticism to this ÖR production of enlightening measures.)

In (5), we cannot clearly identify which opinion the author supports due to missing context. In (6), there is no indication of an opinion or a personal evaluation, but rather an explanation is given. Even though the author mentions they want to express their criticism to the producer of the video in (7), we lack too much context to say whether their position is in support of or against gender diversity.

Additionally, detecting the comment's stance can be enhanced by paying attention to linguistic indicators of opinion statements, such as:

- first person pronouns with verbs indicating (dis)approval (Göhring et al. 2021, 2): *ich + zustimmen, ablehnen, unterstützen, widersprechen, einverstanden sein, dagegen sein*;
- predicative statements (ibid.): *das ist Blödsinn*;
- modal constructions (ibid.) used to propose an action/idea: *der Paragraph sollte abgeschafft werden, das sollte egal sein*;
- pragmatic opinion markers: *ich glaube, ich denke, ich bin der Meinung*, and intensifying or mitigating adjectives: *lächerlich, unvorstellbar, absolut*.

We randomly generated an annotation dataset from the corpus. The annotation process involved two separate steps. In a first step, the authors annotated 4,811 comments in total, following guidelines that were put together beforehand and regularly discussing cases that were unclear. In order to objectify the annotations, in a second step, the same sample of comments was given to two student annotators, along with the guidelines including additional examples and clear instructions. To discuss the progress of the annotations and possible disagreements regarding the annotation guidelines as well as to ultimately improve the inter-annotator agreement, three meetings were held: after 50, 200, and another 500 comments. In total, the students annotated 1,300 comments and an inter-annotator agreement, calculated as Cohen's Kappa score (Landis & Koch 1977), of 0.668 was reached, which can be considered substantial agreement.

The number of comments annotated by the authors for training and testing purposes amounts to a total of 4,811. The following figure shows the distribution of the different stances.

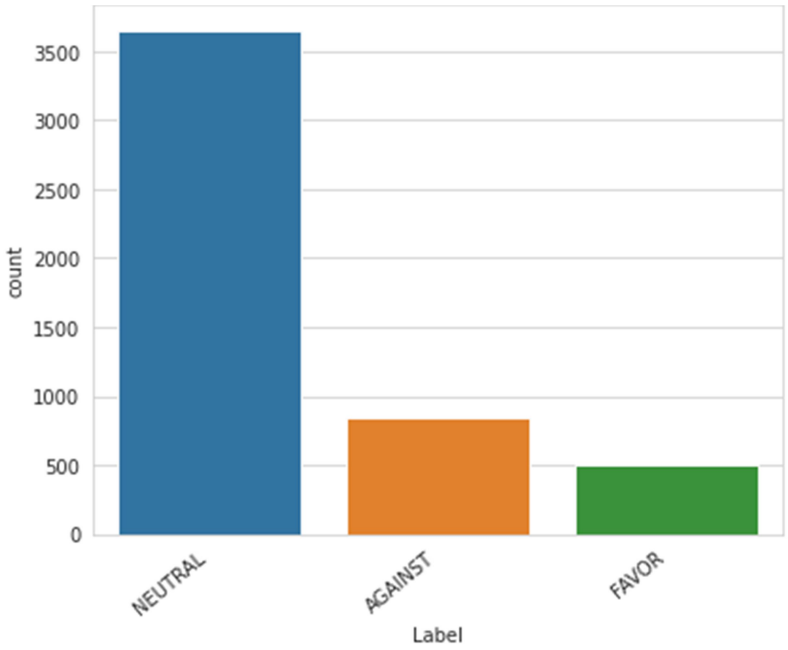

FIGURE 5. STANCE DISTRIBUTION (ANNOTATIONS)

As can be seen, neutral comments represent the largest part of the entire annotation dataset, while comments annotated as against were detected roughly 1,000 times, constituting almost 20% of the dataset, and comments with a stance in favor of gender diversity are underrepresented with only a bit more than 500 hits accounting for 10-11% of the dataset.

5. Stance Detection

Since the extreme imbalance of the annotated dataset can be problematic from the point of view of machine learning, which is why we tested several settings of the dataset. First, we intuitively kept the dataset unbalanced, assuming that the distribution of positively and negatively connotated comments would be less visible in the entire corpus and that keeping the dataset unbalanced is thus representative of the corpus structure itself. However, since an unbalanced dataset can lead to relatively poor model performance due to the extreme difference between the classes, we applied three different techniques to see if they significantly improve the models' performances. As a first method, we introduced a class weighting parameter to the models, which makes it possible to weight the contribution of each particular sample towards the overall loss of the models. As a second and third method we experimented with under- and oversampling techniques. (cf. He and Garcia 2009) For undersampling, we reduced the sizes of the two bigger classes to the size of the smallest one (i.e., 'in favor'), randomly selecting the examples to include. For oversampling we increased the sizes of the underrepresented classes to match the size of the biggest class by duplicating random data samples within the smaller classes.

5.1. SVM with TF-IDF

Support Vector Machines (SVM) have a proven record in the tasks of text classification. An SVM is a supervised machine learning algorithm commonly used for classification, regression, and outlier detection problems. Based on a simple mathematical model ($y = wx' + \gamma$), the algorithm either linearly separates the data into classes within their original domain or it transforms the data to a higher dimension, the so-called feature space, plotting each data point according to its features, so that the data can then be divided linearly. For mapping the data to a feature space, the algorithm utilizes a kernel function, turning the linear equation into a nonlinear one ($y = w\varphi(x') + \gamma$) by including parametrization and optimization objectives. (Suthaharan 2016; Taher et al. 2018)

In our case, there are three categories or labels. The SVM defines the gaps to separate between these categories and maps the examples of each class (favor, against, and neutral) within these gaps. We vectorized the comments in the train and test sets with TF-IDF and applied the SVM scikit-learn classifier to the data. We tested the performance of all main kernel functions of the SVM: linear, polynomial, radial basis (RBF), and sigmoid. The figure below shows the accuracy of the SVM using the different kernel functions.

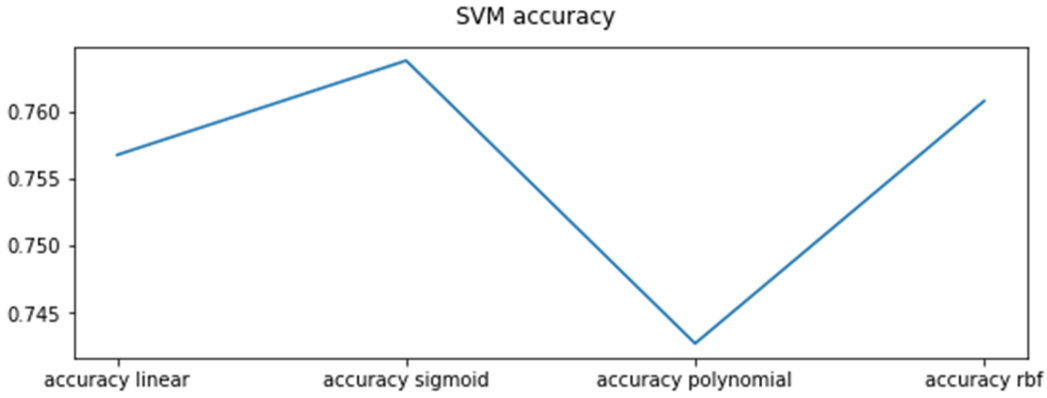

FIGURE 6. SVM ACCURACY USING DIFFERENT KERNEL FUNCTIONS (BEFORE OVERSAMPLING)

As can be seen, the accuracy varies between 74% for the polynomial function and 76.4% for the sigmoid function. The following tables outline the different scores for each kernel function.

	in favor	(oversampled)	**against**	(oversampled)	**neutral**	(oversampled)
Precision	0.666	0.96	0.615	0.96	0.768	0.96
Recall	0.108	0.99	0.142	0.97	0.983	0.92
F1 score	0.186	0.97	0.230	0.96	0.862	0.94

TABLE 1. SVM WITH RBF (BEFORE AND AFTER OVERSAMPLING)

The radial basis function ($k(x_i,x_j) = exp(-\gamma||x_i - x_j||^2)$) displayed relatively high precision for all of

the three classes; however, it displayed an unsatisfactory recall performance of about 10% for the positive stance and around 14% recall for the negative stance.

	in favor	(oversampled)	**against**	(oversampled)	**neutral**	(oversampled)
Precision	0.5	0.99	0.47	0.99	0.751	0.97
Recall	0.07	0.99	0.04	0.98	0.985	0.98
F1 score	0.132	0.99	0.08	0.98	0.8522	0.98

TABLE 2. SVM WITH POLYNOMIAL FUNCTION (BEFORE AND AFTER OVERSAMPLING)

Compared to the RBF, the polynomial function ($k(x_i,x_j) = (1 + x_i * x_j^d)$) displayed lower precision for all of the classes and lower recall for the 'in favor' and 'against' class, and can thus not be considered a good fit for our task.

	in favor	(oversampled)	**against**	(oversampled)	**neutral**	(oversampled)
Precision	0.571	0.85	0.548	0.83	0.789	0.82
Recall	0.173	0.90	0.266	0.83	0.951	0.76
F1 score	0.266	0.87	0.358	0.83	0.862	0.79

TABLE 3. SVM WITH SIGMOID FUNCTION (BEFORE AND AFTER OVERSAMPLING)

Precision using the sigmoid function ($k(x_i,x_j) = \tanh(\alpha x^T y + c)$) appeared to be lower than using the RBF one; however, the recall of the 'in favor' and 'against' classes was higher, resulting in higher F1 scores respectively.

	in favor	(oversampled)	**against**	(oversampled)	**neutral**	(oversampled)
Precision	0.403	0.94	0.518	0.91	0.824	0.86
Recall	0.271	0.98	0.414	0.95	0.895	0.78
F1 score	0.324	0.96	0.46	0.93	0.858	0.80

TABLE 4. SVM WITH LINEAR FUNCTION (BEFORE AND AFTER OVERSAMPLING)

The linear function ($k(x_i,x_j) = x_i * x_j$) – though displaying relatively low results regarding precision of the 'in favor' class – displays the highest recall and, respectively, the highest F1 scores for all of the three labels.

Thus, without applying any sampling techniques, the linear kernel appears to be the preferable option for using an SVM for SD. Introducing the class weighting parameter as well as normalizing the dataset by undersampling have proven to be ineffective techniques for improving the model's performance (no matter the kernel function). In contrast, oversampling had a more beneficial effect. The following figure shows the model's accuracy using the different kernel functions and applied to the oversampled dataset.

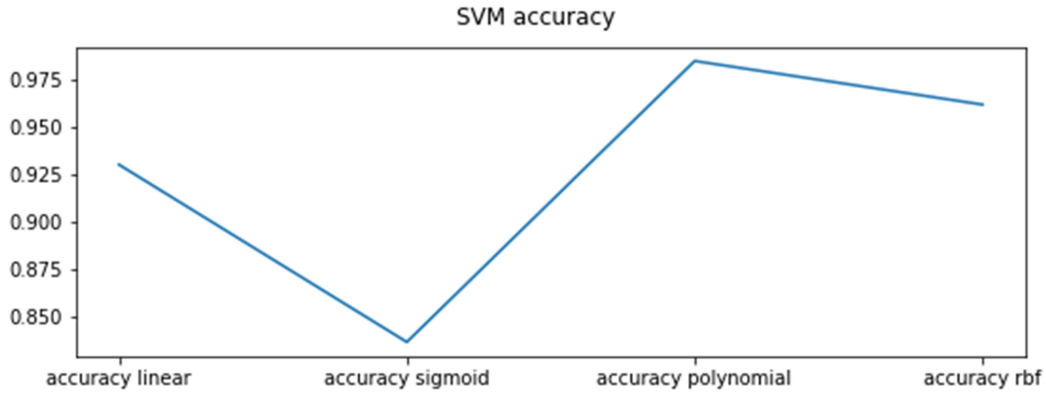

FIGURE 7. SVM ACCURACY USING DIFFERENT KERNEL FUNCTIONS (AFTER OVERSAMPLING)

However, while oversampling increased the model's accuracy to 97.5% for polynomial kernel, over 95% for rbf kernel, 83% for sigmoid kernel and over 92.5% for linear kernel, as can be seen in Tables 1 to 4, it might also result in overfitting the model, as, for example, in case of the SVM with an RBF (Table 1) and the SVM with a polynomial function (Table 2). Nevertheless, in each of the cases it helped to increase the recall of all classes.

5.2. Deep neural networks

Neural networks are deep learning algorithms that are frequently used for classification tasks. We tested three different types of deep neural networks: BERT, an LSTM, and a CNN.

5.2.1 BERT

Since BERT is a state-of-the-art language model and proved to be very helpful for the sentiment classification task (see Section 3), we decided to apply a BERT-based classification model from the Simple Transformers library for the SD task working with our own annotated dataset. We chose the German cased DistilBERT model and fine-tuned it with our training dataset. DistilBERT is a distilled (i.e., compressed) and pre-trained version of BERT, being smaller and faster but retaining 97% of BERT's original performance, i.e., its language understanding capacities. DistilBERT has the same general architecture as BERT, but it has a lower number of layers, and its operations "are highly optimized in modern linear algebra frameworks" (Sanh et al. 2020, 2). "As such, DistilBERT is distilled on very large batches leveraging gradient accumulation (up to 4K examples per batch) using dynamic masking and without the next sentence prediction objective." (Sanh et al. 2020, 3)

We fine-tuned the DistilBERT on our training set with 20% of our annotated data put aside for testing purposes. The model displayed worse performance results than the other neural networks (see Sections 5.2.2. and 5.2.3.) with an accuracy of 60% and a training loss of 1.5. One reason for this relatively bad performance could be the small size of our dataset, since BERT requires rather large datasets to work efficiently. (cf. Ezen-Can 2020) For the same reason, normalizing the dataset by introducing the weighting parameter, underfitting, or overfitting had no significant effect on the performance of the model.

5.2.2 Long Short-Term Memory (LSTM)

Like most neural networks, Recurrent Neural Networks (RNNs) use supervised learning. They represent a particular type of artificial neural networks that uses sequential data to solve temporal or ordinal problems. Fed with training data, RNNs use their 'memory' to "take information from prior inputs to influence the current input and output" (IBM a). Whereas traditional deep neural networks work under the assumption of inputs and outputs to be independent from each other, the RNN's output is generated dependent on the previous elements of the sequences processed. Calculating errors from its output and input layers and leveraging a backpropagation through time (BPTT) algorithm, the model trains itself. However, this process, by which the gradients are determined, can lead to exploding or vanishing gradients. To solve this problem, Hochreiter and Schmidhuber (1997) introduced the Long Short-Term Memory (LSTM) network, which contains hidden layers with three gates (input, output, forget) to control the flow of information needed to predict the output. While the simple RNN (S-RNN) is not capable of retaining information over longer distances, the LSTM is able to retain information from previous states to influence the current predictions, that is, they can remember and process long-term dependencies. (IBM a) As such, LSTMs can be defined as "gated RNNs that can store and forget memory from previous iterations" (Chopra et al. 2017, 4). They have since been widely applied for sequence prediction problems (Brownlee 2017).

Indeed, LSTMs have a proven record of being effectively used for SD. Nevertheless, the majority of studies on LSTMs focused on English texts and used a corpus of tweets as training dataset. Tweets, due to their limited number of symbols, have a more structured and concise phrasing than can be expected from YouTube comments, which could impact the accuracy of the model.

We set up a sequential network model (LSTM) consisting of (1) the input and (2) embedding layers with a maximum number of 200 for words per comment and 128 dimensions of the dense embedding (numbers calculated through testing), (3) a spatial dropout one-dimensional layer set to 0.001, (4) the LSTM layer with 10 units and recurrent dropout set to 0.001, and (5) the final output layer with our 3 output units and the activation function set to default (softmax). The number of training epochs was identified with an early stopping mechanism and eventually set to 10, with a learning rate of 0.001. We trained the model on our data, measured the loss with the categorical crossentropy loss function, and optimized the model using the Adam optimization algorithm (Kingma & Ba 2015). Moreover, a weighting parameter was added to help balance out the dataset. The figure below outlines the setup of the model.

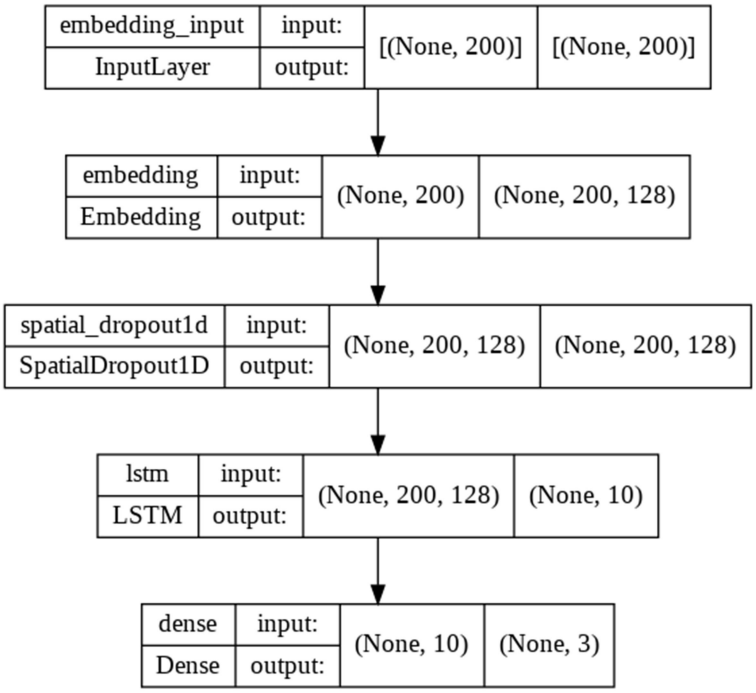

FIGURE 8. LSTM STRUCTURE

The accuracy of the model displayed an increasing tendency throughout the training epochs. It started out with 40% and reached almost 96% at the end of the training, which indicates that the model was overtraining by that time due to the limited size of the training dataset. There is only slight improvement of the validation accuracy (20% of the total dataset) throughout the training, but by the end of the training, it reached 79%, which is significantly higher than in similar research (Dey, Shrivastava & Kaushik 2018). The testing accuracy (on a test set consisting of 570 comments) reached 80%. In accordance with the accuracy, there is a steady decrease in loss when the model is making predictions on the training data. However, the model is still not optimal for minimizing loss when testing it on unseen data: Even though validation loss slightly decreases in the middle of the training, it increases as the training continues.

The application of the weighting parameter as well as undersampling proved to have no positive effect on the model's performance showing a decrease in accuracy to 50% when using the undersampled dataset. Surprisingly, oversampling, too, had a negative effect on the model's performance, with the accuracy decreasing to 60%. The following table shows the categorical precision and recall achieved by the model before and after oversampling.

	in favor	(oversampled)	**against**	(oversampled)	**neutral**	(oversampled)
Precision	0.0	0.92	0.38	0.48	0.8	1
Recall	0.0	0.95	0.34	0.98	0.92	0.0
F1 score	0.0	0.96	0.36	0.64	0.85	0.01

TABLE 5. CLASSIFICATION REPORT LSTM (BEFORE AND AFTER OVERSAMPLING)

Even though we expected the 'in favor' class to score low in the classification report due to the relatively small number of occurrences of the stance, the model's poor performance in the recognition of this class is surprising, considering the model's high accuracy. None of the instances of the class were detected, which results in zero precision, recall, and F1 score. As can also be seen in the table above, oversampling resulted in a total decrease of performance for the neutral class, with 0.0 recall, while the scores improved for the less represented classes.

The model's weaknesses can be ascribed to both the overall small size of the dataset as well as its imbalance. Given its overall high accuracy on the training set as well as its good credit in the academic literature, though, it can be concluded that the LSTM model is still a noteworthy method for SD. However, lower validation accuracy and increasing validation loss should be taken into consideration when deciding whether to use an LSTM for the given task or not.

5.2.3. Convolutional Neural Network (CNN)

In contrast to RNNs (and thus, the LSTM), Convolutional Neural Networks (CNNs) are feedforward neural networks, i.e., they do not feed results back into the network like RNNs. While the number of layers in CNNs can vary, in principle, the first layer is a convolutional layer, which works with a particular input (of a specific number of dimensions), checked by a filter (or 'kernel') for the presence of features, and outputs a feature map. The output size does not have to match the input size, making the convolutional layers partially connected, and it is affected by three parameters that need to be set before training the network: the number of filters, the stride, and padding. To add nonlinearity to the model, the feature map is transformed by a ReLU function. This initial convolutional layer can be followed by other convolutional layers (making the CNN's structure a hierarchical one), or it can be followed by pooling layers, whose function is to reduce dimensionality (and thereby also reduce complexity but improve efficiency). The last layer of a CNN is the fully-connected layer, using the features extracted through the convolutional and pooling layers to eventually perform the classification task, i.e., classifying the input with a probability between 0 and 1 (utilizing a softmax activation function). Because "[c]onvolutional neural networks are distinguished from other neural networks by their superior performance with image, speech, or audio signal inputs," (IBM b) they have primarily been used for classification tasks (especially image recognition and processing) and in the field of computer vision. However, their range of application has been growing in the field of NLP too over the past decade (Kim 2014; Goldberg 2016; Poddar et al. 2018).

We applied a one-dimensional CNN for the SD task. The model consisted of (1) the input and (2) embedding layers with 250 dimensions and a maximal length of 250 words per comment, (3) the CNN layer activated with a ReLu function, (4) a Global Max Pooling 1D layer to downsample the input presentation and (5) another Flatten layer to further remove dimensions, (6) a dropout layer set to 0.05, and (7) the final output layer with our 3 output units and the activation function set to default (softmax). The model was trained on 20 epochs within 16 batches consisting of 250 comments each. Loss was calculated with the categorical crossentropy loss function, and the model was optimized using the Adam optimization algorithm and a learning rate of 0.001. Similar to the LSTM, a class weighting parameter was introduced to

balance out the dataset. The architecture of our model is shown in the following figure.

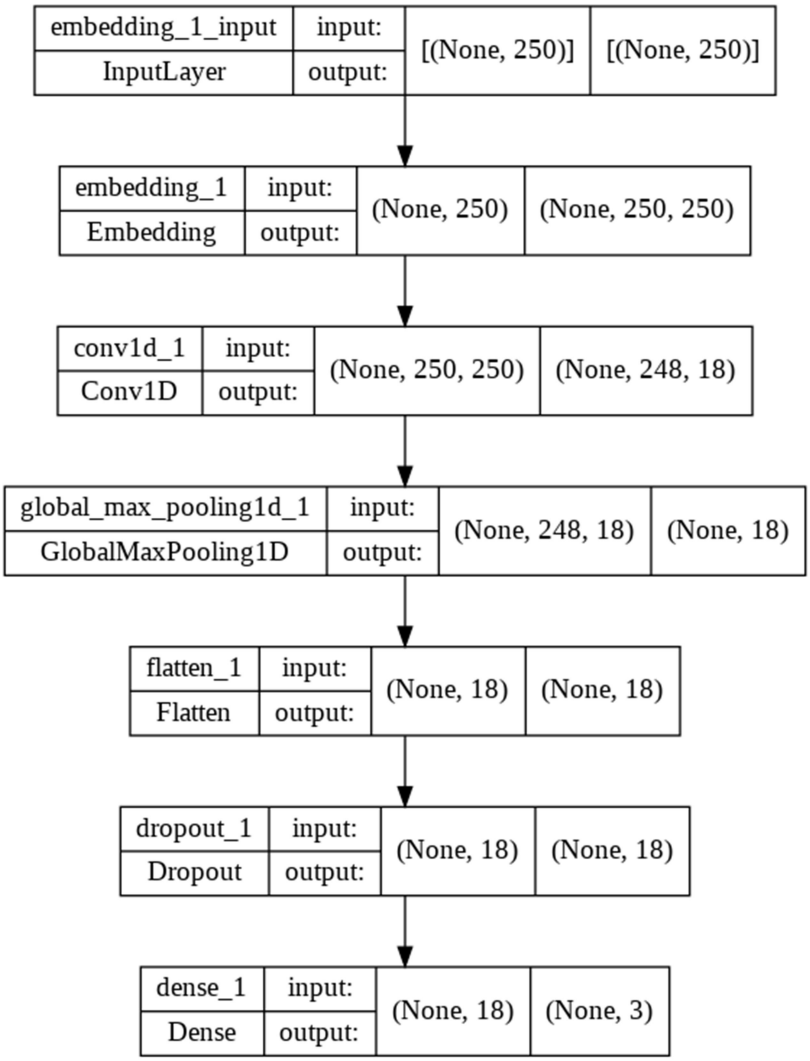

FIGURE 9. CNN MODEL STRUCTURE

The accuracy of the model at the beginning of the training was estimated to be 72% for the validation and 65% for the training data. By the tenth epoch, the model reached 98% accuracy with the training data and 76% accuracy with the validation data. Testing accuracy reached 78.3%. The training loss gradually increased from 0.97 to 0.18 by the end of the training, accompanied by the gradual increase of the validation loss from 0.83 to 0.65. More training and validation data might further reduce the loss.

As with the other models, applying the weighting parameter and undersampling had rather negative effects on the model. Oversampling, however, led to a significant increase of the model's performance: Training accuracy reached 98%, validation accuracy 94%, and testing accuracy 92%, and the validation loss steadily decreased from 0.93 to 0.15 resulting in a testing loss of 0.5. The following table outlines the model's categorical precision and recall before and after oversampling.

	in favor	(oversampled)	**against**	(oversampled)	**neutral**	(oversampled)
Precision	0.57	0.94	0.59	0.90	0.78	0.93
Recall	0.37	0.98	0.46	0.95	0.96	0.84
F1 score	0.45	0.96	0.52	0.92	0.86	0.88

TABLE 6. CLASSIFICATION REPORT CNN (BEFORE AND AFTER OVERSAMPLING)

The CNN displays a high recall and F1 score for the neutral class, which, given the previously provided classification report from the SVM, is almost always correctly detected. It is also capable of providing correct results for 59% of the negatively labeled comments and identifying 46% of them. Similar to the SVM using a sigmoid or linear kernel function, the 'in favor' stance class is the one the model has the smallest number of correct hits with. Even though, after oversampling, the largest class (i.e., neutral stance), shows a decreased recall and F1 score, the scores for the two underrepresented classes, and, in particular, the 'in favor' class, have significantly improved.

5.2.4. Interim Summary

Comparing the performances of the neural networks, it can be noted that in contrast to BERT, the LSTM and the CNN represent two valuable models for detecting stance on a relatively small and unbalanced dataset. The small size of the 'in favor' class negatively affected both models' performance in detecting this particular class, particularly when applied to the unbalanced data set, and, as expected before balancing out the dataset, the class which was best detected was the largest one, i.e., the neutral stance. Interestingly, even though the LSTM's accuracy was better than the CNN's (before oversampling) and CNNs do not have that much academic record of successful use for multiclass text classification (being more widely applied with image recognition tasks), the CNN was able to detect all of the classes in the non-balanced annotated dataset, while LSTM failed to detect the least represented class, i.e., the 'in favor' stance. Moreover, the LSTM's fairly high accuracy dropped down after oversampling the dataset, while the CNN's accuracy significantly increased, outperforming the LSTM by more than 30%. Finally, while introducing the weighting parameter and undersampling the dataset did not lead to any favorable changes in all of the models' performances, oversampling turned out to be a much more promising technique to overcome the machine learning difficulties posed by the strong imbalance of the dataset. Therefore, in order to improve the multiclass classification results, it is recommended to 1) chose oversampling over undersampling techniques and 2) feed the models with a larger dataset.

6. CNN Stance Detection vs. BERT Sentiment Analysis

As previously mentioned, there is a huge difference between the methods of SD and SA, with SA being more focused on the voice of the sentence, whereas SD seeks to identify the opinion of the author towards a specific target, which is why we were interested to see whether the two (sentiment and stance) correlate in a particular way. Thus, we decided to compare the

classification output of the SA with BERT with the output of the SD. The correlation was calculated with a Cohen's Kappa score, a statistical coefficient used to measure inter-rater reliability of the categorical results. The SD labels 'in favor' and 'against' were replaced by 'positive' and 'negative', respectively, to match the labels of the SA.

The dataset used for both classifications is the test set from the previous SD task, which consists of 999 comments, and the model we chose to work with is the CNN, since it produced the best results. Because oversampling significantly impacted the output of the model, we used both results of the model (results of CNN trained on unbalanced data and results of CNN trained on oversampled data) to calculate the correlations between stance and sentiment.

6.1. Categorical correlation before oversampling

The CNN trained on the unbalanced data was able to identify 865 comments with a neutral stance, 107 with a negative one, and 27 with a positive one. The small number of positive stance comments can be traced back to the similarly small number of positively labeled comments in the training set. Thus, the most frequent category is the neutral stance class (61.7%), which also meets the label's frequency in the training data. In contrast, the most frequent sentiment identified by our BERT model is the negative one (63.4%). Similar to the least represented stance, the least represented sentiment is the positive one (12%). Figure 10 shows the categorical correlation of the comments' stance and sentiment.

Our initial hypothesis was that the predictions of the two models would not coincide, and, in fact, the coreference between the three labels in the SA task and those in the SD task was calculated to be 2.9%, which is close to complete disagreement. The strongest correlation exists between negative stance and negative sentiment (74.8%), whereas the correlation is almost non-existent in the case of negative stance and positive sentiment, which makes sense considering the unlikelihood of a person clearly opposing gender diversity in a clearly positive or joyful way. What is surprising though is that while only 25% of the comments with a neutral stance also possess a neutral sentiment, 61.7% of the neutral stance comments have a negative sentiment, and only some (12.7%) of the comments in which people do not express an explicit stance have a neutral sentiment. Another rather surprising finding is that even the majority (70.4%) of those comments which were identified as expressing a positive stance shows a negative sentiment, and only 3 out of 999 comments are both positive in terms of stance and sentiment.

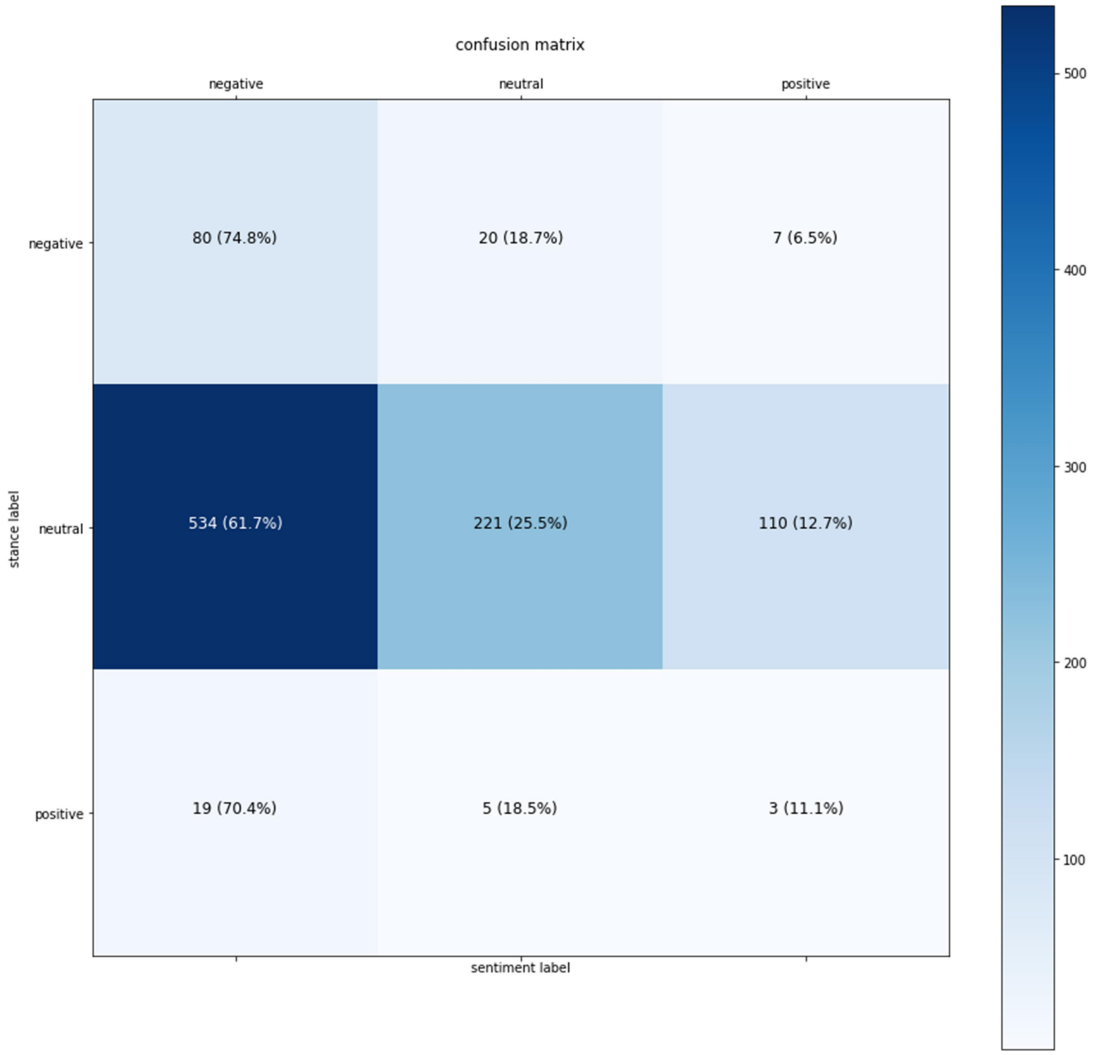

FIGURE 10. STANCE DETECTION/SENTIMENT ANALYSIS CATEGORICAL CORRELATION (BEFORE OVERSAMPLING)

6.2. Categorical correlation after oversampling

Trained on the oversampled training set, the distribution of the different stances identified by the CNN in the test set only slightly changed: while the neutral stance is still the largest class (71.3%), its number of comments decreased by 88, and, consequently, some more comments were identified as having either a negative (17.4%) or a positive (11.3%) stance. The categorical correlation of the comments' stance and sentiment is given in the figure below.

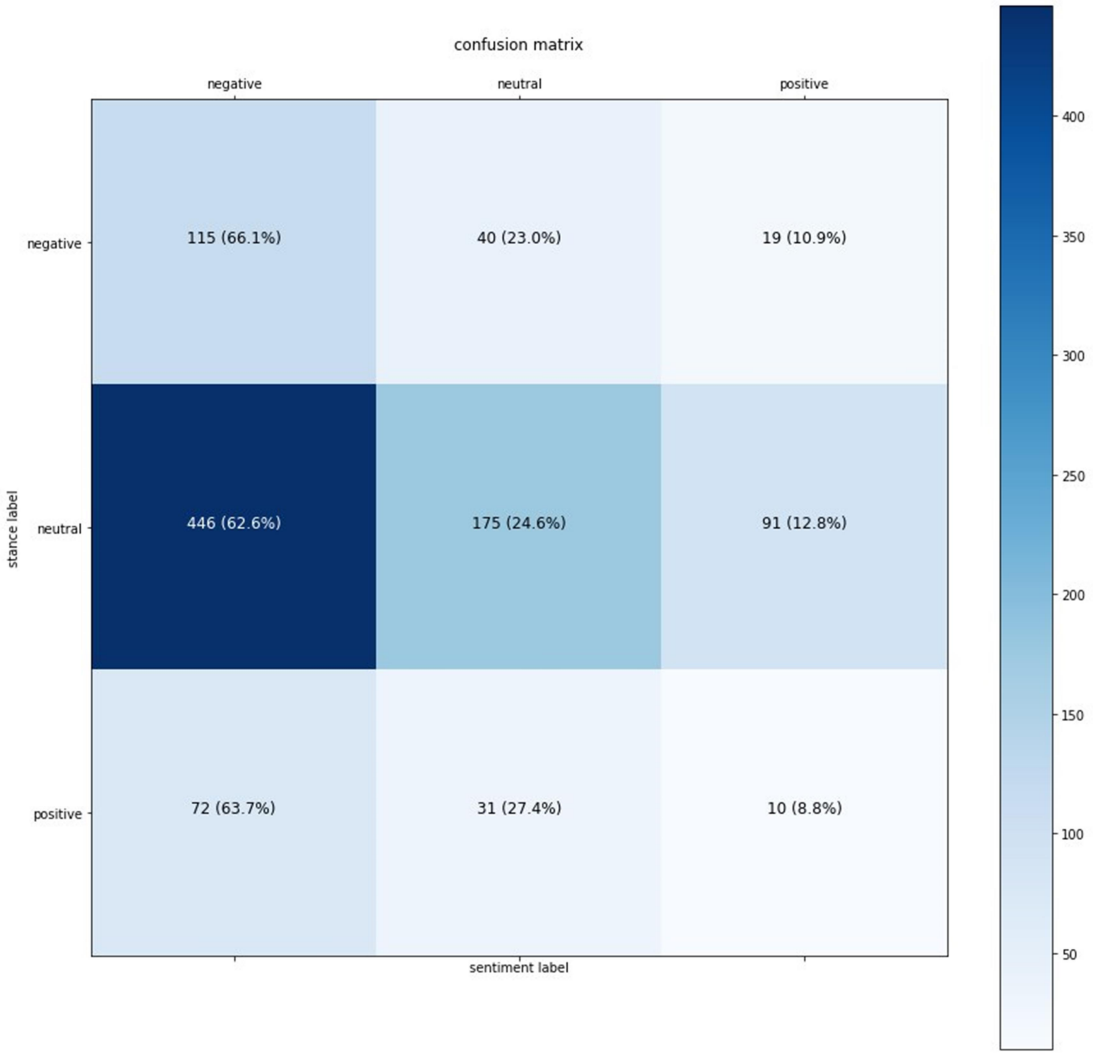

FIGURE 11. STANCE DETECTION/SENTIMENT ANALYSIS CATEGORICAL CORRELATION (AFTER OVERSAMPLING)

7. Conclusions

SD is a challenging yet important task of NLP, which can neither be approximated by nor solved with the help of existing methods for SA. Though both methods belong to the field of text classification, SD specifically seeks to identify the position of the subject towards a particular target, whereas SA aims at obtaining the general sentiment of a comment.

In our research, we demonstrated that there is close to no correlation between the results of SD and SA. The fact that the overall sentiment of the discourse characterizes as negative not only shows that people are emotionally involved and critically, if not aggressively, engage in the discourse, but it also underlines the importance of the topic and the discourse revolving around it, since strong, and particularly negative, emotions are indicative of strong opinions and potential conflicts in society. Moreover, it points to the relevance of the topic itself regarding people's constructs of identity and, more precisely, legitimacy of self-determination. In fact, the small number of comments actually – or rather, completely and unconditionally – in favor of gender diversity implies that ruling over one's own body and identity is by no means

considered a private and thus autonomous matter. Instead, a person's gender, sex, and sexuality seem to constitute a public business, in that people feel personally affected by other people's identities or choices and see their own systems of values and ideas questioned and even endangered, ultimately resulting in people feeling threatened by and thus subtly or loudly opposing people who do not conform to these values. Against the background of current societal as well as legal conservative backlashes, the negative sentiment of the discourse and the lack of support for gender diversity in the majority of comments appears especially alarming. Besides these negative implications of the findings, however, the fact that this discourse is emotionality charged as well as the large number of comments that neither show an 'in favor' stance nor an 'against' stance indicate the high interest in the topic itself and that people are eager to share their views and ultimately also learn about gender diversity and issues that relate to this topic. Beyond that, a classification that takes into account more than just the three stances identified in our research, would undoubtedly draw a much more detailed and precise picture of the opinions expressed in the discourse (cf. Poddar et al. 2018).

Hence, SD on YouTube comments can provide valuable insights into public opinions regarding frequently debated topics. It helps to identify people's stances towards that specific topic and relate them to the specific arguments made and the sentiment that characterizes the comment. The respective findings can help to understand potential fears and concerns and thus be used to share knowledge and promote understanding, enhance respectful and effective communication between those involved in the discourse as well as policy making, and, ultimately, reach more acceptance and improve the life of those being currently oppressed or otherwise negatively affected by cis-hetero-normativity.

Training models on social media data requires a number of specific tasks to be taken into consideration: language identification and filtering of those comments fully or mostly written in a foreign language (i.e., non-German in our case), standardization and error correction as part of data preprocessing as well as test-based identification of the most appropriate embedding type and language model. A wrong embedding type can result in both over- and underfitting, negatively affecting the general performance of the model. Against this background, one of the surprising results of our research is the rather negligible difference in the performance of a quite simple TF-IDF-based SVM and the more advanced technology of a CNN relying on 250-dimensional word embeddings, with which we received only as much as 1% improvement of categorical recall and precision, while the processing time needed for it is significantly higher than that of the SVM.

Moreover, although keeping the dataset unbalanced seemed compelling for reasons of representativity, balancing out the training data through oversampling in order to enlarge the underrepresented classes had a significant impact on the model's performance. However, in some cases it also led to overfitting, which is why detecting class imbalances or overlappings is crucial. To fix the respective imbalances or overlappings, to maximize performance, and to achieve the desired results, different sampling techniques on the level of the data as well as the classifier should be considered. (cf. He and Garcia 2009; also Prati et al. 2004) Standardizing the data derived from social media in a way that it only contains phrases representative of a specific

stance might be another advantageous step forward. After all, the task requires big amounts of annotated data for it to be solved by more complex machine learning methods.

References

ALDayel, Abeer, and Walid Magdy. 2021. "Stance detection on social media: State of the art and trends." *Information Processing and Management* 58: 1–22. doi:10.1016/j.ipm.2021.102597.

Augenstein, Isabelle, Tim Rocktäschel, Andreas Vlachos, and Kalina Bontcheva. 2016. „Stance Detection with Bidirectional Conditional Encoding." In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Austin, Texas, November 01-05. Association for Computational Linguistics. 876–885. doi:10.18653/v1/D16-1084.

Biber, Douglas, and Edward Finegan. 1988. "Adverbial stance types in English." *Discourse Processes* 11(1): 1–34. doi:10.1080/01638538809544689.

Birjali, Marouane, Mohammed Kasri, and Abderrahim Beni-Hssane. 2021. "A comprehensive survey on sentiment analysis: Approaches, challenges and trends." *Knowledge-Based Systems* 226(107134). doi:10.1016/j.knosys.2021.107134.

Brownlee, Jason. 2017. *Long Short-Term Memory Networks With Python: Develop Sequence Prediction Models With Deep Learning. Machine Learning Mastery*. Retrieved from https://machinelearningmastery.com/lstms-with-python/.

Chopra, Sahil, Saachi Jain, and John Merriman Sholar. 2017. "Towards Automatic Identification of Fake News: Headline-Article Stance Detection with LSTM Attention Models." CS224N project report, Stanford University. https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1174/reports/2761028.pdf.

Cieliebak, Mark, Jan Milaln Deriu, Dominic Egger, and Fatih Uzdilli. 2017. *A Twitter Corpus and Benchmark Resources for German Sentiment Analysis.* Association for Computational Linguistics.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Minneapolis, Minnesota, June 02-07. Association for Computational Linguistics. 4171–4186. doi:10.18653/v1/N19-1423.

Dey, Kuntal, Ritvik Shrivastava, and Saroj Kaushik. 2018. "Topical Stance Detection for Twitter: A Two-Phase LSTM Model Using Attention." Paper presented at 40th European Conference on IR Research 2018, Grenoble, France, March 26-29. doi:10.48550/arXiv.1801.03032.

Du Bois, John W. 2007. "The stance triangle." In *Stancetaking in Discourse. Subjectivity, evaluation, interaction*, edited by R. Englebretson, 139–182. Amsterdam/Philadelphia: John Benjamins Publishing Company.

Ezen-Can, Aysu. 2020. "A Comparison of LSTM and BERT for Small Corpus." Available online: https://arxiv.org/abs/2009.05451.

Go, Alec, Richa Bhayani, and Lei Huang. 2009. "Twitter Sentiment Classification using Distant Supervision." CS224N project report, Stanford University. https://www-cs.stanford.edu/people/alecmgo/papers/TwitterDistantSupervision09.pdf.

Göhring, Anne, Manfred Klenner, and Sophia Conrad. 2021. "DeInStance: Creating and Evaluating a German Corpus for Fine-Grained Inferred Stance Detection." In *Proceedings of the 17th Conference on Natural Language Processing*, Düsseldorf, Germany, September 06-09. KONVENS 2021 Organizers. 213–217. http://aclanthology.org/2021.konvens-1.20/.

Goldberg, Yoav. 2016. "A Primer on Neural Network Models for Natural Language Processing." *Journal of Artificial Intelligence Research* 57(1): 35l–420. https://jair.org/index.php/jair/article/download/11030/26198/.

Goldhahn, Dirk, Thomas Eckart, and Uwe Quasthoff. 2012. *Building Large Monolingual Dictionaries at the Leipzig Corpora Collection: From 100 to 200 Languages.* European Language Resources Association (ELRA).

Gonçalves, Pollyanna, Matheus Araújo, Fabrício Benevenuto, and Meeyoung Cha. 2014. "Comparing and Combining Sentiment Analysis Methods." In *Proceedings of the first ACM conference on Online social networks*, Boston, Massachusetts, October 07-08. New York: Association for Computing Machinery. 27–38. doi:10.1145/2512938.2512951.

Guhr, Oliver, Anne-Kathrin Schumann, Frank Bahrmann, and Hans-Joachim Böhme. 2020. "Training a Broad-Coverage German Sentiment Classification Model for Dialog Systems." In *Proceedings of the 12th Conference on Language Resources and Evaluation*, Marseille, France, May 11-16. European Language Resources Association. 1627–1632. http://aclanthology.org/2020.lrec-1.202/.

He, Haibo, and Edwardo A. Garcia. 2009. "Learning from Imbalanced Data." *IEEE Transactions on Knowledge and Data Engineering* 21(9): 1263 l–1284. doi: 10.1109/TKDE.2008.239.

IBM a. "What are Recurrent Neural Networks?" September 14, 2020. https://www.ibm.com/cloud/learn/recurrent-neural-networks.

IBM b. "Convolutional Neural Networks." October 20, 2020. https://www.ibm.com/cloud/learn/convolutional-neural-networks.

Kim, Yoon. 2014. "Convolutional Neural Networks for Sentence Classification." In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, Doha, Qatar, October 25-29. Association for Computational Linguistics. 1746–1751. doi:10.3115/v1/D14-1181.

Kingma, Diederik P., and Jimmy Lei Ba. 2015. "Adam: A method for stochastic optimization." Paper presented at the 3rd International Conference for Learning Representations, San Diego, California, May 07-09. http://arxiv.org/pdf/1412.6980.pdf.

Krejzl, Peter, Barbora Hourová, and Josef Steinberger. 2017. "Stance detection in online discussions." Work-in-progress paper. doi:10.48550/arXiv.1701.00504.

Landis, J. Richard, and Gary G. Koch. 1977. "The Measurement of Observer Agreement for Categorical Data." *Biometrics* 33(1): 159–174. doi:10.2307/2529310.

Medhat, Walaa, Ahmed Hassan, and Hoda Korashy. 2014. "Sentiment analysis algorithms and applications: A survey." *Ain Shams Engineering Journal* 5: 1093–1113. doi:10.1016/j.asej.2014.04.011.

Munaro, Ana Cristina, Renato Hübner Barcelos, Eliane Cristine Francisco Maffezzolli, João Pedro Santos Rodrigues, and Emerson Cabrera Paraiso. 2021. "To engage or not engage? The features of video content on YouTube affecting digital consumer engagement." *Journal of Consumer Behaviour* 20(5): 1336–1352. doi:10.1002/cb.1939.

Poddar, Lahari, Wynne Hsu, Mong Li Lee, and Shruti Subramaniyam. 2018. "Predicting Stances in Twitter Conversations for Detecting Veracity of Rumors: a Neural Approach." In *Proceedings of the 2018 IEEE 30th International Conference on Tools with Artificial Intelligence*, Volos, Greece, November 05-07. The Institute of Electrical and Electronics Engineers, Inc. 65–72. doi:10.1109/ICTAI.2018.00021.

Prati, Ronaldo C., Gustavo E.A.P.A. Batista, and Maria C. Monard. 2004. "Class imbalances *versus* class overlapping: an analysis of a learning system behavior." In *MICAI 2004: Advances in Artificial Intelligence, Third Mexian International Conference on Artificial Intelligence*, Mexico City, Mexico, April 26-30, 2004, 312-321. Berlin/Heidelberg: Springer. doi: 10.1007/978-3-540-24694-7_32.

Saif, Hassan, Yulan He, and Harith Alani. 2012. "Semantic Sentiment Analysis of Twitter." In *The Semantic Web – ISWC 2012. Proceedings, Part I*, Boston, Massachusetts, USA, November 11-15. Berlin/Heidelberg: Springer. 508–524. doi:10.1007/978-3-642-35176-1_32.

Sänger, Mario, Ulf Leser, Steffen Kemmerer, Peter Adolphs, and Roman Klinger. 2016. *SCARE — The Sentiment Corpus of App Reviews with Fine-grained Annotations in German*. European Language Resources Association (ELRA).

Sanh, Victor, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter." Paper presented at *The 5th Workshop on Energy Efficient Machine Learning and Cognitive Computing, Co-located with the 33rd Conference on Neural Information Processing Systems 2019*, Vancouver, British Columbia, December 13. doi:10.48550/arXiv.1910.01108.

Sarlan, Aliza, Chayanit Nadam, and Shuib Basri. 2014. Twitter Sentiment Analysis. In *2014 International Conference on Information Technology and Multimedia*, Putrajaya, Malaysia, November 18-20. IEEE. 212–216. doi:10.1109/icimu.2014.7066632.

Sidarenka, Uladzimir. 2016. *PotTS: The Potsdam Twitter Sentiment Corpus.* European Language Resources Association (ELRA).

Suthaharan, Shan. 2016. *Machine Learning Models and Algorithms for Big Data Classification*. New York: Springer.

Taher, SM Abu, Kazi Afsana Akhter, and K.M. Azharul Hasan. 2018. "N-gram Based Sentiment Mining for Bangla Text Using Support Vector Machine." In *2018 International Conference on Bangla Speech and Language Processing*, Sylhet, Bangladesh, September 21-22. IEEE. 70-75. doi:10.1109/ICBSLP.2018.8554716.

Wojatzki, Michael, Eugen Ruppert, Sarah Holschneider, Torsten Zesch, and Chris Biemann. 2017. "GermEval 2017: Shared Task on Aspect-based Sentiment in Social Media Customer Feedback." In *Proceedings of the GermEval 2017*, Berlin, Germany, September 12. GSCL. 1–12. doi:10.17185/duepublico/72074.

Yusof, Nor Nadiah, Azlinah Mohamed, and Shuzlina Abdul-Rahman. 2015. "Reviewing Classification Approaches in Sentiment Analysis." In *Soft Computing in Data Science. First Interntational Conference 2015. Proceedings*, Putrajaya, Malaysia, September 02-03. Singapore: Springer. 43–53. doi:10.1007/978-981-287-936-3_5.

Appendix

Keywords used for finding suitable YouTube videos:

- 'transgender' *(transgender)*
- 'transsexueller' *(transsexual)*
- 'genderidentität' *(gender identity)*
- 'transgender+Identität' *(transgender identity)*
- 'transgender+Rechte' *(transgender rights)*
- 'gleichberechtigung+transgender' *(equality transgender)*
- 'transsexueller+gesetz' *(transsexual law)*
- 'transgesetz' *(trans law)*
- 'Transsexuellengesetz' *(transsexual law)*
- 'transsexuellengesetz' *(transsexual law)*
- 'TSG+Gesetz' *(TSG law)*
- 'Selbstbestimmungsgesetz' *(law of self-identification)*
- 'transsexuellenrecht' *(transsexual right)*
- 'reform+transsexuellengesetz' *(reform transsexual law)*
- 'transsexuellengesetz+2021' *(transsexual law 2021)*
- 'transsexuellengesetz+2020' *(transsexual law 2020)*
- 'Neuregelung+des+Transsexuellengesetzes' *(revision of the transsexual law)*
- 'trans-gesetz' *(trans law)*
- 'reform+des+transgesetzes' *(reform of the trans law)*
- 'neues+transgesetz' *(new trans law)*
- 'das+Transsexuellengesetz' *(the transsexual law)*
- 'debatte+transsexuellengesetz' *(debate transsexual law)*
- 'trans-recht+gesetz-entwurf' *(trans right law draft)*
- 'entwurf+transgesetz' *(draft trans law)*
- 'Transgeschlechtlichkeit' *(transgenderism)*
- 'trans*Mann' *(trans* man)*
- 'trans*Frau' *(trans* woman)*
- 'Detransition' *(detransition)*
- 'detransition'
- 'Geschlechtangleichung' *(gender alignment)*
- 'nicht+binär' *(non-binary)*
- 'nichtbinär' *(non-binary)*
- 'diversgeschlechtlich' *(diversegender)*
- 'agender' *(agender)*
- 'bigender' *(bigender)*
- 'genderfluid' *(genderfluid)*
- 'genderqueer' *(genderqueer)*
- 'non-binär' *(non-binary)*
- 'Geschlechtsumwandlung' *(gender transformation)*
- 'Geschlechtsumwandlungen' *(gender transformations)*
- 'GA-OP' *(GA-OP)*
- 'geschlechtsangleichende+Operation' *(sex reassignment surgery)*
- 'Metaidoioplastik' *(metaoidioplasty)*
- 'Transition' *(transition)*
- 'Trans*Menschen' *(trans*people)*
- 'Trans*-Solidarität' *(trans* solidarity)*
- 'Intergeschlechtlichkeit' *(intersexuality)*
- 'Intergeschlecht' *(intersex)*
- 'Geschlechtsvielfalt' *(gender diversity)*
- 'Transsexualismus' *(transsexualism)*
- 'Transgenderformen' *(transgender forms)*
- 'Transfrau' *(trans woman)*
- 'Transmann' *(trans man)*
- 'Mann-zu-Frau-Transsexuelle' *(man-to-woman-transsexual)*
- 'Frau-zu-Mann-Transsexuelle' *(woman-to-man-transsexual)*
- 'Heteronormativität' *(heteronormativity)*
- 'Geschlechtsinkongruenz' *(gender incongruity)*
- 'X-gender' *(X-gender)*
- 'drittes+Geschlecht' *(third gender)*
- 'Transgression' *(transgression)*
- 'Retransition' *(retransition)*
- 'retransition'

- 'Phalloplastik' *(phalloplasty)*