Context-aware Classification of News Comments

Johannes Filter
Hasso Plattner Institute, University Potsdam
Potsdam, Germany
Johannes.Filter@student.hpi.de

ABSTRACT

Automatically analyzing user comments can support journalists to manage the increasing amount of comments in the comment section. Previous work focused on comments without considering their context, such as the corresponding article or other comments. In this master's thesis, we propose and evaluate several deep learning approaches to classify news comments *with* their context. We compare our results to previous work on published, annotated corpora of online comments. In addition, we propose a method to harness user votes to derive a comparable popularity score for each comment.

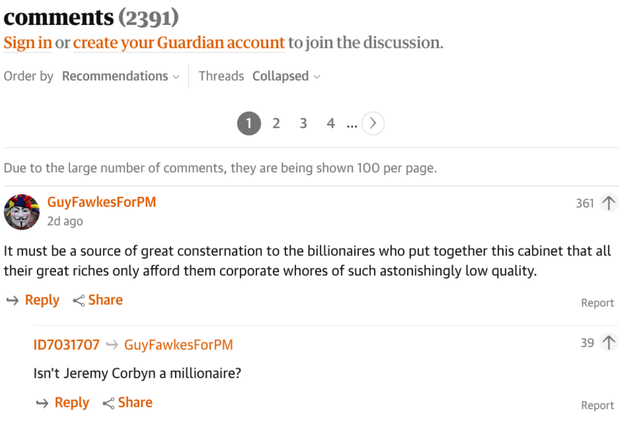
KEYWORDS

Online News Comments, User-generated Content, Natural-Language Processing, Text Classification, Machine Learning, Deep Learning

1 INTRODUCTION

Online news outlets are drowning in the vast quantity of user comments. While there are certainly high-quality comments that, i.e., give a new perspective to a story, there are as well comments which are rather unwelcome. They range from out-of-topic discussions to use of offensive language. A large number of websites are closing their comment sections because they cannot cope with the sheer amount of user-generated content. News organizations already have economic difficulties[1] and they cannot afford to moderate comments. However, with the help of new Natural-Language Processing (NLP) and machine learning techniques, there is hope to automatically analyze user comments. As a consequence, it frees up resources from moderation and opens the comment sections again.

There already exist works to detect hate speech or abusive language in user-generated content [5, 23, 33, 37]. In this work, we want to take a different perspective on the issue. Instead of eliminating 'bad' we want to identify high-quality (or 'good') comments. We achieve this by classifying according to fine-grained criteria that constitute those comments. Such criteria are, for instance, whether a comment is about the topic of the article or if it supports its claims with an argument. In this master's thesis, we want to achieve a higher precision and recall on classifications on corpora of news comments reported in the scientific literature. The detailed description of what classes we want to predict on which datasets are given in further sections. The successful classification of specific fine-grained criteria allows us to (hopefully) detect high-quality comments. For the assignment of which criteria belongs to 'bad' comments or 'good', we rely on research in the field of media studies.



comments (2391)
Sign in or create your Guardian account to join the discussion.

Order by Recommendations | Threads Collapsed

1 2 3 4 ...

Due to the large number of comments, they are being shown 100 per page.

GuyFawkesForPM
2d ago
361

It must be a source of great consternation to the billionaires who put together this cabinet that all their great riches only afford them corporate whores of such astonishingly low quality.

Reply Share Report

ID7031707 → GuyFawkesForPM
39

Isn't Jeremy Corbyn a millionaire?

Reply Share Report

Figure 1: An opinion article of the Guardian about the UK Prime Minister Theresa May received 2391 comments within two days.[2]

In contrast to prior work, we assume that the article and other comments are important to be considered when making classifications. To illustrate our motivation, we take two humanly-annotated comments from a Canadian newspaper [16]: "Time for the elders and chiefs to stand up to the plate and take a leadership role!". This comment is labeled as constructive (or in other words: high-quality) and "Maybe this will motivate the cabbies in TO to clean their filthy cars! That are a disgrace." This is labeled as non-constructive. For us as humans, it is hard to make a judgment without reading the article first. Moreover, the annotators were required to read the article first before deciding whether an article is constructive or not. So we should be fair and also give the machine the possibility to obtain the context before classifying.

In order to formally define the problem, let C and A be the set of all comments and articles respectively. In addition, we define:

$$\begin{aligned}&\text{isArticle} = \{(c, a) | \forall (c, a) \in C \times A : c \text{ is a comment of } a\}\\&\text{isSurrounding} = \{(c, o, a) | \forall (c, o, a) \in C \times C \times A :\\&\qquad \text{isArticle}(c, a) \wedge \text{isArticle}(o, a) \wedge c \neq o\}\end{aligned}$$

[1] https://en.wikipedia.org/wiki/Decline_of_newspapers

[2] https://www.theguardian.com/commentisfree/2018/oct/03/the-guardian-view-on-theresa-mays-speech-getting-by-on-borrowed-time

The training set $T = \{(c_n, a_n, s_n, y_n)\}_{n=1}^{N}$ consists of quadruple-wise data, where $c \in C$, $a \in A$, $s \subseteq \{o | \forall o \text{ isSurrounding}(c, o, a)\}$ and $(c, a) \in \text{isArticle}$. In addition, $y \in Y$ is the corresponding label for $Y = \{1, ..., l\}$ classes. We wish to learn a classifier γ that maps a comment, its article, and its surrounding comments to classes: $\gamma : C \times A \times C^{*} \rightarrow Y$.

In addition, we present a method of harnessing user votes to derive a comparable popularity score for each comment. We assume that the popularity of a comment is a rough proxy for its quality. So, with a comparable score, we can, i.e., divide comments into high-quality and low-quality comments. Nevertheless, we are aware that a high number of up-votes does not necessarily indicate a good comment. A German right-wing internet group 'Reconquista Germanica'[3] coordinates attacks on their political opponents – among other things – in the comment section. They flood it with racist remarks and also up-votes them. Consequently, considering those comments and votes may lead to confounding good with racist. However, we still assume that, especially in an enormous corpus, the vote of the crowd has a meaning. News aggregators such as Hacker News[4] or Reddit[5], live from their user votes to identify high-quality content. Still, there are a lot of factors that come into play when people vote and our method only operates on the number of votes at one specific moment in time. We neither know who casts these votes nor when. This may not be enough to come to a reasonable, objective, and comparable popularity score. We will critically investigate our method on multiple existing comment datasets before drawing conclusions.

So what is the bigger picture of user comments? Why exactly is it important to have them? Glenn Greenwald[6] states:

> "Journalists often tout their responsibility to hold the powerful accountable. Comments are a way to hold journalists themselves accountable."

So, one facet is about controlling the journalists themselves. Before the Internet, people could only send letters to the editors to express their opinions. A fraction of those letters was printed but those letters were obviously moderated strictly. Thus, the introduction of online news comments was an emancipatory act of depriving journalists of their role as gatekeepers. For the first time, a debate about an issue could happen in the wider public. This touches on the discourse ethics[7] as proposed by Jürgen Habermas into practice. The principle of discourse ethics is that a group of people with a rationale debate comes to a common conclusion which manifests the morale. This is a stark contrast to Immanuel Kant's categorical imperative which focuses on individuals' convictions. The theoretical concept of the discourse ethics can be set in practice in the comment section albeit in some variation. In essence, it is about the belief that the collective can come to better conclusions than a sole person. The journalist who wrote the article has a limited view on the issue. Over the course of the debate in the comment section, all participants will eventually reach a common conclusion. But to do so, they have to follow specific discourse rules – an idealistic scenario. Nevertheless, Jürgen Habermas, and other philosophers, thoughts can help us to answer fundamental questions.

The field of journalism is tightly coupled with the technological advancement of our society. Only through the invention of the letterpress by Johannes Gutenberg could humans spread information so fast. This allowed the journalistic profession to establish and flourish. The ongoing digital revolution also affects journalists and there is a long tradition of supporting their work with digital technologies. It started in the late 1960s with computer-assisted reporting[8] and lead to current ideas about automatic reporting[9]. Right now, there is a larger effort by supporting newspapers in managing their user comments. On example is the unprecedented Coral Project[10], a cooperation among Mozilla, the New York Times and the Washington Post, that interviewed more than 400 experts in 150 newsrooms to develop an IT system to manage comments. In the following section, we give a detailed overview of related scientific work of machine-learning-based natural-language processing on news comments.

2 RELATED WORK

The literature that is related to this work can be roughly split into two categories. One is about news comments and machine learning applied to it. And the second is about recent trends in NLP with deep learning.

2.1 News Comments

With the beginning of Web 2.0, there is an abundance of user-generated content. Some influential earlier work analyzed comments on Digg[11] [9, 17], or predicted the popularity of online content on Youtube and Digg [39] or Reddit [34]. Coming from the area of human-computer interaction, Park et al. [25] build a system to manage comments by incorporating traditional, feature-based machine learning.

Recent work focused on identifying moderator-picked comments on New York Times website, so-called *NYT Picks*. Nicholas Diakopoulos [6] presents nine criteria for comments that distinguish NYT Picks from non-NYT Picks. Three criteria can be computed. Kolhatkar and Taboada [15] predict the NYT Picks with traditional, feature-based machine learning as well as deep learning. They achieved an F1 score of 0.84. In their follow-up work, they constructed the *SFU Opinion and Comments Corpus* [16], where they labeled over 1000 comments for *constructiveness* and they achieve an accuracy of 72.59% [14]. They define constructive as:

[3] https://en.wikipedia.org/wiki/Social_media_disruption_by_far-right_groups_in_Germany

[4] https://news.ycombinator.com/

[5] https://www.reddit.com/

[6] https://theintercept.com/2017/12/18/comments-coral-project/

[7] https://en.wikipedia.org/wiki/Discourse_ethics

[8] https://en.wikipedia.org/wiki/Computer-assisted_reporting

[9] https://en.wikipedia.org/wiki/Automated_journalism

[10] https://coralproject.net/

[11] http://digg.com/

> "Constructive comments intend to create a civil dialogue through remarks that are relevant to the article and not intended to merely provoke an emotional response. They are typically targeted to specific points and supported by appropriate evidence."

Napoles et al. [21] focused on detecting 'good' discussions on a comment thread level. They define 'good' discussions as ERIC: Engaging, Respectful, and/or Informative Conversation. They as well created a corpus of comments, the *Yahoo News annotated comments corpus* [22], labeled a subset of about 10k comments and 2.4k threads and predicted ERIC threads with an F1 score of 0.73. For German, Schabus et al. [36] constructed a dataset of comments and annotated over 11k of them. They present several approaches and experiment with features-based as well as deep learning based. In their follow-up paper Schabus and Skowron [35] describe how they resort to a feature-based machine method for usage in a production system. A detailed comparison of all related datasets is presented later in Section 4.1.

The work so far did not consider the context of an article. Namely, the article and also other comments. The very recent of work by Cheng et al. [4] considers the abstract of the news article as well as surrounding comments to classify comments. It adapts the text matching methods of Wang et al. [40] that uses an LSTM [11] with attention mechanism. However, their work has one weakness. They interpret all comments with over 10 up-votes as positive and the rest as negative samples in a binary classification problem. This is a gross simplification for multiple reasons. For instance, earlier comments are more likely to get more up-votes. So they may only predict comments that appear earlier in the discussions than others. Even though they achieved an accuracy of 70.75% and an F1 score of 0.8073, their true contribution is unclear. Nevertheless, their deep learning network architecture of combining comment, article, and other comments can be a starting point for our work.

There has been significant work on detecting hate speech and offensive language with NLP. A closer look is out of the scope of this thesis and the interested reader is guided to a survey by Schmidt and Wiegand [37]. In particular, one paper by Gao and Huang [8] is worth mentioning because they work on context-aware classification. They as well point out that work on comments neglects its context. They developed an architecture of three parallel LSTMs, one for the text, one for the author, and one for the article. The three LSTMs are combined into a classifier. They constructed their own datasets of annotated tweets that relate to news articles. In their experiments, they claim that their context-aware model outperforms the one without context. Unfortunately, they did not apply their method to a commonly used dataset to put their contribution into perspective.

Loosely related is the work by Qin et al. [31] who automatically generate high-quality comments. Also, the problem of *stance detection* is related. In this field, it is about detecting whether a response to a statement is affirmative or opposing. Some promising work has been carried out by Kochkina et al. [13]. In addition, the prediction of the helpfulness of user products reviews done by Singh et al. [38] is relevant. Another work on product reviews and recommender system has been done by Zheng et al. [41]. They encoded the product description as well as reviews with an LSTM respectively before combining the two sub-networks into a classifier.

Outside of the computer science community, there exists qualitative analysis of comments that should guide our work. Loosen et al. [18] formulated several comment quality indicators after conducting interviews with news professionals as well as developing a software mockup. Among other things, they list 'comment length' and 'readability' but also the 'reference to the article' and 'references to other comments' as an indicator. Their work is built upon earlier research done by Diakopoulos et al. [7] and Noci et al. [24]. There is also an abundance of comment guidelines that outline good comments. The New York Time writes in their guidelines[12]: "We are interested in articulate, well-informed remarks that are relevant to the article." Also, the community guidelines by the Guardian requires commentators to "[k]eep it relevant"[13].

So the relation to the article plays a role when judging comments for its quality. Current machine learning methods often are too simplistic and neglect this feature all-together.

2.2 Transfer Learning with Language Models

Transfer learning describes methods on how to re-use (intense) computations on downstream tasks. Pre-trained word embeddings, as popularized by Word2Vec [20], GloVe [26] or fastText [2], are one example of transfer learning in NLP. Within the deep learning NLP community, there is a trend of abandoning word embeddings in favor of text representations derived from pre-trained language models. There are two main reasons for this: first, the word tokens are global which means that they are context-agnostic. This is a drawback because there might be multiple meanings for a word. For instance, 'bank' can describe a financial institution or an embankment. The true meaning in a sentence can only be inferred by considering the context. Second, the meaning of the word is lost after the embedding layers. Subsequent, deeper layers do not have access to the 'meaning' of a word that was injected through the word embeddings.

Language Models (LM) try to predict the next word based on previous words. This is a challenging task. In such a manner, the model has to learn the nuances of a language over long sequences of text. A good thing is that they do not require annotated data and there is an abundance of raw text freely available (e.g. from Wikipedia). After training an LM, there is the hope of transforming its capabilities to other tasks. The Elmo embeddings by Peters et al. [28] use

[12] https://help.nytimes.com/hc/en-us/articles/115014792387-Comments

[13] https://www.theguardian.com/community-standards

the potential of LM to surpass earlier work on contextual word embeddings by McCann et al. [19] (CoVe). They use the internal state of a language model to obtain vectors as a building block that can directly replace traditional word embeddings. Similar work has been done by of Abkib et al. [1] for Flair[14]. Peters et al. [29] describe LMs and their use in text representation in more detail in their recent publication.

Howard and Ruder show with Ulmfit [12] how to apply transfer learning to fine-tune from an ordinary LM to downstream classification task. They achieved multiple state-of-the-art performances on text classification[15] and sentiment detection[16], e.g. on the IMDb dataset[17]. In contrast to Elmo, Ulmfit consists of a full architecture and methodology to apply text classification to real-world data. They point out that the challenging part is the method of how to fine-tune from the LM to the task. The OpenAI transformer by Radford et al. [32] also starts with a trained LM and *transforms* it to downstream tasks. Their performance for text classification is below than Ulmfit. But they provided also ways to transform the LM to other tasks such as question answering. Both papers highlight that there is far more to explore in the field of LM fine-tuning. There is a comparison of several approaches done by Perone et al. [27] – unfortunately without Ulmfit or the OpenAI transformer.

Transfer learning in NLP is currently a hot topic and there might be no clear answers yet. There are also other approaches without LM, i.e., the 'Universal Sentence Encoder' by Cer et al. [3]. They also create context-aware text representation but build upon their prior work about a custom artificial neural network architecture.

3 CONTEXT-AWARE CLASSIFICATION OF NEWS COMMENTS

We want to draw from ideas of the recent developments in NLP to classify news comments considering their context. We propose and evaluate several deep learning architectures. We interpret the problem as a combination of text classification and text matching. We follow the recent advancements of language model fine-tuning as done by Ulmfit [12] and the OpenAI Transformer [32]. We will first present several context-unaware baselines approaches before describing our context-aware architectures.

3.1 Baseline Models

The baseline methods range from traditional feature-based approaches to well-established deep learning approaches. All of them do not have the context of the article. For the features, the work by Kolhatkar and Taboada [15] with features such as average word/sentence/comment length is a starting point. And also stylometric features as described by Potthast et al. [30] should help to distinguish certain quality criteria. Most importantly we use well-established, i.e., attention-based deep learning architectures. Further, we will use text classification with the original Ulmfit [12] solely on the comment's text. This way we have to ensure that the additional context is useful.

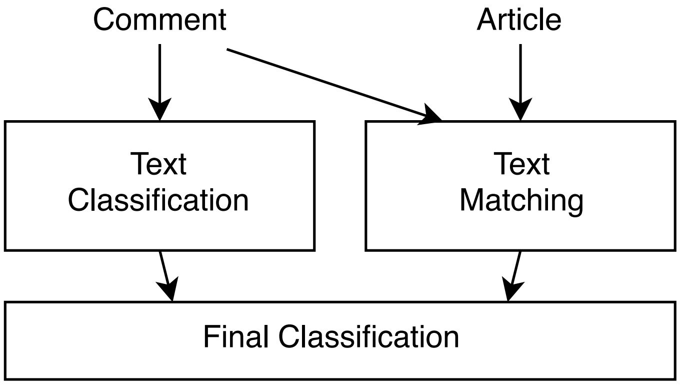


Figure 2: Overview of the model architecture.

3.2 Context-aware Models

We propose three different directions for context-aware models. For the beginning, we limit the models to only consider the article as context and not the surrounding comments. First, we start with simple architectures using multiple recurrent neural networks, i.e., LSTMs, in parallel. One for the comment and one for the article before joining together in a fully connected layer. This follows the work of Gao and Huang [8] who used it to classify hate speech tweets. Or Zheng et al. [41] who used it for product reviews.

Second, we propose a model architecture consisting of two sub-networks, one only on the comment's text and one also for the text matching between the article's text and the comment's text. Both sub-structures are combined into a final layer to classify the comments. For the text classification, we follow the work of Ulmfit [12]. For the text matching, we follow the work of the OpenAI transformer [32]. Text matching describes the general research problem whether two texts match while the matching criteria can be defined. Examples of text matching are question answering. So the general idea is that we first train a general language model on a large text corpus. Then, transform and fine-tune the trained language model to our task. An overview of the complete architecture is shown in Figure 2. The whole process of fine-tuning looks as follows:

(1) Train a language model on a large, un-labeled corpus (or use a pre-trained one)
(2) Re-train the language model on an un-labeled corpus of comments
(3) Fine-tune the model on a labeled sub-set of comments

[14] https://github.com/zalandoresearch/flair
[15] http://nlpprogress.com/text_classification.html
[16] http://nlpprogress.com/sentiment_analysis.html
[17] https://ai.stanford.edu/~ang/papers/acl11-WordVectorsSentimentAnalysis.pdf

Third, we want to experiment with whether classification criteria ‘off-topic’ can be solved as a text matching problem. To our knowledge, this has not tried out so far. This would make the complex two-step architecture proposed in the previous paragraph superfluous.

Dataset	Description	Annotations
SFU Opinion and Comments Corpus [16] (SOCC), in English	10k articles with their 663k comments from 303k comment threads, from 2012 to 2016, from Canadian newspapers, with up- and down-votes	1,043 comments in responses to 10 articles, labeled for constructiveness and toxicity
Yahoo News Annotated Comments Corpus [22] (YNACC), in English	522k comments from 140k threads posted in response to Yahoo News articles, with up- and down-votes	9.2k comments labeled for agreement, audience, persuasiveness, sentiment, tone, (off-)topic and 2.4k threads labeled for agreement, constructiveness, type
One Million Posts Corpus [35, 36] (OMPC), in German	12k articles, 1M comments, from an Austrian Newspaper, with up- and down-votes	11k comments with the following labels: sentiment, off-topic, inappropriate, discriminating, feedback, personal studies, argument used
Tencent News Corpus (TNC) by Qin et al. [31], in Chinese	200K articles and 4.5M comments, from a Chinese news website, 2017, with up-votes	40k comments labeled for quality (from 1 to 5)

Table 1: Overviews about annotated corpora of comments.

3.3 Further Ideas

To further improve the model, we could incorporate external knowledge into our model. Some concrete information about people and e.g. their affiliation, i.e., political party membership, may not be in the training data. One way would be to use a Named-Entity Recognition (NER) method and then to look up the entity on external knowledge resources such as Wikipedia or Wikidata.

4 DATA

In this section, we will give an overview of available partly-annotated data sources as well as a method to harness user votes.

4.1 Corpora of News Comments

There exists four corpora of news comments where part of the comments are annotated for its quality. Table 1 gives an overview about them, two are in English, one in German and in Chinese.

For our work, the YNACC and the OMPC are especially interesting because of their fine-grained annotations. The number of annotated comments in SOCC are too few. TNC has enough data but only one label of quality. YNACC and OMPC have also some overlapping annotation criteria: off-topic and sentiment. This allows us to compare the same method with identical labels on both datasets. Since YNACC contains English and OMPC German comments, we can test our approach on two languages.

Besides, all of the corpora contain comment-level user votes. In order to make use of them, a method is presented in the next section.

4.2 Harnessing User Votes

News comments allow people to express their opinion about news articles. In the same breath, users are also able to vote on other users' comments. This can be interpreted as a label of the popularity of a comment. However, a problem is that one cannot take the raw number of votes to compare comments. Articles that attract more readers than others receive more comments and also more votes. Comments posted shortly after the article publication are more likely to be read and consequently receive more votes. So to make use of user votes, they first need to be preprocessed to obtain a comparable popularity score.

In our previous, unpublished work, we developed a method to create binary popularity labels out of up-votes as follows:

(1) Remove non-root comments
(2) Remove articles with few comments
(3) Remove articles with few up-votes
(4) Sort comments chronologically
(5) Only consider first N comments per article
(6) Calculate relative portion of up-votes for each comment under *one* article
(7) For each comment rank, classify the first X comments as positive and the last X comments as negative, whereby $X * 2 <=$ number of articles

So we normalize the up-votes first and then, leave out the comments with an average score because we assume that there the 'crowd' did not come to clear decision. Only the very popular and the very unpopular comments were used as positive and negative samples respectively. The constants N and X were derived empirically. In this master's thesis, we want to generalize this method and also determine X and N more scientifically. First, we split the whole process into two parts. The development of an objective popularity score and the method of turning this score into classes. Moreover, we want to adapt it to also work for down-votes. This enables us to use this approach on the datasets in Table 1 which have down-votes. For this two additional classes are required but how exactly the comments are assigned to a class is due to further research.

Finding a way of harnessing user votes to get meaningful labels would allow us to have a magnitude more of annotated comments. Halevy et al. [10] showed that more data, in general, helps to improve the performance of deep learning models. So with this method applied to datasets as presented in Table 1 would give us more data. This gives us the possibility to train a model to detect 'good' comments. A hope is that if you feed those data into a deep learning model, it picks of the general tendency and leaves out the outliers. However, these assumptions need to get evaluated and we will describe in the next section.

5 EVALUATION

We have two different sub-sections for the evaluation because our contribution is two-fold.

5.1 Context-Aware Classification

To test the performance of our proposed architectures as described in Section 3, we compare it to our baselines results and results reported in the scientific literature. The authors of OMPC report baseline values [35] for all their classification criteria. The criteria 'off-topic' with a prediction of 0.2579, recall of 0.6241 and an F1 score of 0.3516 leaves room for improvement. Because our model has the relation between comment and article, it ought to achieve better results. Also, the majority of the other criteria have F1 scores ranging from 0.5 to 0.7 that show these problems are far from being solved. (There will be a paper from Hamburg available shortly, where they also work on the OMPC and report F1 scores). It is our goal to outperform all classifications on OMPC in respect to precision and recall (and thus F1 scores). The classification criteria are: sentiment, off-topic, inappropriate, discriminating, feedback, personal studies, argument used.

Unfortunately, there are no reported values for the comments in YNACC – only per thread level. For SOCC, the authors [14] report an accuracy of 0.7259 for identifying 'constructive' comments. We claim that the context of a comment is important for its classification. Since in the author's definition of constructive is implied, that it is relevant to the article, we have to outperform them to prove our approach works.

5.2 Harnessing User Votes

First, to evaluate our method as proposed in Section 4.2, we compare the resulting classification against the annotated comments in Table 1. There are certain labels such as 'constructiveness', 'argument used', or 'on-topic' that should occur significantly more often in our 'popular' comments.

Second, we preprocess a dataset from Table 1 with our method and train a model, i.e., with a context-aware architecture. Then, we classify unseen comments. The final classification results are then evaluated by human annotators (10 articles, 100 comments, 5 people). With this, we can evaluate the whole multi-step process. We hope, that the combination of our harnessing method and the deep learning model can predict quality labels of comments.

Third, depending on the course of the master's thesis, we can conduct a larger user study. The study is with ten participants with a quantitative and a qualitative part to determine whether the 'popular' comments are actually superior to other comments. First, participants are required to assign quality labels to comments after they read the corresponding article. Participants are asked to read three articles and judge ten comments per article. Second, in a semi-structured interview, we will try to elicit additional information about their decisions to get an understanding of what a good comment is in particular and how it relates to user voting on comments.

REFERENCES

[1] Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual String Embeddings for Sequence Labeling. (2018), 12.

[2] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics* 5 (2017), 135–146.

[3] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. 2018. Universal sentence encoder. *arXiv preprint arXiv:1803.11175* (2018).

[4] D. Chen, S. Ma, P. Yang, and X. Sun. 2018. Identifying High-Quality Chinese News Comments Based on Multi-Target Text Matching Model. *ArXiv e-prints* (Aug. 2018). arXiv:cs.CL/1808.07191

[5] Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated Hate Speech Detection and the Problem of Offensive Language. In *Proceedings of the 11th International AAAI Conference on Web and Social Media (ICWSM '17)*. 512–515.

[6] Nicholas Diakopoulos. 2015. Picking the NYT Picks: Editorial criteria and automation in the curation of online news comments. (2015).

[7] Nicholas Diakopoulos and Mor Naaman. 2011. Towards Quality Discourse in Online News Comments. In *Proceedings of the ACM 2011 Conference on Computer Supported Cooperative Work (CSCW '11)*. ACM, New York, NY, USA, 133–142. https://doi.org/10.1145/1958824.1958844

[8] Lei Gao and Ruihong Huang. 2017. Detecting Online Hate Speech Using Context Aware Models. *RANLP 2017 - Recent Advances in Natural Language Processing Meet Deep Learning* (Nov 2017). https://doi.org/10.26615/978-954-452-049-6_036

[9] Vicenç Gómez, Andreas Kaltenbrunner, and Vicente López. 2008. Statistical Analysis of the Social Network and Discussion Threads in Slashdot. In *Proceedings of the 17th International Conference on World Wide Web (WWW '08)*. ACM, New York, NY, USA, 645–654. https://doi.org/10.1145/1367497.1367585

[10] Alon Halevy, Peter Norvig, and Fernando Pereira. 2009. The unreasonable effectiveness of data. *IEEE Intelligent Systems* 24, 2 (2009), 8–12.

[11] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[12] Jeremy Howard and Sebastian Ruder. 2018. Universal Language Model Fine-tuning for Text Classification. *arXiv:1801.06146 [cs, stat]* (Jan. 2018). http://arxiv.org/abs/1801.06146 arXiv: 1801.06146.

[13] Elena Kochkina, Maria Liakata, and Isabelle Augenstein. 2017. Turing at SemEval-2017 Task 8: Sequential approach to rumour stance classification with branch-LSTM. *arXiv preprint arXiv:1704.07221* (2017).

[14] Varada Kolhatkar and Maite Taboada. 2017. Constructive language in news comments. In *Proceedings of the First Workshop on Abusive Language Online*. 11–17.

[15] Varada Kolhatkar and Maite Taboada. 2017. Using New York Times Picks to Identify Constructive Comments. Association for Computational Linguistics, 100–105. https://doi.org/10.18653/v1/W17-4218

[16] Varada Kolhatkar, Hanhan Wu, Luca Cavasso, Emilie Francis, Kavan Shukla, and Maite Taboada. 2018. The SFU Opinion and Comments Corpus: A Corpus for the Analysis of Online News Comments. (2018).

[17] Cliff Lampe and Paul Resnick. 2004. Slash(Dot) and Burn: Distributed Moderation in a Large Online Conversation Space. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '04)*. ACM, New York, NY, USA, 543–550. https://doi.org/10.1145/985692.985761

[18] Wiebke Loosen, Marlo Hring, Zijad Kurtanovi, Lisa Merten, Julius Reimer, Lies van Roessel, and Walid Maalej. 2017. Making sense of user comments: Identifying journalists requirements for a comment analysis framework. *Studies in Communication | Media* 6, 4 (2017), 333–364. https://doi.org/10.5771/2192-4007-2017-4-333

[19] Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. In *Advances in Neural Information Processing Systems*. 6294–6305.

[20] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *CoRR* abs/1301.3781 (2013). arXiv:1301.3781 http://arxiv.org/abs/1301.3781

[21] Courtney Napoles, Aasish Pappu, and Joel R Tetreault. 2017. Automatically Identifying Good Conversations Online (Yes, They Do Exist!).

[22] Courtney Napoles, Joel Tetreault, Aasish Pappu, Enrica Rosato, and Brian Provenzale. 2017. Finding good conversations online: The Yahoo News annotated comments corpus. In *Proceedings of The 11th Linguistic Annotation Workshop*. 13–23.

[23] Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, and Yi Chang. 2016. Abusive Language Detection in Online User Content. In *Proceedings of the 25th International Conference on World Wide Web (WWW '16)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 145–153. https://doi.org/10.1145/2872427.2883062

[24] Javier Díaz Noci, David Domingo, Pere Masip, JL Micó, and C Ruiz. 2012. Comments in news, democracy booster or journalistic nightmare: Assessing the quality and dynamics of citizen debates in Catalan online newspapers. In *International Symposium on Online Journalism*, Vol. 2. 46–64.

[25] Deokgun Park, Simranjit Sachar, Nicholas Diakopoulos, and Niklas Elmqvist. 2016. Supporting Comment Moderators in Identifying High Quality Online News Comments. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 1114–1125. https://doi.org/10.1145/2858036.2858389

[26] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *In EMNLP*.

[27] C. S. Perone, R. Silveira, and T. S. Paula. 2018. Evaluation of sentence embeddings in downstream and linguistic probing tasks. *ArXiv e-prints* (June 2018). arXiv:cs.CL/1806.06259

[28] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365* (2018).

[29] Matthew E. Peters, Mark Neumann, Luke Zettlemoyer, and Wen-tau Yih. 2018. Dissecting Contextual Word Embeddings: Architecture and Representation. *arXiv:1808.08949 [cs]* (Aug. 2018). http://arxiv.org/abs/1808.08949 arXiv: 1808.08949.

[30] M. Potthast, J. Kiesel, K. Reinartz, J. Bevendorff, and B. Stein. 2017. A Stylometric Inquiry into Hyperpartisan and Fake News. *ArXiv e-prints* (Feb. 2017). arXiv:cs.CL/1702.05638

[31] L. Qin, L. Liu, V. Bi, Y. Wang, X. Liu, Z. Hu, H. Zhao, and S. Shi. 2018. Automatic Article Commenting: the Task and Dataset. *ArXiv e-prints* (May 2018). arXiv:cs.CL/1805.03668

[32] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. (2018).

[33] Julian Risch and Ralf Krestel. 2018. Delete or not Delete? Semi-Automatic Comment Moderation for the Newsroom. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*. 166–176.

[34] Georgios Rizos, Symeon Papadopoulos, and Yiannis Kompatsiaris. 2016. Predicting News Popularity by Mining Online

Discussions. In *Proceedings of the 25th International Conference Companion on World Wide Web (WWW '16 Companion).* International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 737–742. https://doi.org/10.1145/2872518.2890096

[35] Dietmar Schabus and Marcin Skowron. 2018. Academic-Industrial Perspective on the Development and Deployment of a Moderation System for a Newspaper Website. (2018), 4.

[36] Dietmar Schabus, Marcin Skowron, and Martin Trapp. 2017. One Million Posts: A Data Set of German Online Discussions. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17)*. ACM, New York, NY, USA, 1241–1244. https://doi.org/10.1145/3077136.3080711

[37] Anna Schmidt and Michael Wiegand. 2017. A survey on hate speech detection using natural language processing. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*. 1–10.

[38] Jyoti Prakash Singh, Seda Irani, Nripendra P. Rana, Yogesh K. Dwivedi, Sunil Saumya, and Pradeep Kumar Roy. 2017. Predicting the helpfulness of online consumer reviews. *Journal of Business Research* 70 (2017), 346 – 355. https://doi.org/10.1016/j.jbusres.2016.08.008

[39] Gabor Szabo and Bernardo A. Huberman. 2010. Predicting the Popularity of Online Content. *Commun. ACM* 53, 8 (Aug. 2010), 80–88. https://doi.org/10.1145/1787234.1787254

[40] Zhiguo Wang, Wael Hamza, and Radu Florian. 2017. Bilateral Multi-Perspective Matching for Natural Language Sentences. *arXiv:1702.03814 [cs]* (Feb. 2017). http://arxiv.org/abs/1702.03814 arXiv: 1702.03814.

[41] Lei Zheng, Vahid Noroozi, and Philip S. Yu. 2017. Joint Deep Modeling of Users and Items Using Reviews for Recommendation. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining (WSDM '17)*. ACM, New York, NY, USA, 425–434. https://doi.org/10.1145/3018661.3018665