The 33rd Conference on Computational Linguistics and Speech Processing (ROCLING 2021)
Taoyuan, Taiwan, October 15-16, 2021. The Association for Computational Linguistics and Chinese Language Processing

What confuses BERT? Linguistic Evaluation of Sentiment Analysis on Telecom Customer Opinion

Cing-Fang Shih
National Taiwan University
r08142004@ntu.edu.tw

Yu-Hsiang Tseng
National Taiwan University
seantyh@gmail.com

Ching-Wen Yang
National Taiwan University
b06102020@ntu.edu.tw

Pin-Er Chen
National Taiwan University
cckk2913@gmail.com

Hsin-Yu Chou
National Taiwan University
r10142008@ntu.edu.tw

Lian-Hui Tan
National Taiwan University
b06102036@ntu.edu.tw

Tzu-Ju Lin
National Taiwan University
b05102085@ntu.edu.tw

Chun-Wei Wang
Chunghwa Telecom Laboratories
chriswang@cht.com.tw

Shu-Kai Hsieh
National Taiwan University
shukaihsieh@ntu.edu.tw

Abstract

Ever-expanding evaluative texts on online forums have become an important source of sentiment analysis. This paper proposes an aspect-based annotated dataset consisting of Chinese telecom reviews on social media. We introduce a data category called implicit evaluative texts, *impevals* for short, to investigate how the deep learning model works on these implicit reviews. We first compare two models, BertSimple and BertImpvl, and find that while both models are competent to learn simple evaluative texts, they are confused when classifying *impevals*. To investigate the factors underlying the correctness of the model's predictions, we conduct a series of analyses, including qualitative error analysis and quantitative analysis of linguistic features with logistic regressions. The results show that local features that affect the overall sentential sentiment confuse the model: multiple target entities, transitional words, sarcasm, and rhetorical questions. Crucially, these linguistic features are independent of the model's confidence measured by the classifier's softmax probabilities. Interestingly, the sentence complexity indicated by syntax-tree depth is not correlated with the model's correctness. In sum, this paper sheds light on the characteristics of the modern deep learning model and when it might need more supervision through linguistic evaluations.

Keywords: Linguistic Evaluation, Sentiment Analysis, Implicit Evaluative Text, Deep Learning

1 Introduction

In recent years, social networking has revolutionized ways of communication and information exchange. People nowadays are allowed to express their feelings and views instantly online. With their immediate and ubiquitous nature, online reviews have become a valuable source of information extraction. To process crucial information, natural language processing (NLP) is applied to analyze textual data. Sentiment Analysis/Opinion Mining is an important branch of NLP to achieve the goal of opinion mining and social listening. Among all the related tasks such as *Opinion holder detection*, *Subjectivity Analysis*, Aspect-based Sentiment Analysis (ABSA) on short texts has been extensively explored, which specifies the polarity of each aspect in a sentence, providing comprehensive data for sentiment classification.

As neural models prosper, deep learning approaches are widely used in sentiment analysis. The participation of pre-trained models has promoted the accuracy of sentiment detection. Although deep learning models perform well in many circumstances, there are still some unresolved problems. The main concern lies in the model's tendency of learning explicit features while overlooking implicit ones, such as sarcasm, common sense, and deep reasoning. These limitations could hinder the models from making progress in recognizing sentiment. Considering the complexity of language,

we aim to find out what linguistic features affect text classification accuracy of BERT (Devlin et al., 2018), the current state-of-the-art NLP model.

This paper is organized as follows. Section 1 provides the introduction and brings about the research question. Section 2 briefly reviews related literature. Annotation methods and details of the model will be explained in section 3. Subsequently, empirical evaluation is discussed in section 4. Section 5 analyzes the results. Finally, section 6 concludes the paper.

2 Literature Review

According to Liu (2012), sentiment analysis can be classified into three levels, namely the document level, the sentence level, and the aspect level. The document-level sentiment analysis assumes that there is only one topic in one document. Similarly, the sentence level extracts sentiment polarity based on one single sentence. However, in real-life social media comments, there could be several topics within only one sentence or document. ABSA thus solves the issue by indicating the polarity of each aspect in a sentence. It has received recognition and has become an important research field in computational linguistics.

Different approaches have been used in sentiment analysis. For instance, Pang and Lee (2009) presented a task utilizing traditional machine learning methods for document-level sentiment analysis. Tsytsarau and Palpanas (2012) later proposed four different approaches for document-level polarity prediction, namely machine learning-based, dictionary-based, semantic-based, and statistical-based respectively. As for ABSA, Schouten and Frasincar (2015) introduced a machine learning technique including aspect extraction and classification. Lately, as neural models flourish, deep learning-based sentiment analysis has become prominent in the research community. Zhou et al. (2019) provided an in-depth analysis of the deep learning-based aspect-level sentiment classification (ASC). Although the neural model is undoubtedly a practical approach in ASC, Zhou et al. (2019) pointed out its limitation of learning explicit emotional expression exclusively. Implicit emotional expressions such as irony, deep reasoning, and common sense were still too complicated for the recent neural networks (Zhou et al., 2019).

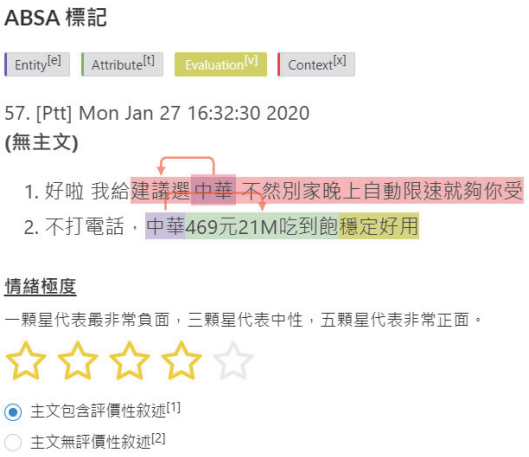

Figure 1: Annotation interface built with Label-Studio. The evaluative text number 1 says "Alright, I suggest choosing CHT, otherwise you will be fed up with other telecoms' automatic speed limit at night." Number 2 says "If you are not calling, the 469 NTD plan with 21M unlimited data offered by CHT is stable and handy." The star signs allow annotators to rate the polarity. One is for the most negative, three is for neutral, and five is for the most positive.

Many attempts have been made to fill in the missing piece of the puzzle. Cui et al. (2020) conducted a quantitative analysis to test the performance of BERT solving the CommonsenseQA task and concluded that with fine-tuning, BERT was able to make use of common sense features on higher layers. Baruah et al. (2020) challenged BERT's ability of context-aware sarcasm detection. They found out that contextual information slightly improved the performance of the Twitter data set, but not the Reddit data set. Ways to utilize the context and its features to improve the model performance is still a debated topic in the research community.

3 Data Annotation

This paper concentrates on the public customer reviews of the telecommunications service. To reflect the realistic opinion of customers, all of the data regarding service providers were extracted from popular anonymous forums, including PTT, Dcard, and Mobile01 to name a few. Comments without

evaluative information, such as reposted news or special offers promotion, were eliminated in this task. If a thread contained an evaluation, its aspect tuple would then be annotated. All of the data were annotated by six linguistic-trained students from National Taiwan University. The annotation interface was built with LabelStudio (Tkachenko et al., 2020-2021) (see Figure 1 for the interface screenshot).

There are three elements in an aspect tuple: (i) an entity, (ii) an attribute, and (iii) an evaluation text. (i) The entity in this task refers to the service provider, and (ii) the attribute refers to the service. To improve the annotation, domain-specific information such as aliases of different providers and types of services are provided by the Department of Customer Service at Chunghwa Telecom. Finally, (iii) the evaluation text is a phrase including a customer's evaluative review of a certain provider or service. In other words, the evaluation text is usually where the sentiment cues appear. An example of a comment thread is demonstrated in (1).

(1) 平常生活圈自用中華網路都蠻順的
píngcháng shēnghuóquān zìyòng
daily living.sphere personal.use
zhōnghuá wǎnglù dōu mán
Chunghwa.Telecom internet always pretty
shùn de
smooth MOD
'For daily personal use in the living sphere, the internet provided by Chunghwa Telecom is always pretty smooth.'

In (1), the entity is the service provider *zhōnghuá* 'Chunghwa Telecom', the attribute is the feature of service *wǎnglù* 'internet', and the evaluation is the phrase *mán shùn de* 'pretty smooth'.

In some cases, there is more than one entity or attribute in a comment, as shown in (2). This is when aspect-based annotation proves useful. All of the entities, attributes, and evaluations found in an opinion thread as well as their corresponding relationship would be annotated.

(2) 中華網路穩定但費用貴
zhōnghuá wǎnglù wěndìng dàn
Chunghwa.Telecom internet stable but
fèiyòng guì
fee expensive
'The internet of Chunghwa Telecom is stable, but the fee is expensive.'

In (2), the entity is *zhōnghuá* 'Chunghwa Telecom'. The mixed feeling appears in the two evaluative adjectives *wěndìng* 'stable' and *guì* 'expensive', which points to two different attributes, *wǎnglù* 'internet' and *fèiyòng* 'fee', respectively.

The rating of the sentiment polarity came right after the annotation of the aspect tuple. Annotators rated the polarity as positive, neutral, or negative according to the sentiment conveyed in the comment thread.

There are some cases that even if a thread does not include a complete aspect tuple, it still conveys evaluative opinion. The sentiment cues may be triggered by certain linguistic constructions or syntactic patterns, as exemplified in (3).

(3) 只有遠傳才有距離
zhǐyǒu yuǎnchuán cái yǒu
only Far.EasTone.Telecom only have
jùlí
distance
'Only Far EasTone Telecom has distance.'

The comment thread in (3) only specifies *yuǎnchuán* 'Far EasTone Telecom' as an entity. Both the attribute and the evaluation text are missing. However, the thread still encodes a negative polarity since it is known as the parody of a catchy slogan, 只有遠傳沒有距離 'Only Far EasTone Telecom has no distance'. In this situation, instead of determining the three elements, the whole thread is annotated as an 'impeval' and is given negative polarity. The flowchart of the annotation task is presented in Figure 2.

We were intrigued by the case that a comment thread could express sentiment even if the information was incomplete. Therefore, a BERT classification task was designed to test its performance of predicting sentiment polarity of these impevals. Since there was a wide variety of sentimental expressions in impevals, we assumed that the addition of impevals should improve the model accuracy if BERT could learn from its linguistic features.

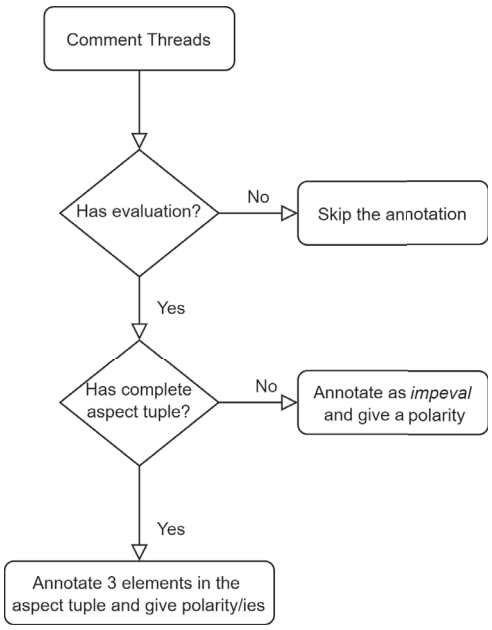

Figure 2: A flowchart presenting the process of the annotation

Dataset	Train ratio	Test ratio	Total
Evaluation	0.8 (1362)	0.2 (341)	1.0 (1703)
Impevals	0.8 (465)	0.2 (117)	1.0 (582)

Note: the number in the bracket indicates the dataset size.

Table 1: The ratio split and data number of each dataset

base model	bert-base-chinese
batch size	32
training epoch	10
learning rate	1.00e-05
weight decay	5.00e-03
optimizer	AdamW
random seed	0

Table 2: Model settings of the sentiment classification

4 Classification model

Two models are trained to compare their performances on simple evaluation texts and impevals. These two models are the same in architecture: they are both fine-tuned sequence classification models based on pretrained Chinese BERT. They are also identical in terms of the number of parameters. The only difference is the first model, BertSimple, only has evaluation texts as its training data. In contrast, the second model, BertImpvl, is trained with both evaluation texts and impevals. The comparison aims to show that impevals are intrinsically different from evaluation texts, as suggested by the annotation guidelines. Therefore, the hypothesis is that the model is less likely to transfer the knowledge it learns from evaluation texts to impevals. Furthermore, the regularities underlying the impevals may be more difficult to capture for the current model.

To train the models, we first split the annotated data into two groups, which were the evaluation dataset and the impeval dataset. In each dataset, training and testing sets were separated with an 8:2 ratio, which is demonstrated in Table 1 (denoted by {ratio}_{dataset_type}, i.e. 0.8_eval, 0.8_impeval, 0.2_eval, 0.2_impeval). To ensure equal label proportion in training and testing distributions, the train-test split was done along with stratification with respect to polarities. We then built two models for further analysis: BertSimple and BertImpvl. The base model is "bert-base-chinese" and its setting is specified in Table 2. While sharing the same architecture and parameters, the only difference between BertSimple and BertImpvl is their experience. BertSimple was trained on 80% of the evaluation dataset (0.8_eval), while BertImpvl was trained on the combination of 80% of the evaluation dataset and 80% of the impeval dataset (0.8_eval+0.8_impeval). It should be noted that since the training set was relatively small after the 8:2 train-test split, no validation set was used during the training phase.

The respective model accuracies of BertSimple and BertImpvl are shown in Table 3. The training accuracies are always very high for both models, which both exceed .98. The model BertSimple, which is only trained with evaluation texts, has validation accuracy of .927 in the evaluation text but only .709 in the impevals. The difference is arguably due to the lack of impevals in the training data. However, the BertImpvl model, trained with impevals, only gains a .06 benefit on the validation accuracy in impevals. It is also worthy to note that the evaluation text accuracy of BertImpvl is .909, which is a slight drop compared with the one of BertSimple.

	Training	Validation	
		Eval	Impevals
BertSimple	0.984	0.927	0.709
BertImpvl	0.981	0.909	0.769

Table 3: Model performance

The overall results are consistent with our hypothesis. Evaluation texts alone are readily learned by transformer-based models, such as BERT. In contrast, impevals are more difficult for the model. Adding the impevals in training data helps, but the model cannot perform as well as it does in evaluation texts. Admittedly, modeling/training-related factors may be responsible for the low accuracy, e.g., inappropriate model architecture, insufficient data size, sub-optimal training parameters, etc. However, evaluation texts and impevals are all linguistic expressions that are used to communicate evaluation polarities. They have similar *functions*, yet their *forms* are different enough so that the same model/training scheme can not readily transfer the regularities between the two. Therefore, it is at least interesting to ask what makes the model confused when classifying the impevals. In the next section, we will try to investigate and show the underlying factors that make the predictions challenging.

5 Linguistic Evaluation on Model Correctness

To investigate the factors underlying the true correctness of model predictions, we follow a three-step analysis scheme. First, we conduct an error analysis against model predictions on impevals. This step serves as an exploratory method to examine possible factors involve in the model's errors. Secondly, we hand-annotated some of the linguistic features in the first step, which are difficult to extract automatically. Once these features are annotated, we could test the relationship between the features and the model's true correctness. Thirdly, we automatically extract the rest of the linguistic features found in step 1. We then show that these features are indeed correlated with the model's correctness of predictions.

5.1 Error Analysis

We conduct an error analysis on the model's predictions of impevals. The model we choose to analyze is BertSimple, which is only exposed to simple evaluation text. With this constraint, we can observe how well the experience of simple evaluations performs on implicit ones. Although having slightly lower accuracy on impevals, it shows a clearer contrast on how models classify impevals based solely on simple evaluation text. We first select all the model's prediction errors and observe, qualitatively, possible factors influencing the predictions. Then, the observations are summarized into 6 categories, as shown in Table 4.

Among the categories identified in error analysis, we aim to find factors that systematically influence the model's correctness. The correctness of the model is conceptually related but distinct from the accuracy metric, which is the proportion of the model's correctness in each item. Some factors are complex, at least not from the off-the-shelf package, to extract automatically, namely, sarcasm and rhetorical questions. We further annotate these factors and explore their relations with model correctness in section 5.2. In contrast, some factors are readily operationalized with automatic NLP tools, such as number of entities, number (and types) of transitions, number of symbols. These factors are further investigated in section 5.3.

5.2 Annotated linguistic features

Rhetorical questions and sarcasm are complex linguistic expressions known to be thorny topics in sentiment classification (Maynard and Greenwood, 2014). Indeed, ongoing studies focus on exploiting the intricacies of linguistic features and developing more flexible model architectures to better detect and weave them into sentiment analysis (Joshi et al., 2017; Seo et al., 2020; Zhuang and Riloff, 2020). However, rhetorical questions and sarcasm are still tricky to handle, and they may still be the contributing factors to why impevals have lower accuracy than simple evaluative texts. Therefore, we try to show that these known-to-be difficult phenomena are still pertinent to the impevals observed in our dataset.

Rhetorical questions and sarcasm expres-

Category	Examples
Sarcasm	只有遠傳，才有距離 Only FET has distance.
Rhetorical question	你看中華吃到飽有限制上網流量嗎？玩不起就不要推出吃到飽 Have you seen CHT unlimited data plan imposing a bandwidth cap? You shouldn't sell the plan if you are not up to it.
Multiple entities	之前遠傳有時收不到，現在中華還算穩定 couldn't get the signal with FET before, but it's stable now with CHT
Transitions	以前爆快但最近有點慢 It's amazingly fast before but slow now.
Symbols	. . . . . , >>>, QQ
Others	我用手機連居然不曾斷過 Surprisingly I never lose the signal with my cell.

Table 4: Error analysis of model's predictions on impevals

	Coeff	*SE*	*z*	*p*
(Intercept)	1.18	0.10	11.54	<0.0001
Rhe. ques.	-0.72	0.07	-1.84	0.0653
Sarcasm	-0.88	0.03	-2.12	0.0342

Table 5: Logistic regression analysis of sarcasm and rhetorical question

sions are manually annotated on each impeval by two annotators. They are both native speakers with linguistics-related majors. Considering the correlation structures among the independent variables, the annotation results were analyzed with a logistic regression model to determine their effect on model correctness. The statistical results are shown in Table 5.

The results show that rhetorical questions and sarcasm are still difficult for the model to capture. BERT tends to mispredict rhetorical questions and sarcasm. Since their actual meanings are usually the opposite of their literal meanings, the prediction becomes even more challenging. This observation is consistent with the statistical results of the significant effect of sarcasm and the negative effect, while not significant, of the rhetorical question. Hence, these complicated linguistic expressions are still challenging for the transformer-based model.

5.3 Automatically extracted linguistic features

5.3.1 Feature extraction

Multiple factors observed in error analysis (section 5.1) can be extracted automatically. These factors include number of entities, number of symbols, and transitions. Transitions are especially pertinent in sentiment classification, since they may suggest contrasting evaluations presented in impevals. Therefore, we further analyze the transitions with their dependency structures.

To begin with, each of the impevals was passed through the spaCy dependency parser (Honnibal et al., 2020) to provide syntactic information, which follows the Penn Treebank tag set. The tags and dependencies generated would later be utilized in extracting features. Additionally, dependency tree's maximum depth of each sentence was computed. It was used as an indicator of the complexity of the syntactic structure of the sentences. Finally, we used transition word lists in the literature to see if certain words influence the model performance.

Intending to provide a detailed observation of the impevals, we made efforts to create more variables. Aside from descriptive features such as the number of entities and special characters, tags and dependencies were utilized to build the features of transition. A transition

	Transition dependency 1	Transition dependency 2
Tag after the transition	CD, NN, NT, P, PN, VV, JJ	VC
Dependency after the transition	auxmod, advmod, case, compound:nn, dep, dobj, nsubj, nmod:tmod, nmod:range, ROOT	cop

Table 6: Details of the three transition features

can be found in words such as *dànshi* 'however' and *zhǐbùguò* 'no more than', indicating the change of the speaker's attitude. A transition word is often tagged with AD, which refers to an adverb, or CS, which refers to subordinating conjunction. Based on this constraint, two transition features were created by specifying the properties of the token right after the transition, as shown in Table 6.

We distinguished two transitional features with the help of dependency structures. Transition dependency 1 was defined by observing the actual impeval data. An example could be found in 遠傳至少還有 LM 可以在雙十一出來應戰 'at least Far EasTone Telecom still got LM (Line Mobile) to compete with others on Double Eleven Day'. In this example, the transition was *zhìshǎo* 'at least' with the tag 'AD'. The word next to it was *hái* 'still', with a tag 'NN' and the dependency 'dep'.

Another important transitional feature was the situation that some transitions were not followed by a noun but a copula *shì* 'is', Transition dependency 2 aimed to find out all the *shì* 'is' as a copula. It was found in 我家反而是亞太沒信號 'instead, Asia-Pacific has poor reception at my house'. The transition word here was *fǎnér* 'instead', and the word next to it was *shì* 'is', with a tag 'VC' and the dependency 'cop'.

In addition to transitional features constructed with dependency parsing, we also included 'transition with word list' feature based on Chang's (2018) research to see if the data contained any transition words. Another word list consisting of *chúle*, *chúfēi*, and *chúwài* was used to build the 'exception' construction.

To serve as a baseline, we additionally computed the model's prediction probabilities to indicate the model *confidence*. In past studies, it was shown deep learning models are not necessarily "well-calibrated" so that their confidences matched the actual correctness (Guo et al., 2017). Specifically, we evaluated the full impeval dataset with BertSimple and used the probabilities of the predicted class as the model's confidence.

5.3.2 Evaluation Results & Discussion

We include 8 features in the final logistic regression model. The statistical results are shown in Table 7. First, the significance coefficient of the model `confidence` shows that the model tends to be correct when it is confident. It is not always the case since past studies suggest deep learning models may not be confidence-calibrated (Guo et al., 2017). The effect indicates that although BertSimple is only trained on simple evaluation texts, it nevertheless learns, to an extent, the relation between evaluative language and its sentiment. It is just that impevals involve additional factors than what the model captures.

These additional factors are what we try to describe with linguistic features. From the statistical results, the number of entities and `exception` features significantly reduce the model's correctness. In contrast, the number of symbols and Transition dependency 1 slightly increase the model's correctness. One possible pattern that emerged from the results is that the model tends to be confused with local, focused features when these features potentially involve a "global revision" of the whole sentence's representation. This revision needs to occur when multiple entities occur, which may imply a comparison, and multiple evaluations need to compete for the evaluating entities. In the case of "exception", the prominent transition word signals an evaluation that could have complex relations with the main sentence. This observation is consistent with other positive effects of the number of symbols and Transition dependency 1. That is, as long as there's contextual information for the model to learn, the model could do better in these circumstances. This argu-

	Coeff	*SE*	*z*	*p*
Intercept	-3.22	0.72	-4.46	<0.0001
confidence	5.27	0.70	7.56	<0.0001
complexity	-0.08	0.08	-0.98	0.3271
No. entities	-0.43	0.13	-3.29	0.0010
No. symbols	0.36	0.17	2.08	0.0373
Exception	-2.08	0.68	-3.08	0.0021
Trans. words	-0.51	0.36	-1.40	0.1615
Trans. dep. 1	0.44	0.22	1.97	0.0493
Trans. dep. 2	-0.32	0.45	-0.71	0.4798

Table 7: Logistic regression analysis of all features

ment is also in line with recent studies that the transformer layers operate to help model mixing and capturing relations among input data (Lee-Thorp et al., 2021; Bronstein et al., 2021). Consistently and interestingly, the sentence complexity itself is not a problem for the model, as seen by the non-significant coefficient of the `tree depth` variable.

Finally, the contribution of linguistic features is independent of the model's prediction confidence. It is supported by the likelihood ratio test between two models: a base model, which only includes model confidence, and the full model described above. The likelihood ratio test is significant, $\chi^2(7) = 28.11$, $p < 0.0005$. The statistical result indicates that even though model confidence is highly correlated with the model's true correctness, the linguistic features complement the factors that the model is confused about.

6 Conclusion

Sentiment analysis has been an ongoing popular topic among the NLP community. This paper compiles an aspect-based annotated dataset consisting of social media comment threads about telecommunication services. To fully describe the versatility of evaluative texts, we distinguish between simple evaluation texts and implicit evaluation texts, *impevals*. As deep learning becomes more and more crucial in the field of computational linguistics, interpreting deep learning models is essential to understand how they come to a result and when they could possibly fail.

In particular, we focus on BERT, the current NLP state-of-the-art, and build two models: BertSimple and BertImpvl, to test BERT's knowledge about impevals. We find that the model learns a little about impevals, but not as well compared to simple evaluations. Hence, we conduct a series of qualitative and quantitative analyses on the factors that make impevals difficult. The overall results show that local features that require a global representation update confuse the model, such as multiple target entities, transitional words, sarcasm, and rhetorical questions. Consistently, the sentence complexity does not affect the model's correctness. It is also worthy to note although the model is predominantly trained with simple evaluations, the model confidence does reflect its correctness on impevals. However, the linguistic features complement the model confidence. That is, they together better explain when the model will be less accurate in each instance. To sum up, we expect that the linguistic evaluations of sentiment classification with BERT could help us understand more the characteristics of the model and when it might need more supervision with the help of linguistic feature analysis.

References

Arup Baruah, Kaushik Das, Ferdous Barbhuiya, and Kuntal Dey. 2020. Context-aware sarcasm detection using bert. In *Proceedings of the second workshop on figurative language processing*, pages 83–87.

Michael M Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. 2021. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*.

Li-Li Chang. 2018. The formation of the modal adverbs occurring frequently in adversative clauses. 成大中文學報, (63):191–230.

Leyang Cui, Sijie Cheng, Yu Wu, and Yue Zhang. 2020. Does BERT solve commonsense task via commonsense knowledge? *arXiv preprint arXiv:2008.03945*.

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*.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. In *International Conference on Machine Learning*, pages 1321–1330. PMLR.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength natural language processing in python.

Aditya Joshi, Pushpak Bhattacharyya, and Mark J Carman. 2017. Automatic sarcasm detection: A survey. *ACM Computing Surveys (CSUR)*, 50(5):1–22.

James Lee-Thorp, Joshua Ainslie, Ilya Eckstein, and Santiago Ontanon. 2021. Fnet: Mixing tokens with fourier transforms. *arXiv preprint arXiv:2105.03824*.

Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.

Diana Maynard and Mark Greenwood. 2014. Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 4238–4243, Reykjavik, Iceland. European Language Resources Association (ELRA).

Bo Pang and Lillian Lee. 2009. Opinion mining and sentiment analysis. *Comput. Linguist*, 35(2):311–312.

Kim Schouten and Flavius Frasincar. 2015. Survey on aspect-level sentiment analysis. *IEEE Transactions on Knowledge and Data Engineering*, 28(3):813–830.

Seungwan Seo, Czangyeob Kim, Haedong Kim, Kyounghyun Mo, and Pilsung Kang. 2020. Comparative study of deep learning-based sentiment classification. *IEEE Access*, 8:6861–6875.

Maxim Tkachenko, Mikhail Malyuk, Nikita Shevchenko, Andrey Holmanyuk, and Nikolai Liubimov. 2020-2021. Label Studio: Data labeling software. Open source software available from https://github.com/heartexlabs/label-studio.

Mikalai Tsytsarau and Themis Palpanas. 2012. Survey on mining subjective data on the web. *Data Mining and Knowledge Discovery*, 24(3):478–514.

Jie Zhou, Jimmy Xiangji Huang, Qin Chen, Qinmin Vivian Hu, Tingting Wang, and Liang He. 2019. Deep learning for aspect-level sentiment classification: survey, vision, and challenges. *IEEE access*, 7:78454–78483.

Yuan Zhuang and Ellen Riloff. 2020. Exploring the role of context to distinguish rhetorical and information-seeking questions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 306–312, Online. Association for Computational Linguistics.