Abstract

The rise in hateful and offensive language directed at other users is one of the adverse side effects of the increased use of social networking platforms. This could make it difficult for human moderators to review tagged comments filtered by classification systems. To help address this issue, we present the ViHOS (Vietnamese Hate and Offensive Spans) dataset, the first human-annotated corpus containing 26k spans on 11k comments. We also provide definitions of hateful and offensive spans in Vietnamese comments as well as detailed annotation guidelines. Besides, we conduct experiments with various state-of-the-art models. Specifically, XLM-R_large achieved the best F1-scores in Single span detection and All spans detection, while PhoBERT_large obtained the highest in Multiple spans detection. Finally, our error analysis demonstrates the difficulties in detecting specific types of spans in our data for future research. Our dataset is released on GitHub.

Disclaimer: This paper contains real comments that could be considered profane, offensive, or abusive.

1 Introduction

Social networking sites have been widely used all over the world. Here, users can easily share their thoughts, connect with others, or earn money by selling items, creating content, and so on. Since these sites are universally accepted, many extreme users misuse comment functions to abuse other individuals or parties with hate and offensive language. Consequently, it has been proved that these types of speech could harm other users’ health (Moham et al., 2017; Anjum et al., 2018). Sometimes these behaviors can be considered cyberbullying, cyber threats, or online harassment.

However, current studies are mainly about classifying comments as a whole with binary labels (Zampieri et al., 2019; Nguyen et al., 2021b) or multiple labeling schemes of abusive behaviors (Davidson et al., 2017; Founta et al., 2018; Mathur et al., 2018). These efforts are made to aid human moderators, who need to review a massive number of online tagged comments that violate their community standards. However, a system that can highlight the spans that make a comment hateful or offensive can be more advantageous to human moderators who frequently deal with long and tedious comments and prefer explanations over a system-generated unexplained tag per comment. Furthermore, in some cases, using highlighted spans and moderators’ context knowledge, they can take some actions to stop cyberbullying or online harassment. Nevertheless, there is only a study on toxic spans, SemEval-2021 Task 5: Toxic Spans Detection (Pavlopoulos et al., 2021). On the other hand, in a study of Mathew et al. (2020), hate and offensive spans worked as a rationale to support models in classifying the whole comments.

In Vietnamese, the resources about hate and offensive language are limited, namely ViHSD (Luu et al., 2021), HSD-VLSP (Vu et al., 2020), and UIT-ViCTSD (Nguyen et al., 2021b). Indeed, there
is no study about hate and offensive spans in Vietnamese. This motivated us to (i) develop a new task of extracting hate and offensive spans from Vietnamese social media texts that conceivably impact research and downstream applications and (ii) provide the Natural Language Processing (NLP) research community with a new dataset for recognizing hate and offensive spans in Vietnamese social media texts.

Our two main contributions are summarized:

1. We created the first human-annotated dataset for Vietnamese Hate and Offensive Spans (ViHOS) comprising 26,467 human-annotated spans on 11,056 comments. Our dataset is annotated with a clear definition of hate and offensive spans, along with detailed and specialized guidelines for a less-studied language like Vietnamese. Compared to the toxic spans dataset at SemEval-2021 Task 5 (Pavlopoulos et al., 2021), which is built to detect toxic spans from toxic comments, or the HateXplain dataset (Mathew et al., 2020), which has spans working as a rationale for classifying the whole sentence, ours includes not only a large number of texts with annotated hate and offensive spans but also clean texts without any spans. This effort is made to serve a new task of detecting hate and offensive spans from Vietnamese online social media comments.

2. To evaluate the efficacy of our dataset, strong baselines are empirically investigated on ViHOS, including BiLSTM-CRF (Lample et al., 2016), XLM-R (Conneau et al., 2019), and PhoBERT (Nguyen and Nguyen, 2020). We conducted various experiment schemas, including comparing the full dataset having additional clean comments with the dataset that does not have; Single span detection, Multiple spans detection, and All spans detection. We obtain that: (i) Additional clean comments help the baselines have better performance than the dataset without them for 10±2% (ii) After fine-tuning the deep learning model and pre-trained language models, results show that the pre-trained language models outperform the deep learning models.

2 Related work

To the best of our knowledge, much of the research in the field of hate speech detection has been conducted in English due to the abundance of corpora and the robust pre-trained models. Many benchmark datasets for hate and offensive speech in other languages have also been published in recent years, including Arabic (Mubarak et al., 2020), Dutch (Tulkens et al., 2016), and French (Chiril et al., 2019). Novel models are introduced to improve the efficiency of hate and offensive speech detection. Initial approaches were based on typical machine learning and deep neural networks with word embeddings. Transformer-based models such as BERT (Devlin et al., 2018), BERTology (Rogers et al., 2020), and BERT-based transfer learning (Ruder et al., 2019) have recently been used to detect hate and offensiveness that achieved competitive results in major SemEval shared tasks such as SemEval-2020 Task 12 (Zampieri et al., 2020), and SemEval-2021 Task 5 (Pavlopoulos et al., 2021). However, research in Vietnamese is still limited in terms of the dataset and experimental methods. Only a few outstanding research exist, such as ViHSD (Luu et al., 2021), HSD-VLSP (Vu et al., 2020), and UIT-ViTSD (Nguyen et al., 2021b).

For the topic of detecting foul spans, there are only a few case studies in English that are closely related, namely the SemEval-2021 Task 5: Toxic Spans Detection dataset (Pavlopoulos et al., 2021) and the HateXplain dataset (Mathew et al., 2020). The toxic spans, defined in the SemEval-2021 Task 5 dataset, are the sequences of words that make a text toxic. There are a total of 10,629 posts in this dataset, which stems from the Civil Comments dataset (Borkan et al., 2019). Another dataset with hate and offensive spans at the word level is HateXplain. The HateXplain contains 20,148 Gab and Twitter posts. Each post is manually classified into one of three labels: hateful, offensive, and normal.

In this study, we focus on Vietnamese to close the gap and develop the first Vietnamese hate and offensive spans detecting benchmark.

3 Dataset Creation

3.1 Dataset Source

ViHOS consists of 11,056 comments derived from the ViHSD dataset (Luu et al., 2021). The Vietnamese Hate Speech Detection dataset (ViHSD) is one of the few large and credible social media text datasets in a low-resource language like Vietnamese. ViHSD contains 27,624, 3,514, and 2,262 of CLEAN, HATE, and OFFENSIVE comments, respectively. Comments in ViHSD are public and collected from social media platforms.
Thus, metaphors, idioms, proverbs, and other tricky characteristics of online comments abound.

All of the HATE, OFFENSIVE comments from ViHSD after removing duplicates (5,528 comments left) are used to annotate the hate and offensive spans. Otherwise, 5,528 CLEAN comments, which also come from ViHSD and do not violate any hate or offensive definition defined in Section 3.2, are manually annotated for our dataset. We append the 5,528 CLEAN comments because: (*) We aim to detect the hate and offensive spans directly in online comments; (**) With an equal number of span and non-span (clean) comments helps models not be biased towards any type.

3.2 Annotation Guidelines

Our goal is to create a dataset that contains comprehensive hate and offensive thoughts, meanings, or opinions within the comments rather than just a lexicon of hate and offensive terms. We define the hate or offensive spans as follows to help annotators understand our goals:

- Harassing, cursing, insulting, disrespecting others.
- Sexual or verbal abuse towards one or a group of individuals based on their sensitive characteristics such as region, religion, politics, body, gender, etc.
- Insinuations, metaphors, metonymy used for hate, offensive or controversial purposes on sensitive issues such as region, gender, religion, politics, human rights, etc.
- Disuniting any factions or parties based on their politics, religion, ideologies, genders, etc.
- Causing verbal disrespect by using inappropriate pronouns.
- If replaced or removed, the sentence will no longer be hateful or offensive.

However, the hate or offensiveness in Vietnamese comments might cover one or even many components of a sentence. For instance: “thằng ad thở ra cái tư duy như trẻ lớp mầm” (Eng: the admin speaks as his mind just like a kindergarten boy.) This comment consists of three nouns/nouns phrases: "thằng ad" (it is offensive when calling a guy as a "thằng") as subject; "cái tư duy" (the appearance of the word "cái" causes this noun phase become offensive) and "trẻ lớp mầm" (kindergarten boy) as objects; one verb: "thở" (it usually means breathe, but in this context, we could consider it as speak but in a hate manner). As defined above, we must annotate a part of the subject, "thằng," and the whole phrase of the verb with its objects, "thở ra cái tư duy như trẻ lớp mầm" (Eng: speaks as his mind just like a kindergarten boy) in order to capture the whole hate or offensive ideas.

Therefore, we provided detailed guidelines (Figure 2) to assist annotators in determining when to annotate one or multiple components in a hateful and offensive sentence. As we observed, most of the comments in our dataset are colloquial. These comments are written freely and lack many grammatical rules. As a result, we frequently witness comments that lack subject(s), verb(s), conjunc-
Table 1: Inner-annotator agreement scores in three phases of annotation. In which, 1st, 2nd, 3rd, 4th are corresponding with four rounds of training annotators in Phase 1.

<table>
<thead>
<tr>
<th>Phase</th>
<th>1st</th>
<th>2nd</th>
<th>3rd</th>
<th>4th</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kappa score</td>
<td>0.4161</td>
<td>0.4568</td>
<td>0.4936</td>
<td>0.6402</td>
</tr>
<tr>
<td>F1-score</td>
<td>0.7085</td>
<td>0.7186</td>
<td>0.7534</td>
<td>0.8219</td>
</tr>
</tbody>
</table>

3.3 Dataset Construction Process

For dataset construction, we conducted three phases in which Phases 2 and 3 were inspired by Truong et al. (2021) and used metrics as bellow to calculate Inter-Annotator Agreement (IAA) among annotators. LightTag (Perry, 2021) is the tool we used for annotating data.

Assessment of Inter-Annotator Agreement
Cohen’s Kappa is widely used to measure inter-annotator agreement (IAA) in most tasks and is accepted as the standard measure (McHugh, 2012). However, numerous studies indicated that Kappa is not the most proper measure for the NLP sequence tagging task like NER (Hripcsak and Rothschild, 2005; Grouin et al., 2011). The reason is that the definite number of negative cases required to calculate the Kappa does not exist for named entity spans. Spans in our task are the sequences of characters rather than sequences of tokens since hate and offensive spans could be icon(s), word(s), or distinct character set(s) (see Table 8 for more details). Therefore, the pre-existing fixed number of characters to consider in the process of annotating is not existent.

A solution to deal with this is to calculate a character-level-based Kappa. Still, it has two associated problems: (1) annotators need to look at sequences of one or more characters instead of characters alone, causing the Kappa not to reflect the annotation task well; and (2) the "O"-labeled characters (the negative cases) outnumber the hate and offensive ones (the positive cases), provoking the Kappa to be computed on highly imbalanced data. For these reasons, the F1-score calculated without the negative cases is usually the measure for calculating IAA for the NLP tagging tasks like NER (Deleger et al., 2012). In this paper, IAA based on both F1-score (macro average) and character-level-based Kappa are calculated, while the former is the primary measure.

Phase 1: Pilot Annotation
Six undergraduate students were hired for our annotation tasks. The primary purpose of this pilot annotation phase was to familiarize our annotators with this task before entering the Main Annotation phase. We then developed an initial version of annotation guidelines with examples and distributed them to annotators. All annotators were required to carefully study the guidelines and give feedback before annotating the same 100 random samples from the 5,528 HATE, OFFENSIVE comments from ViHSD. This process was conducted four times with the F1-score and the Kappa for measuring IAA, which was calculated by averaging the results of pairwise comparisons across all annotators, shown in Table 1. All annotators were qualified as there was no F1-score of pairwise comparisons below 0.8.

Phase 2: Ground Truth Annotation
We randomly sampled a Ground Truth set of 600 comments from the 5,528 HATE, OFFENSIVE comments for this phase. Two guideline developers annotated the Ground Truth set separately using the well-developed guidelines from the former phase, resulting in an F1-score of 0.86 and Kappa (Cohen’s Kappa) of 0.72. Afterward, we hosted a discussion to deal with annotation conflicts and update the annotation guidelines.

Phase 3: Main Annotation
We split the remaining HATE and OFFENSIVE
Table 2: ViHOS statistics. Vocabularies size and comments length are calculated at the syllable level.

<table>
<thead>
<tr>
<th></th>
<th>Train</th>
<th>Dev</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of clean comments</td>
<td>4,552</td>
<td>569</td>
<td>575</td>
</tr>
<tr>
<td>Number of Ha/Off comments</td>
<td>4,422</td>
<td>553</td>
<td>553</td>
</tr>
<tr>
<td>Average clean comments length</td>
<td>8.69</td>
<td>8.50</td>
<td>9.04</td>
</tr>
<tr>
<td>Average Ha/Off comments length</td>
<td>16.81</td>
<td>17.68</td>
<td>16.13</td>
</tr>
<tr>
<td>Clean comments vocabulary size</td>
<td>4,234</td>
<td>1,423</td>
<td>1,400</td>
</tr>
<tr>
<td>Ha/Off comments vocabulary size</td>
<td>5,162</td>
<td>2,089</td>
<td>2,013</td>
</tr>
<tr>
<td>Number of multi-span comments (%)</td>
<td>2,322 (26.26)</td>
<td>308 (27.85)</td>
<td>296 (26.76)</td>
</tr>
<tr>
<td>Number of single-span comments (%)</td>
<td>1,970 (22.27)</td>
<td>229 (20.70)</td>
<td>235 (21.25)</td>
</tr>
<tr>
<td>Number of non-span comments (%)</td>
<td>4,552 (51.47)</td>
<td>569 (51.45)</td>
<td>575 (51.99)</td>
</tr>
<tr>
<td>Average number of spans</td>
<td>2.10</td>
<td>2.09</td>
<td>2.00</td>
</tr>
</tbody>
</table>

Table 3: Spans quantity and length statistics.

<table>
<thead>
<tr>
<th>Spans Quantity</th>
<th>Train</th>
<th>Dev</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 span (%)</td>
<td>4,522 (51.47)</td>
<td>569 (51.45)</td>
<td>575 (51.99)</td>
</tr>
<tr>
<td>1 span (%)</td>
<td>1,970 (22.27)</td>
<td>229 (20.70)</td>
<td>235 (21.25)</td>
</tr>
<tr>
<td>2 - 3 spans (%)</td>
<td>1,527 (17.27)</td>
<td>207 (18.72)</td>
<td>202 (18.26)</td>
</tr>
<tr>
<td>4 - 6 spans (%)</td>
<td>601 (6.80)</td>
<td>75 (6.78)</td>
<td>68 (6.15)</td>
</tr>
<tr>
<td>7 - 10 spans (%)</td>
<td>164 (1.85)</td>
<td>18 (1.63)</td>
<td>21 (1.90)</td>
</tr>
<tr>
<td>&gt;10 spans (%)</td>
<td>30 (0.34)</td>
<td>8 (0.72)</td>
<td>5 (0.45)</td>
</tr>
<tr>
<td>1 syllable (%)</td>
<td>5,253 (52.03)</td>
<td>699 (52.48)</td>
<td>647 (52.77)</td>
</tr>
<tr>
<td>2 - 3 syllables (%)</td>
<td>3,554 (35.20)</td>
<td>466 (34.98)</td>
<td>422 (35.68)</td>
</tr>
<tr>
<td>4 - 6 syllables (%)</td>
<td>916 (9.07)</td>
<td>122 (9.16)</td>
<td>112 (9.14)</td>
</tr>
<tr>
<td>7 - 10 syllables (%)</td>
<td>259 (2.57)</td>
<td>31 (2.33)</td>
<td>19 (1.55)</td>
</tr>
<tr>
<td>&gt;10 syllables (%)</td>
<td>114 (1.13)</td>
<td>14 (1.05)</td>
<td>14 (1.14)</td>
</tr>
</tbody>
</table>

4 Experiments and Results

4.1 Baseline Models

We treat the task of detecting hate and offensive spans as a task of sequence tagging. As a result, we make use of IOB format (Ramshaw and Marcus, 1995) to tag characters for model training, and testing. We conduct experiments on a set of solid baseline models, including BiLSTM-CRF and two pre-trained language models, XLM-R and PhoBERT, to assess the difficulty of our dataset.

**BiLSTM-CRF:** We use BiLSTM-CRF (Lample et al., 2016), a model that achieves high performance in the span detection tasks (Pavlopoulos et al., 2021; Nguyen et al., 2021a). We implemented this model with three main layers: (1) The word embedding layer using pre-trained PhoW2V (Nguyen et al., 2020), (2) The BiLSTM layer, and (3) the Conditional Random Field (CRF).

**XLM-R:** XLM-RoBERTa (Conneau et al., 2019) is a multilingual language model and a variant of RoBERTa, pre-trained on 2.5T of data across 100 languages containing 137GB of Vietnamese texts. On several cross-lingual benchmarks, XLM-R outperforms mBERT.
### 4.2 Experimental Settings

We empirically fine-tuned all pre-trained language models using *simpletransformers*\(^2\). For the tokenizer, each comment was tokenized using VnCoreNLP (Vu et al., 2018) in word-level and syllable-level for fine-tuning the PhoBERT and the XLM-R, respectively. In addition, we used Adam optimizer with a learning rate of 2e-5, a batch size of 8, and trained with 10 epochs.

We utilized a pre-trained word embedding - PhoW2V both syllable-level and word-level settings (Nguyen et al., 2020) with 100 dims to implement the BiLSTM-CRF model. The optimal hyper-parameters of BiLSTM-CRF are described in Table 6. All baseline models were trained on a system having 26GB RAM and an NVIDIA Tesla P100 GPU.

### 4.3 Evaluation Metrics

The macro-average F1-score (F1) is used to evaluate our models. For each pair of gold-predicted spans, we compute F1 and then calculate the arithmetic mean of F1 for each of these cases. It should be noted that the final F1-score, Accuracy, and Precision reported are an average of more than ten runs with various random seeds.

### 4.4 Experiments and Results

Table 4 reports the baseline results before and after adding the 5,528 additional clean comments. We discover that after the addition, the performances improve 0.1002 ± 0.0210. Specifically, PhoBERT\(_\text{Large}\) considerably outperforms other models in the dataset without additional clean data, achieving 0.6867 in F1-score. In addition, the best model trained on Full data is XLM-R\(_\text{Large}\), which has an F1-score of 0.7770. We find that XLM-R\(_\text{Large}\) increased by 0.1014 and PhoBERT\(_\text{Large}\) increased by 0.0849. These results demonstrate that the appearance of the additional clean comments successfully reduces model bias and improves performance.

Table 5 reports our results in three subsets corresponding to Single, Multiple, and All spans. Both Single and Multiple spans subsets are made by the process of splitting the All spans, which also known as the test set, based on the number of spans in each comment. Their results are described as follows:

- **Single span**: We experimented with both

---

\(^2\)https://simpletransformers.ai/ (ver.0.63.3)
syllable-level and word-level language models. We discover that the pre-trained language models outperform the BiLSTM-CRF model by 0.3521 ± 0.0099 in F1. This significant gap proves the fact that word embedding and features extraction of the pre-trained language models on the social media texts are superior to the BiLSTM-CRF. The XLM-R_{Large} model achieves the best performance with a 0.7214 in F1-score. On the other hand, PhoBERT_{Large} achieves a 0.7067 in F1-score. These results show no significant difference in performances among the multilingual and monolingual pre-trained models in the Single span.

Multiple spans: We experimented with the syllable-level and word-level and found that the pre-trained language models beat the BiLSTM-CRF model by 0.3212 ± 0.0132. In addition, the performance of XLM-R_{Large} is slightly better than the PhoBERT_{Large} by 0.0036 in F1-score. The results on the Multiple spans are always better than the Single span, which might be explained by the fact that data in Multiple spans comprise more hate and offensive spans that can assist the models in learning more features of the data.

All spans: The results of the experiments on the All data are higher than the Single span and the Multiple spans. Specifically, in terms of F1-score, results of the XLM-R_{Large} model are higher by 0.0556 and 0.0413 than the highest in the Single span and the Multiple spans, respectively while the figures for the PhoBERT_{Large} are 0.0649 and 0.0395, respectively.

4.5 Results Analysis

We choose two best models: PhoBERT_{Large} and XLM-R_{Large} to conduct error analysis. As shown in Figure 3, we report the statistics of the ratio of various types of error cases of 100 random samples in the dev set. We notice that wrong spans prediction, wrong spans boundary, allusion, annotation error, abbreviation, spelling mistake, needing domain knowledge, lack of context, idiom or proverb, missing diacritical marks, metaphor or metonymy, and others (rare characters, mixing other languages, all words stick together, etc.). These error cases are defined in Appendix C.

Figure 3: Error analysis conducted on prediction on dev set made by PhoBERT_{Large} and XLM-R_{Large}. We divide error cases into 12 categories including wrong spans detection, wrong spans boundary, allusion, annotation error, abbreviation, spelling mistake, needing domain knowledge, lack of context, idiom or proverb, missing diacritical marks, metaphor or metonymy, and others (rare characters, mixing other languages, all words stick together, etc.).
people. In the third instance, the phrase "cướp đêm là giặc, cướp ngày là quan," (Eng: night thieves are enemies, day thieves are bureaucrats) which is an idiom that originated from folk poetry, also misleads the prediction. In the final example, the verb "cào" has no object and must be comprehended in context. These intriguing and challenging linguistic phenomena encourage more research into more robust models and methods in this field.

5 Conclusion and Future Work

We presented ViHOS, a new Vietnamese dataset for evaluating hate and offensive spans detection models. ViHOS includes 26,467 human-annotated spans on 11,056 comments. In addition, state-of-the-art models are conducted as the first baseline models, including BiLSTM-CRF and pre-trained language models such as XLM-R\textsubscript{Base}, XLM-R\textsubscript{Large}, PhoBERT\textsubscript{Base}, and PhoBERT\textsubscript{Large}. As a result, the XLM-R\textsubscript{Large} model achieves the best performance, with an F1-score of 0.7770. Furthermore, we discover that the performance when detecting multiple spans is better than the performance in detecting single spans in Vietnamese hate and offensive spans detection. Our dataset is available publicly at the GitHub link\footnote{https://github.com/phusroyal/ViHOS}.

Despite the study’s many promising contributions, the proposed research work still has several potential concerns, especially since the performance is still modest, and incorrect predictions could harm users’ reputations if they rely heavily on our method. We intend to expand the dataset size and diversity of hate and offensive context for Vietnamese in the future to address this shortcoming. Furthermore, pre-and post-processing techniques will be used to standardize social networking texts (Clark and Araki, 2011) and deal with complex cases (as discussed in Subsection 4.5) to improve model performance (Suman and Jain, 2021; Kotyushov et al., 2021; Chhablani et al., 2021), particularly for Vietnamese pre-trained language models.
Limitations, Social Impacts, and Ethical Considerations

Limitations and Social Impacts

There are numerous incomprehensible comments in our dataset due to the lack of context. Consequently, our annotators had to place themselves in imaginary contexts in order to annotate those comments (see Table 9 for more details about our solution). This shortcoming combined with the limitations of the neural networks in terms of understanding various linguistic phenomena (see Table 8 for more details about nine different linguistic phenomena) caused their performances of this task still insufficient to become practical.

We also acknowledge the risk associated with publicizing a dataset of hate and offensive spans (e.g. utilizing ours as a source for building abusive chatbots). However, we firmly believe that our proposed benchmark creates more value than risks.

Ethical Considerations

The undergraduate students in the annotation process are Vietnamese native speakers; have at least 12 years of studying Vietnamese with average scores on the Vietnam National Exam on Literature of 6.5; have at least three years of using social network platforms. They were explicitly warned that their tasks will display hateful and offensive content and if they became overwhelmed, they were also urged to stop labeling. These undergraduate students were paid $0.1 per comment, which takes an average of 6.44 seconds to complete (excluding the time used by workers who took exceptionally lengthy comments).

All the comments in ViHOS originated in the study of Luu et al., 2021, which preserved users’ anonymity by removing all of them when creating the ViHSD. As a result, the comments in our dataset do NOT reflect our thoughts or viewpoints. ViHOS is available to the public under a usage agreement for research and related purposes only.

Acknowledgments

This work has been funded by The VNUHCM-University of Information Technology’s Scientific Research Support Fund.

References


## A Abusive Language Characteristics

Table 8: Characteristics of abusive language in ViHOS with examples, explanations, and solutions for annotators.

<table>
<thead>
<tr>
<th>Abusive language characteristics</th>
<th>Examples</th>
<th>Explanations and Solutions</th>
</tr>
</thead>
</table>
| Non-diacritical marks comments   | (1) Dit me cai quy trinh, vao cap cuu deo co tien, deo bao hiem thi nam do di.  
(Eng: *Fuck the procedure, without money or insurance, you could just lay there and no one cares about your emergency*)

(2) Dung la con linh dien dien vua tho chang qua nt goi de choc my dien tho "con ng" nhu linh dien thi ai them  
(Eng: *That must be the Crazy Linh, so crazy! I just call to tease the Crazy My. No one gonna love Crazy/The person like Crazy Linh!*) | Explanation: some of these non-diacritical marks comments might trick annotators a little bit.  
(1): this is a non-diacritical marks comment but still able to understand.  
(2): there are some problems causing annotators to re-read multiple times as no punctuation, diacritic, and the text "con ng" could be considered as "crazy girl" or "the type (of human)" and both of these meanings is inappropriate.  
Solution: Non-diacritical marks comments are annotated as others. Annotators have to re-read until they fully understand the texts if needed. Those examples are annotated as follows:  
(1): ["Dit me", "deo", "deo"]  
(Eng: ["Fuck", "fuck", "fuck"])  
(2): ["con", "dien", "dien", "con ng", "dien", "ai them"]  
(Eng: ["con", "crazy", "crazy", "crazy/the person like", "crazy", "No one gonna love"] in which the word "con" is an inappropriate way to call a woman.) |
<table>
<thead>
<tr>
<th>Metaphors, metonymies</th>
<th>Explanation: in our dataset, many comments use metaphors or metonymy to convey their hate or offensiveness in another way. (1): this is a metaphor of a mouth as a vagina. (2): this is a metonymy of Juventus’ jersey as prison shirts (a common metonymy in Vietnamese).</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) cái miệng rộng quá đẻ con ra còn lọt</td>
<td>(Eng: The mouth is too big that a baby could even be born through it)</td>
</tr>
<tr>
<td>(2) Dm! qua đợt dịch này thì thằng sống ích kỷ này chắc sớm gia nhập juventuts</td>
<td>(Eng: Fuck it! After the pandemic, this selfish boy will soon join Juventuts)</td>
</tr>
<tr>
<td><strong>Explanation:</strong> in our dataset, many comments use metaphors or metonymy to convey their hate or offensiveness in another way. (1): this is a metaphor of a mouth as a vagina. (2): this is a metonymy of Juventus’ jersey as prison shirts (a common metonymy in Vietnamese).</td>
<td><strong>Solution:</strong> Annotators are required to annotate the whole ideas of metaphors, metonymy. (1): [&quot;cái miệng rộng quá đẻ con ra còn lọt&quot;] (2): [&quot;Dm&quot;, &quot;thằng sống ích kỷ này chắc sớm gia nhập juventuts&quot;] (Eng: [&quot;Fuck&quot;, &quot;this selfish boy will soon join Juventuts&quot;])</td>
</tr>
<tr>
<td>Puns</td>
<td><strong>Explanation:</strong> Some comments use phrases that only read them backwards, they make sense. As in the example, &quot;Bồn Kỳ Lắc&quot;, if this phrase is read backwards, it is &quot;Bắc Kỳ Lồn&quot; (pussy north).</td>
</tr>
<tr>
<td>(1) Ad đăng bài này cũng là Bồn Kỳ Lắc nè nè nè :)</td>
<td>(Eng: The admin, who posts this, is also a &quot;Bồn Kỳ Lắc&quot;)</td>
</tr>
<tr>
<td><strong>Solution:</strong> Annotators are required to annotate these puns too. (1): [&quot;Bồn Kỳ Lắc&quot;]</td>
<td></td>
</tr>
<tr>
<td><strong>Explanation:</strong> Some comments use phrases that only read them backwards, they make sense. As in the example, &quot;Bồn Kỳ Lắc&quot;, if this phrase is read backwards, it is &quot;Bắc Kỳ Lồn&quot; (pussy north).</td>
<td><strong>Solution:</strong> Annotators are required to annotate these puns too. (1): [&quot;Bồn Kỳ Lắc&quot;]</td>
</tr>
<tr>
<td>Using non-words characters to form hieroglyphs</td>
<td><strong>Explanation:</strong> (1): the <em>knife symbol</em> is used instead of the word knife. (2): the existence of &quot;()&quot; can be considered as pussy in this context.</td>
</tr>
<tr>
<td>(1) có tin t lấy <em>knife symbol</em> xiên chết cụ m ko :))</td>
<td>(Eng: Do you belive that I could get a <em>knife symbol</em> to fucking kill you?)</td>
</tr>
<tr>
<td>(2) Đâm vào ()</td>
<td>(Eng: stab in the ()</td>
</tr>
<tr>
<td><strong>Solution:</strong> Annotating the non-words only if they can convey a full meaning of hate or offensiveness, and the whole phase if they can not. (1): [&quot;t&quot;, &quot;<em>knife symbol</em>&quot;, &quot;xiên chết cụ m&quot;] (Eng: [&quot;t&quot;, &quot;<em>knife symbol</em>&quot;, &quot;fucking kill you&quot;]) in which &quot;t&quot;, &quot;m&quot; are inappropriate pronouns.) (2): [&quot;Đâm vào ()&quot;]</td>
<td></td>
</tr>
</tbody>
</table>
Table 8 continued from the previous page.

Spelling mistakes

(1) Tôi đeo hiểu bạn noi cai gì luôn a :)))
(Eng: I have no fucking idea about what you saying)

(2) Mà dx cái chửi ngta nghe hài vcl bù lại cùng đỡ
(Eng: Those fucking curses is so funny that can even refill that)

(3) Khi bạn xai trình tã nhưng cá ghén vít đúng trình tã :))
(Eng: When you make spelling mistakes but trying to fix it :))

Explanation:
(1): the words "đeo," "nói," "cai," "a" are spelling mistakes. The words can be understood as "déo," "nói," "á". As that, these words are the same in accidentally missing acute marks.

(2): the words: "chữi," "cùng," "dở" are spelling mistakes is a phenomenon of mistaking tild mark for hook above mark, and this often happens in some parts of Vietnam (Nguyen Hoai Nguyen, 2010). There are also familiar phenomena of mistaking marks such as tild mark for underdot mark, acute mark for hook above mark, and so on (Nguyen Hoai Nguyen, 2010).

(3): the phases: "xai trình tã," "cả ghén," "vít đúng trình tã" are spelling mistakes but on purpose. This comment utilizes spelling mistakes to attack opponents who also have spelling mistakes. Furthermore, these spelling mistakes also abuse opponents based on regional distinctions in accent, which cause some phenomenon of mistaking diacritical marks as in Example (2).

Solution: The same as dealing with non-diacritical marks comments, annotators work as usual.

(1): ["deo"]
(Eng: ["fuck"])  
(2): ["chữi", "vcl"]
(Eng: ["curse", "fuck"])  
(3): ["bạn xai trình tã nhưng cá ghén vít đúng trình tã :))"]
### Table 8 continued from the previous page.

**Eng:** If it had something, it was done! It is just like the way CS (stands for Communism) chooses people, beauty over the profession. Detoxifying Communism is hard; detoxifying the left-wing political ideologies is even harder.  
(2) Rút kinh nghiệm lại được xài nghìn tỷ  
**Eng:** Just say learned and then can use trillion VND |
|------------------|---------------------------------------------------|
| **Solution:** Annotators are required to annotate the whole profound abuse.  
(1): ["toi", "hồng hơn chuyên", "Giải độc cộng sản đã khó, giải độc tư tưởng cánh tả còn khó hơn"]  
(Eng: ["die", "beauty over the profession", "Detoxifying Communism is hard; detoxify the left-wing political ideologies even harder"])  
(2): ["Rút kinh nghiệm lại được xài nghìn tỷ"] |
| Homonym | (1) coin card  
("coin card" is a homonym of "con cặc", which means dick) |
| **Solution:** annotate the whole abusive homonym.  
(1): ["coin card"] |
| Mixing languages | (1) Vl fake  
**Eng:** Fuck! Fake)  
(2) phe X compat tổng Y =))  
**Eng:** X’s side combats Y’s side)  
(3) Tôi Thầy stream đá phò đi thầy ơi  
**Eng:** You should stream fucking some whores tonight, please! |
| **Solution:** treat foreign words as others and annotate if they meet the definition of hate and offensive spans.  
(1): ["Vl", "fake"]  
(Eng: ["fuck", "fake"])  
(2): ["compat"]  
(Eng: ["combats"])  
(3): ["đá phò"]  
(Eng: ["fuck some whores"]) |
Trick hate speech detecting systems on purpose

<table>
<thead>
<tr>
<th>No.</th>
<th>Example</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>Đấy Ông già xạo l*n của các bạn được tắm bồn Thái Lan đấy=))) (Eng: See, your fucking old liar is Thai bathing again =)))</td>
<td>Some hate or offensive comments use punctuation to censor their inappropriate words (as in (1), asterisk is used in the word l*n (pussy)) or to disunite characters in words (as in (2), dot and space are used to disunite the words chó (dog) into c h.ó). These efforts actually can trick many hate speech detecting systems, or to put all words together (as in (3), vualonemlabonhieu cm should be &quot;vừa lồn em là bao nhiêu cm&quot;).</td>
</tr>
<tr>
<td>(2)</td>
<td>thằng c h.ó ngu (Eng: Fucking stupid guy)</td>
<td></td>
</tr>
<tr>
<td>(3)</td>
<td>vualonemlabonhieu cm (Eng: How long in cm to fit your pussy?)</td>
<td></td>
</tr>
</tbody>
</table>

**B Notices for annotators**

Table 9: Additional notices for annotators.

<table>
<thead>
<tr>
<th>Notices</th>
<th>Example</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Try to figure out and consider as in the original context of the comments.</td>
<td></td>
<td>This could help annotators understand complex and non-context hate, offensive comments.</td>
</tr>
<tr>
<td>Do not let emotion affect the annotating process.</td>
<td></td>
<td>Annotators exposed in a long time to toxic comments are reported to get used to the frequently appearing hate, offensiveness.</td>
</tr>
<tr>
<td>Check the provided Vietnamese Dictionary if there is any uncertainty in being sure a word is hate or offensive.</td>
<td></td>
<td>We use the most reputable Vietnamese dictionary (Hoang Phe, 1983) to provide to annotators.</td>
</tr>
<tr>
<td>Should span the object is compared to in an inappropriate comparison.</td>
<td>“Ăn cơm nhìn như chó” (Eng: You eat like a dog) Spans: [&quot;nhìn như chó&quot;]</td>
<td>We consider this is an inappropriate comparison in which annotators must span &quot;chó&quot; dog.</td>
</tr>
</tbody>
</table>
Table 9 continued from the previous page.

<table>
<thead>
<tr>
<th>Description</th>
<th>Example</th>
<th>Annotation</th>
</tr>
</thead>
<tbody>
<tr>
<td>However, we should span the whole comparison if spanning only the object is compared to might not convey the complete hate or offensive idea.</td>
<td>&quot;ý thức như trẻ lớp mầm&quot; (Eng: Your awareness is just like a kindergarten kid)</td>
<td>If we only span &quot;trẻ lớp mầm&quot; (kindergarten kid), it will not convey the complete offensive idea. As that, annotators are encouraged to span the whole text instead.</td>
</tr>
<tr>
<td>Do not span conjunctions; exceptional Vietnamese cases; standard ways to call LGBTQ+.</td>
<td>(1) Vì thế, nên, nhưng, mà, etc. (Eng: so, so, but, but, etc.) (2) Gay, les, etc (3) Bống, bể đẻ, ái nam ái nữ, etc.</td>
<td>(1) Some conjunctions in Vietnamese. (2) Appropriate ways to specify LGBTQ+. (3) Inappropriate ways to specify LGBTQ+ that need to be spanned in comments.</td>
</tr>
<tr>
<td>Blatant hate and offensive words prioritize being spanned over the others, especially in sentences without diacritics and could not be understood.</td>
<td>Muy Cha H O an Tro. Cap. Ranh c. Di Ngoại le. Nhieu chuyen. Do Cai Thu, do tam than</td>
<td>Similar to this non-diacritical and incomprehensible comment, words as highlighted ones are more straightforward to be spanned.</td>
</tr>
<tr>
<td>Span the whole phase violating human rights.</td>
<td>&quot;Về nước anh nên vào tù ở trước thay vì di cách lý.&quot; (Eng: When you return to Vietnam, you should go to jail first instead of going to isolation)</td>
<td>Comments violate human rights, usually complex to specify hate, offensive words to span. As in this example, a Vietnamese citizen comes back from a foreign country has a right to have isolated healthcare firstly.</td>
</tr>
<tr>
<td>Span the whole obscene acronyms.</td>
<td>(1) clgv (Eng: wtf is that?) (2) clmn (Eng: your mom’s pussy) (3) cmn (Eng: your mother).</td>
<td>Some comments have strings being constructed by many words missing diacritical marks, but still able to understand. The annotators should only span the hate and offensive characters set out of the string as in the example.</td>
</tr>
<tr>
<td>Follow the rule with words that do not have diacritics and conjoin to span out the hate, offensive spans.</td>
<td>cailongithe (Eng: wtf is that?)</td>
<td></td>
</tr>
<tr>
<td>Span hate, offensiveness separately.</td>
<td>&quot;Cả đám dlv là nhóm ngu dốt, trẻ trâu potay vcl luôn.&quot; (Eng: The whole dlv crew are stupid, bull-headed kids (so I have) no fucking thing to say.)</td>
<td>This comment must be spanned as in the example, but not as &quot;đám,&quot; &quot;ngu dốt, trẻ trâu,&quot; &quot;vcl,&quot; &quot;NGU.&quot;</td>
</tr>
</tbody>
</table>

668
C Definition of Error Cases for Error Analysis

We introduce 12 error definitions as follows:

1. **Wrong spans prediction**: The model predicts clean spans as hateful and offensive.

2. **Wrong spans boundary**: The model predicts inadequate boundary or fails to predict correctly.

3. **Allusion**: The comment refers to another person or subject in an indirect and disrespectful way.

4. **Annotation error**: Annotators have improperly annotated the span. The reason might be that they somehow do not follow the provided guidelines. However, there is no modification in the final dataset.

5. **Abbreviation**: The comment contain short forms of words.

6. **Spelling mistake**: The comment is spelling mistake.

7. **Needing domain knowledge**: Dialect and professional expertise are required to detect span in comments.

8. **Lack of context**: In different contexts, the comment could be understand in multiple meaning.

9. **Idiom or proverb**: The comment contains idiom or proverb.

10. **Missing diacritical marks**: Words in the comment do not have diacritical marks.

11. **Metaphor or metonymy**: The comment contains metaphor or metonymy.

12. **Others**: The comment contains rare characters, other languages, words in it are stick together, etc.