Extracting and analyzing emotions in Statements about the Future

Nico Schmidt (ps42begu@studserv.uni-leipzig.de)
Marvin Müller (ja25opir@studserv.uni-leipzig.de)
Klaus Schmidt (ks80wofy@studserv.uni-leipzig.de)

Leipzig University - Faculty of Mathematics & Computer Science
Neues Augusteum, Augustusplatz 10, 04109 Leipzig - Germany

Abstract

Detecting and analyzing people's attitudes and feelings towards the future using tools from Natural Language Processing (NLP) is a considerable endeavor that requires large amounts of labeled data to train models for detecting future statements as well as models for detecting sentiment and emotion. Thanks to the Internet Archive, we received access to an exceedingly large size of web data spanning over the past 10 years. Due to this data being entirely unlabeled, we devised filtering strategies as well as an approach using Active Learning (AL) in order to extract relevant train data and train a model for identifying statements about the future. AL being a semi-supervised learning approach leverages readily available unlabeled data, reducing the need for costly manual annotation. We extracted 200 samples from the Internet Archive data and manually labeled them as either future or not future statement and added 200 additional train samples with the help of Active learning. The resulting model achieved a macro precision of 0.81 with an acceptable precision on future statements of 0.76. In comparison with baseline model before AL was applied, the macro precision increased by 10 percentage points and 12 points on future statements. The model was applied on a dataset containing 20487 potential future statements. Identified future statements were then labeled using a pre-trained model for emotion analysis. Our results show that a majority of future statements are mostly neutral, followed by statements expressing joy. Other emotions were poorly represented according to our findings. Active Learning proved to be a viable tool for enhancing model performance despite small data set sizes.

1 Introduction

The internet is by far the largest collection of information in our society and every day more information is added. In 2020 it was estimated that nearly 4.5 billion people were using the internet, which is more than half of the world's population[1]. Analyzing such vast quantities of data produced by the internet requires computational methods that automate the procedure, allowing the swift processing of data which would be tedious or even impossible to do manually. Due to this vastness, one can assume that all kinds of things are being discussed and written down on the world wide web, including topics about the future. Commonly, statements are part of those topics and many of which also contain sentiment which describes the author's attitudes and opinions towards future themes.

When it comes to data processing, it must first be gathered from somewhere. In our case this step was already done by the Internet Archive, a non-profit organisation which has made its mission to build a freely accessible digital library of Internet websites. By the time of writing this paper, the Internet Archive contains about 625 billion web pages[2], from which we were able to access about 10 years of saved data via a dedicated pipeline, namely the WARC-DL pipeline (Deckers, 2022).

With the advent of Deep learning, notably word embeddings and the Transformer architecture (Vaswani et al., 2017), highly effective tools for Natural Language Processing (NLP) became widely available. Thanks to this and NLP ecosystems such as Huggingface[3], the usefulness of this approach for answering various questions concerning textual data has greatly improved over the last couple of years.

In this work, we want to explore people's attitudes

[1] https://www.britannica.com/technology/Internet
[2] https://archive.org/about/
[3] https://huggingface.co/

towards the future by analyzing the emotional qualities of future statements. For one, we want to find out which kind of emotions are most commonly expressed when people make statements about the future.

2 Related Work

2.1 WARC-DL pipeline

The extraction of website data was mainly performed with the WARC-DL pipeline (Deckers, 2022). This pipeline consists primarily of a CPU cluster and a GPU server. The CPU cluster is able to extract data from WARC files using FastWARC and to pre-filter them by applying a first CPU-based filter. After this processing step, the data is pickled into objects and passed to the GPU server via TCP. On the GPU server, different kinds of Keras models can be applied for classification tasks onto the extracted dataset, before the final results are being saved. An overview of this architecture can be seen in figure 1.

2.2 Active Learning

Active Learning (AL) is a technique to leverage vast quantity of unlabeled data that can be found everywhere. Being a so-called semi-supervised learning method, a pool of labeled data and unlabeled data is used to train the model. Since the acquisition of sufficiently sized labeled datasets is very costly and time-consuming, AL provides the opportunity to significantly trim down on the size of the labeled dataset by leveraging a vastly larger unlabeled dataset that is interactively and selectively labeled by a human annotator (known as the oracle in AL jargon). The AL method is an iterative process in which 1) a model is trained from the labeled pool, 2) samples are selected from the unlabeled pool and labeled by the oracle 3) newly labeled samples are added to the training pool, 4) model is retrained on the new training pool. The advantage of AL lies in the use of query strategies that select particularly difficult samples from the unlabeled dataset which are closely located in the vicinity of the decision boundary (in a classification task) thus labeling such samples should improve the model performance significantly more than random selection and therefore reduce the need to provide excessive amounts of labeled data.
In recent years, great strides were made to apply Transformer based models in the field of Active learning. The Python library *small-text* (Schröder et al., 2021b) provides an easy-to-use framework for performing Active Learning using various kinds of machine learning techniques such as SVMs, Randomforest, Pytorch models implementing RNNs and CNNs as well as transformer based models provided by Huggingface. Using AL and Transformer-based models in the field of text classification tasks provided promising results. (Schröder et al., 2021a) (Schröder et al., 2022)
As the base model for the AL procedure, we decided to use RoBERTa (Liu et al., 2019), a BERT based (Devlin et al., 2018) Transformer model architecture. In terms of the emotion analysis, a pre-trained model (Hartmann, 2022) from Huggingface was used.

3 Determination of classification and possible restrictions

To achieve our goal of extracting and analyzing statements people made about the future a further determination of what should be considered as a statement about the future is needed. The Oxford Dictionary defines a statement as "something that you say or write that gives information or an opinion"[4]. Further a statement is not a question, request, exclamation or command in any form, but a sentence that mostly ends in a full stop[5]. When a given sentence matches these requirements and the object of the statement is related to the future we classify it as such. Statements are not classified as about the future e.g. when the statement is about a period of time that could be in the future or the past. An example from our dataset that we labeled as not a statement about the future is "Subject to the condition that such companies shall divest 26% of their equity in favour of Indian public in 5 years, if these companies are listed in other parts of the world.". Another sentence that at first could look like a positive example, but that we also classified as not a statement about the future is "Latha wants to visit me with her family someday, may be you and Siri can see her too!" which is a request and therefore not a statement. Rules like that, typos as "may be" instead of "maybe" and other noise that couldn't be deleted during the data cleaning could be a problem when increasing the models precision. Furthermore edge cases as statements about the future in direct speech in sentences that

[4] https://www.oxfordlearnersdictionaries.com/us/definition/english/statement_1

[5] https://www.theschoolrun.com/what-statement

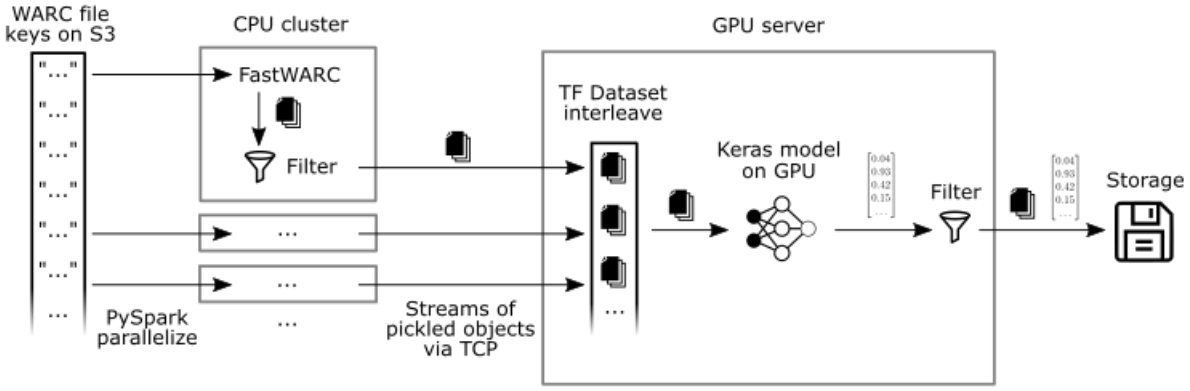

Figure 1: Architecture of WARC-DL pipeline (Deckers, 2022)

end with a question mark and not as most of the statements with a full stop could make that even harder. Mistakes and subjectivity during manual labeling are more error sources.

When it comes to emotion detection we use the model introduced by (Hartmann, 2022) with seven different emotion classes. At this point subjectivity is an even bigger problem as humans agree only about 80% when classifying text into two categories. Therefore humans would still disagree in 20% of the cases even if a model had 100% precision (Roebuck, 2011).

4 Data

4.1 Extraction

The task of data extraction was performed with the WARC-DL pipeline (Deckers, 2022) introduced in section 2. WARC-DL already includes pipelines to perform extraction of text or images from WARC files as well as classification scripts to detect hatespeech or memes in the gathered data. In particular interest for our project was the RegexCounterPipeline which is used for the search of "interesting snippets" via a distributed filter. If one of the snippets matches one of the extracted website's text, it will then be saved by the TextPipeline. The interesting snippets can involve simple words or word groups and more complex regular expressions to match more extended variations of website text. During the extraction process, the data is already pre-processed by Resiliparse[6], which contains tools for parsing and analyzing web archive data among others. When processing a WARC record, first of all the plain text is extracted and passed to a distributed filter. This filter only passes texts in English language with a length of at least 1000 characters and most importantly only when they match one of the specified snippets.

For our purposes we redesigned the RegexCounterPipeline, by introducing a new tokenizer method along with some adjustments to the distributed filter and the export method, which were overwritten from the TextPipeline. The goal of the extraction is to create datasets containing candidate sentences that could possibly include statements about the future. One of these datasets was used to train a model (described in section 4.1) that can automatically classify sentences from other datasets whether they are statements about the future or not. To achieve this, the existing snippets were replaced by own regular expressions.

4.2 Regex Matching

The difficulty for creating own suitable regular expressions was to think of possible words or word groups that are typically when talking about the future. Of course there were multiple approaches to continue with, but we decided to focus on the temporal parts of sentences, because they are relatively simply to match with regular expressions. An example sentence the finished expressions would match is: "In the future, we´ll play complicated exercises.", where the matching part is "In the future". Another example sentence is: "A coaching contract is usually in six monthly sessions.", which will also match at "in six month". Both sentences

[6] https://resiliparse.chatnoir.eu/en/stable/

will be matched by our filter, even though only the first one is a statement about the future, while the second one is a statement with reference to a time span, but not to the future. For our purpose this matching behavior is exactly what we were looking for, because the goal was not to exclusively match future statements, but to increase the chances for this to happen.

4.3 Cleaning

With this adjustments made to the WARC-DL pipeline, we created an adapted pipeline which allowed us to extract single sentences out of the crawled websites and filter them by regular expressions. This again allowed us to extract only specific sets of data, which were then cleaned to be further used for our model.
Before the actual cleaning process starts, the data is already pre-processed by Resiliparse as mentioned in section 4.1. Furthermore the extracted text is passed to a tokenizer, which then removes URLs and splits the text into sentences. This sentences again will be only saved when matching with the specified snippets in form of regular expressions. The data obtained in this way is saved in separate text files, each containing all sentences from a website that match with a regex. Afterwards the data is ready to be cleaned for further use. The cleaning process can be separated into four steps: 1) Read in the extracted sentences, 2) clean the sentences with regexes, 3) remove duplicate sentences and 4) merge everything into one text file. The data is cleaned by applying three filters to it. The first filter is replacing non-breaking spaces (NBSP) by standard spaces, to increase the compatibility with different character encoding standards. The next filter is removing every leading character and space in a sentence until an alphanumeric character, quote or an opening parenthesis is found. This step should decrease the amount of noise at the beginning of sentences, such as bullet points, special characters or other possible Non-ASCII characters. The last cleaning step is removing any HTML remnants that are still part of the data. After applying all three stages of cleaning, the data is ready for labeling. An example of a sentence cleanup is depicted in figure 2.

4.4 Labeling

To train a model that classifies sentences whether they contain statements about the future or not we needed an annotated training set. The active learning approach described in section 4.1 helped us to keep the amount of manual labeled data as small as possible. For the active learning process a small pool with labeled sentences and a bigger with unlabeled sentences is needed. With the methods described in sections 4.1, 4.2 and 4.3. we created a dataset containing 7.790 candidate sentences that could possibly be statements about the future. After shuffling we sliced off 200 sentences and manually annotated whether they are statements about the future (class 0) or not (class 1) according to the considerations we described in section 3. Since 88 sentences in this labeled pool belong to the class 0 containing the statements about the future and 112 belong to class 1 none of the classes is underrepresented and further data balancing is not needed. Before starting the active learning we further split this set into a training set with 120 and a test set with 80 sentences which corresponds to a division of 60% - 40%. To the remaining 7590 candidate sentences we added 5000 sentences that we extracted without using the future statement pre-filtering described in 3.2. Only the data cleaning steps described in 4.3 were applied. During active learning the user is presented with sentences where the level of certainty is not high enough to get classified by the model itself. The user input helps the model to increase precision and label more and more data itself. Because of our pre-filtering steps and the subjectivity in the decision described in section 3. a set with candidate sentences only would include a lot of edge cases that would be presented to the user. To include some clearer cases and increase the model confidence we added 5000 sentences randomly extracted from the given WARC files and receive an unlabeled pool of 12590 sentences.

5 Experiments

5.1 Active learning using small-text

We proceeded with the active learning step as soon as we deemed the initial dataset appropriate for our needs. During the active learning phase, we employed *roberta-base* from Huggingface as the model of choice for identifying future statements. The pool-based approach on active learning was taken. As for the active learning parameters, we decided to conduct 10 active learning iterations querying 20 samples each using the PredictionEntropy query strategy. In this process, labels 0 or 1 for future statements and non future statements respectively are assigned to each query and are added

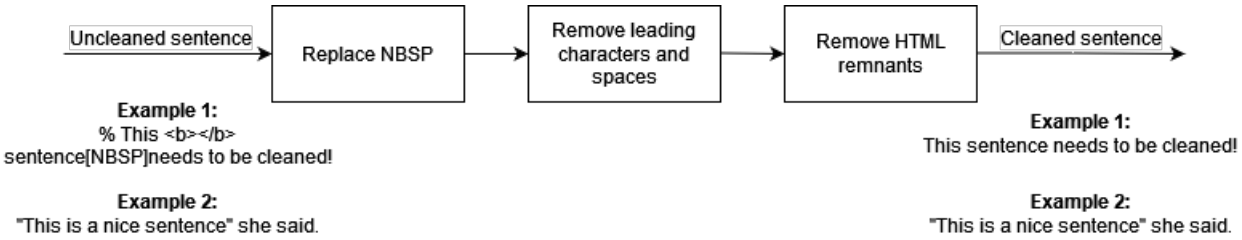

Figure 2: Example of cleaning sentences with regex filters applied to them.

from the unlabeled pool to the train set so that the model is able to make use of the additionally provided information during the retraining phase. Queries which we weren't able to confidently assign a label to were ignored and were not added to the train set.

The model itself was trained with the defaults provided by small-text, namely 10 training epochs with an early stopping criterion to speed up the learning, the AdamW optimizer with a learn rate of $2 * 10^5$ and the default loss provided by roberta-base model.

After the 10th AL iteration, 200 newly labeled samples were added to the train, making up an entire train set of 320 samples.

5.2 Future statement detection and emotion analysis

We put our model into action by classifying a new, huge (20488 samples) unlabeled dataset with our future statement classifier. The output of this operation was then entered into a pretrained emotional classifier (Hartmann, 2022). The output of the emotion classifier was scaled via the softmax function over 7 classes containing the emotions: anger, disgust, fear, joy, sadness, surprise as well as a neutral class. We categorized samples containing future statements according to the highest probability class, i.e. if the class joy has a probability of 0.9 and every other class one of below 0.1, it is assigned to class joy. Samples were no class forms an outright majority were assigned undecided.

6 Results

6.1 Active learning

In our evaluation of the model, we decided to go with a precision focused approach, ignoring recall because our extraction method for the train dataset itself cannot be considered exhaustive in any conceivable way.

Metric		Initial	AL
Precision	Future statement	0.64	**0.76**
	None	0.76	**0.86**
Macro precision		0.70	**0.80**

Table 1: Comparison between the model with initial 120 samples and the model trained on the additional 200 samples provided by Active Learning.

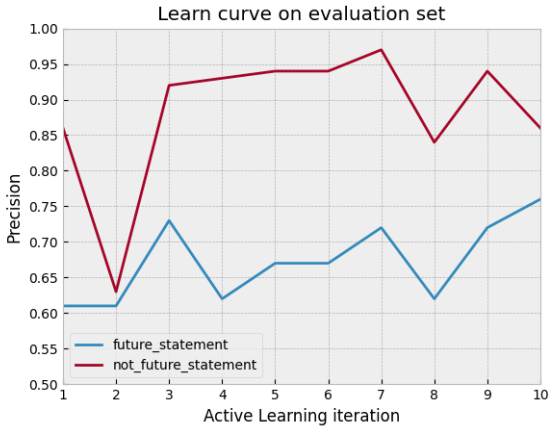

Figure 3: Learn curve on evaluation set

As can be gleaned from the above results, the AL procedure, adding 200 samples, indeed provided sizable performance improvement. We achieved a macro precision of 0.8 with a total of 320 samples in the train set.

Despite the upwards trend of learn curve being discernible, we can also observe "breaks" where the performance of the model after on AL iteration significantly worsens. The worsening is only noticeable within one iterations and improves after the next.

6.2 Classification of Statements about the Future

From the initial unlabeled dataset containing 20488 samples our model classified 9802 sentences as

statements about the future. Table 2 displays some sentences from the classified output.

Sentence	Label
In the next 10 years, investments in renewable energy market is expected to reach 85 billion U.	0
"Phi Sigma Sigma looks forward to working with CampusFundraiser in the future!	0
What's more, the engine manufacturers, the airlines, and the government are actively looking for ways to significantly reduce those emissions in the future.	0
Within a month she knew she had made a mistake.	1
As many purchases as they make within 10 Years of visiting your website, you earn UP TO 50% of EACH one!	1
It was the first new off-Broadway theatre to be built in 50 years in New York City.	1

Table 2: Sentences taken from the labeled dataset. 0 states that a sentence is labeled as a statement about the future.

6.3 Emotion detection

From the 9802 as statements about the future classified sentences the emotion detection model classified a huge majority of 7096 sentences as neutral. This result was expected as people usually don't express any kind of emotions when they state something about things that lie in the future. Surprisingly the mostly expressed emotion was joy as 1048 sentences could be most associated with it. The negative emotions fear, sadness, disgust and anger only count 838 representatives together. 121 sentences were mostly associated with the emotion surprise. 820 sentences were classified as "undecided" as no emotion could reach a confidence of at least 0.5. Example sentences are displayed in Table 3.

Distributions of the classes are displayed in the figures 4 and 5.

Figure 6 displays the average association, of sentences to the class with the highest value, labeled as confidence for this class. The classes were ordered by their frequency as displayed in the figures before. Each class has two bars, one for the confidence with and one without introducing the class "undecided". It is clearly visible that the classes with more representatives also have a higher clas-

Sentence	Emotion
Wang argued that the government had exaggerated the benefits of the deal without mentioning its possible adverse impact on employment and specific sectors, and voiced concerns that the excessive reliance of Taiwanâ€™s investment and exports on Chinaâ€™s market would compromise the countryâ€™s "economic sovereignty" in the future.	anger
I like your ideas about iphone accessories and I hope in the future there can be more bright articles like this from you.	joy
Right of way acquisition could begin in two years.	neutral
In about two yearsâ€™ time, seven of the current nine will have reached retirement age and will need to step aside.	sadness

Table 3: Sentences taken from the dataset after emotion detection. Some sentences still contain some noise but are classified correctly.

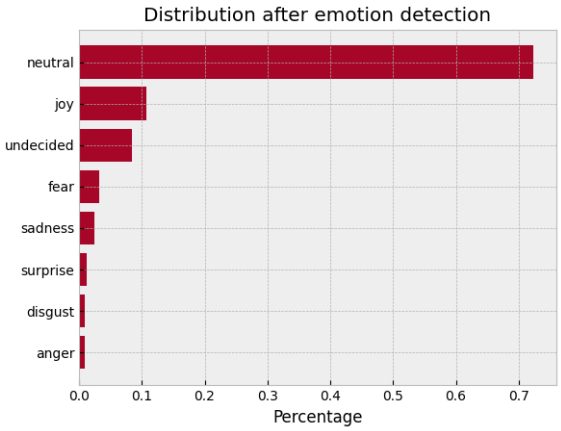

Figure 4: Distribution of emotions associated to the input sentences

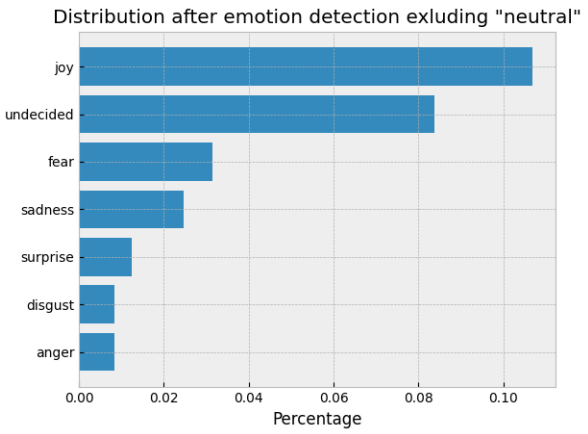

Figure 5: Distribution of emotions associated to the input sentences without the neutral class

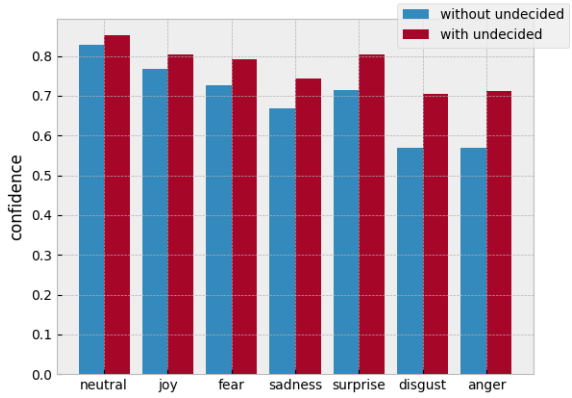

Figure 6: Mean confidence of all classes associated to each sentence with and without pruning sentences with low confidence to the "undecided"-class

sification confidence. Despite the sentences associated with the emotion "surprise" the confidence of all other emotion classes is in the same place as when the classes are ordered by the amount of representatives. In all cases setting a threshold and introducing the "undecided"-class could increase confidence.

7 Conclusion

As one can see on the learn curve of the 10 AL iterations, the increasing improvement of the precision metrics becomes interrupted by "breaks" in iterations 2, 4 and 8. This have to be explained by inconsistencies during the labeling process that are common in subjective text classification tasks. During Active Learning, it is important to label consistently. If not a rigorous approach for annotation is taken, the likelihood for divergence between the train and test increases substantially, threatening the success of the procedure. Pre-defined annotation guidelines and multiple annotators might help alleviate this issue.

The emotion analysis shows, rather unsurprisingly, neutral statements as being by far the most common kind of statement about the future. This highlights the importance of a neutral class in emotion analysis which is sorely missing in many pre-trained models found in HuggingFace hub.

When it comes to sentiment analysis in general, it is worth mentioning that the emotions detected by model may not be necessarily linked to the future scenarios described in the statement. Aspect based sentiment analysis is a set of approaches which try to assign sentiments or emotions to certain aspects of a text or sentence. (Zhang et al., 2022) Using such methods could help to improve the validity of the results by explicitly linking sentiment/emotion with an aspect concerning the future.

Furthermore, methods of Topic Modelling can be employed to extract common topics associated with each emotion class. Classical methods such as Latent Dirichlet Allocation (LDA) but also newer methods using sentence embeddings as in (Angelov, 2020), (Grootendorst, 2022) can be used to explore the content's the data in more detail.

References

Dimo Angelov. 2020. Top2vec: Distributed representations of topics.

Niklas Deckers. 2022. WARC-DL. https://github.com/webis-de/WARC-DL.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Maarten Grootendorst. 2022. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*.

Jochen Hartmann. 2022. Emotion english distilroberta-base. https://huggingface.co/j-hartmann/emotion-english-distilroberta-base/.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Kevin Roebuck. 2011. *Sentiment analysis*. Tebbo.

Christopher Schröder, Andreas Niekler, and Martin Potthast. 2022. Revisiting uncertainty-based query strategies for active learning with transformers. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2194–2203.

Christopher Schröder, Kim Bürgl, Yves Annanias, Andreas Niekler, Lydia Müller, Daniel Wiegreffe, Christian Bender, Christoph Mengs, Gerik Scheuermann, and Gerhard Heyer. 2021a. Supporting land reuse of former open pit mining sites using text classification and active learning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Association for Computational Linguistics.

Christopher Schröder, Lydia Müller, Andreas Niekler, and Martin Potthast. 2021b. Small-text: Active learning for text classification in python.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 2022. A survey on aspect-based sentiment analysis: Tasks, methods, and challenges. *arXiv preprint arXiv:2203.01054*.

A Appendix

Supplementary Materials:

- Repository: https://github.com/ja25opir/nostradamus-project
- Model & Modelcard, Training-Data & Datasheet: https://speicherwolke.uni-leipzig.de/index.php/s/Zbxdsy5sREaS9Jw

Author Contributions:

Nico Schmidt: Conceptualization, Software, Formal Analysis, Investigation, Resources, Data Curation, Writing. **Marvin Müller:** Conceptualization, Software, Formal analysis, Investigation, Resources, Data Curation, Writing, Visualization, Project administration. **Klaus Schmidt:** Conceptualization, Software, Active Learning, Model Application, Writing, Editing, Visualization.