Euphemistic Phrase Detection by Masked Language Model

Wanzheng Zhu and **Suma Bhat**
University of Illinois at Urbana-Champaign, USA
wz6@illinois.edu, spbhat2@illinois.edu

Abstract

It is a well-known approach for fringe groups and organizations to use *euphemisms*—ordinary-sounding and innocent-looking words with a secret meaning—to conceal what they are discussing. For instance, drug dealers often use "pot" for marijuana and "avocado" for heroin. From a social media content moderation perspective, though recent advances in NLP have enabled the automatic detection of such *single-word* euphemisms, no existing work is capable of automatically detecting *multi-word* euphemisms, such as "blue dream" (marijuana) and "black tar" (heroin). Our paper tackles the problem of **euphemistic phrase detection** without human effort for the first time, as far as we are aware. We first perform phrase mining on a raw text corpus (e.g., social media posts) to extract quality phrases. Then, we utilize word embedding similarities to select a set of euphemistic phrase candidates. Finally, we rank those candidates by a masked language model—SpanBERT. Compared to strong baselines, we report 20-50% higher detection accuracies using our algorithm for detecting euphemistic phrases.

1 Introduction

Euphemisms—ordinary-sounding and innocent-looking words—have long been used in human communication as an instrument to conceal secret information (Bellman, 1981). A primary motive of their use on social media is to evade automatic content moderation efforts enforced by such platforms (Cambridge Consultants, 2019; Yuan et al., 2018). For example, a rich lexicon of drug euphemisms has evolved over time, with entire communities subscribing to benign sounding words that allude to drug names (e.g., {"popcorn", "blueberry", "green crack", "blue dream"} $\longrightarrow$ "marijuana", {"coke", "white horse", "happy powder"} $\longrightarrow$ "cocaine").

Research on automatic euphemism detection has recently received increased attention in the natural language processing communities (Durrett et al., 2017; Magu and Luo, 2018; Pei et al., 2019; Felt and Riloff, 2020), and the security and privacy communities (Zhao et al., 2016; Yang et al., 2017; Yuan et al., 2018; Hada et al., 2020; Zhu et al., 2021). However, existing approaches can only detect *single-word* euphemisms (e.g., "popcorn", "coke"), and fail to detect *multi-word* euphemisms (e.g., "black tar", "cbd oil") automatically. Therefore, offenders can simply invent euphemistic phrases to evade content moderation and thwart censorship.

Our paper focuses on the task of **euphemistic phrase detection**—detecting phrases that are used as euphemisms for a list of target keywords—by extending the state-of-the-art single-word euphemism detection algorithm proposed by Zhu et al. (2021). Our proposed approach first mines quality phrases from the text corpus using AutoPhrase (Shang et al., 2018; Liu et al., 2015), a data-driven phrase mining tool. Then, it filters noisy candidates that are not semantically related to any of the target keywords (e.g., heroin, marijuana in the drug category). This serves as a pre-selection step to construct a euphemistic phrase candidate pool. Finally, we rank the pre-selected candidates using SpanBERT (Joshi et al., 2020), a pre-training Masked Language Model (MLM) that is designed to better predict the span of tokens (i.e., phrases) in text.

Evaluating on the benchmark drug dataset in Zhu et al. (2021), we find that our proposed approach yields euphemistic phrase detection results that are 20-50% higher than a set of strong baseline methods. A qualitative analysis reveals that our approach also discovers correct euphemisms that were not on our ground truth list, i.e., it can detect previously unknown euphemisms and even new types of drugs. This is of significant utility in the context of Internet communities, where euphemisms evolve rapidly and new types of drugs may be invented.

Findings of the Association for Computational Linguistics: EMNLP 2021, pages 163–168
November 7–11, 2021.

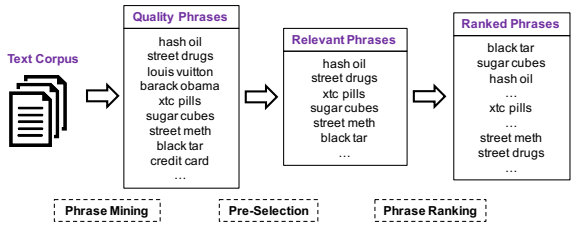

Figure 1: An overview of our proposed framework

2 Proposed Model

In this study, we assume access to a raw text corpus (e.g., a set of posts from an online forum). In practice, forum users may use *euphemisms*—words that are used as substitutes for one of the target keywords (e.g., heroin, marijuana). We aim to learn which multi-word *phrases* are being used as euphemisms for the target keywords. The euphemism detection task takes as input (1) the raw text corpus and (2) a list of target keywords. The output is an ordered ranked list of euphemistic phrase candidates, sorted by model confidence.

Our proposed approach for euphemistic phrase detection has three stages (shown in Figure 1): 1) Mining quality phrases, 2) Pre-selecting euphemistic phrase candidates using cosine similarities of word2vec embeddings (Mikolov et al., 2013a,b), and 3) Ranking euphemistic phrases with a masked language model.

2.1 Quality Phrase Mining

Phrase mining aims to generate a list of quality phrases, which serves as the candidate pool for the algorithm to rank. We select AutoPhrase (Shang et al., 2018; Liu et al., 2015), which has demonstrated superior phrase mining performance in a wide range of settings, to mine quality phrases. This is because we are interested in a data-driven method of detection from a domain-specific text corpus such as subreddit[1], rather than by using trained linguistic analyzers (e.g., dependency parsers) that are less likely to have a satisfactory performance on text corpora with unusual usage of words (euphemisms). By incorporating distant supervision (i.e., Wikipedia) and part-of-speech tags as Shang et al. (2018), we empirically find that AutoPhrase can extract meaningful phrases successfully.

[1]Forums hosted on the Reddit website, and associated with a specific topic.

2.2 Pre-Selection of Phrase Candidates

AutoPhrase takes only a text corpus as its input and produces phrases that may or may not be relevant to any of the target keywords. This stage aims to filter out phrases that are not relevant to the target keywords and thus pre-select the euphemistic phrase candidates. This serves to not only pre-filter noisy candidates, but also to reduce the computational resources in the subsequent ranking algorithm.

Specifically, we use the word2vec algorithm (Mikolov et al., 2013a,b) to learn the embeddings for all the words and phrases.[2] Relying on the distributional hypothesis that semantically similar words occur in linguistically similar contexts, we assume that the euphemistic phrases should not be too far from the target keywords on the embedding space. Therefore, we select the top k phrases[3] in terms of the cosine similarities between the embeddings of each extracted phrase and the average embeddings of all target keywords.

2.3 Euphemistic Phrase Ranking

We extract contextual information of the target keywords and filter out uninformative contexts, following Zhu et al. (2021). Next, with a collection of informative masked sentences (e.g., "This 22 year old former [MASK] addict who I did drugs with was caught this night"), we aim to rank the pre-selected phrase candidates for their ability to serve as a replacement of the masked keyword. Toward ranking the candidates for filling in the mask, a common approach is to use BERT (Devlin et al., 2019), but BERT can be used to only rank single words. Here, we leverage the idea of masked language model applied not at the word level, but at the phrase level to facilitate detection. Therefore, we select SpanBERT (Joshi et al., 2020) to rank the candidates, because it is designed to better represent and predict contiguous spans of text and it enables the likelihood calculation of multi-word candidates in a given context.

We fine-tune the pre-trained SpanBERT model with the text corpus of interest.[4] Then, for each masked sentence m, and for each phrase candidate c, we compute its MLM probability (the proba-

[2]We use the Gensim package in Python3 for word2vec training. We use a context window of 6, an embedding dimension of 100, a minimum count of 5, and a sampling rate of 10^{-4}.

[3]We empirically set $k = 1000$ in our experiments.

[4]https://github.com/facebookresearch/SpanBERT.

bility of the phrase c occurring in m as predicted by the masked language model) $h_{c,m}$ by the fine-tuned SpanBERT model. Therefore, given a set of masked sentences, the weight w_c of a word candidate c is calculated as: $w_c = \sum_{m'} h_{c,m'}$. Lastly, we rank the phrase candidates by their weights.

3 Empirical Evaluation

We evaluate our proposed model (denoted as "EPD") and the following baselines on the benchmark *drug* dataset in Zhu et al. (2021), and compare it with the following baseline models:

- **SentEuph** (Felt and Riloff, 2020) recognizes euphemisms by sentiment analysis and a bootstrapping algorithm for semantic lexicon induction. For a fair comparison, we do not include its manual filtering stage and exclude the single-word predictions from the output.
- **Word2vec**: we follow Section 2.1 and 2.2 to rank all phrases by the cosine similarities between each phrase and the input target keywords. We do not include the final euphemistic phrase ranking step in Section 2.3. This is one of the most straightforward baselines and also, an ablation study to investigate the effectiveness of the euphemistic phrase ranking step.
- **EigenEuph** (Magu and Luo, 2018) leverages word and phrase embeddings (following Section 2.1 and 2.2) and a community detection algorithm, to generate a cluster of euphemisms by the ranking metric of eigenvector centralities.
- **EPD-rank-all** is a simpler version of EPD. It does not pre-select euphemistic phrase candidates described in Section 2.2 but uses SpanBERT to rank *all* phrases mined by AutoPhrase.
- **EPD-ILM** ranks the pre-selected phrase candidates by Infilling by Language Modeling (ILM)[5] (Donahue et al., 2020) instead of SpanBERT. ILM is optimized for predicting fixed-length missing tokens of a document. We set the token length to be 2, since a majority of euphemistic phrases (i.e., 749 out of 820 in the drug dataset) have 2 words.

Following Zhu et al. (2021), we use the evaluation metric **precision at k** ($P@k$) to compare the generated candidates of each method with the ground truth list of euphemistic phrases. For a fair comparison of the baselines, we experiment with different combinations of parameters and report the best performance for each baseline method.

[5] https://github.com/chrisdonahue/ilm

	$P@10$	$P@20$	$P@30$	$P@50$
SentEuph	0.00	0.00	0.03	0.02
Word2vec	0.10	0.10	0.07	0.06
EigenEuph	0.10	0.15	0.13	0.10
EPD-rank-all	0.20	0.25	0.20	0.16
EPD-ILM	0.00	0.10	0.10	0.12
EPD	**0.30**	**0.30**	**0.27**	**0.22**

Table 1: Results on euphemistic phrase detection. Best results are in bold.

Euphemistic Phrase Candidates
black tar, nitric oxide, nitrous oxide, **hash oil**, citric acid, crystal meth, lysergic acid, hydrochloric acid, **cbd oil**, magic mushroom, sour diesel, xtc pills, crystal meth, isopropyl alcohol, **sugar cubes**, speed paste, **og kush**, fentanyl powder, **brown sugar**, pot brownies, xanax bars, hemp oil, **coca cola**, dnm coke, co2 oil, **blue dream**, gold bullion, cannabis tincture, oxy pills, amphetamine powder

Table 2: Top 30 output by EPD. Purple bold words are correct detections as marked by the ground truth list.

3.1 Results

Table 1 summarizes the euphemistic phrase detection results. We note that our proposed approach outperforms all the baselines by a wide margin for the different settings of the evaluation metric.

SentEuph's poor performance could be attributed to the absence of the required additional manual filtering stage to refine the results. As mentioned before, this was done to compare the approaches based on their automatic performance alone.

Word2vec is one of the most straightforward baselines. By taking advantage of the distributional hypothesis, it can output some reasonable results. However, its performance is still inferior largely because it learns a single embedding for each token and therefore does not distinguish different senses of the same token. EigenEuph, which leverages a community detection algorithm to enhance the similarity for different tokens, has slightly better results than the vanilla Word2vec baseline.

By comparing the performance of EPD and Word2vec, we conclude that it is effective to adopt SpanBERT for the final ranking of the pre-selected euphemistic phrase candidates. Comparing the performance of EPD and EPD-rank-all, we demonstrate that it is effective to pre-select a set of eu-

ID	Euphemism Candidates	Sentences Associated
1	nitrous oxide	• really incredible short lasting high nitrous oxide is so much more effective while on mdma • ive done multiple other drugs and im going to try nitrous oxide for the first time • so i have done a few different substances so far including weed mdma acid nitrous oxide and ketamine
2	sour diesel	• he likes the stupidly pungent sour diesels and kushs and its all he smokes • vendor sftreats product oz of sour diesel price $280 okay let me start off by saying holy shit • us vendor sour diesel green crack
3	speed paste	• i bought cheap cocaine and cheap speed paste • ordered 3g speed paste from an onion patch dried it out and tried a small bomb and a couple bumps ~60mg total • iv dosage of 70 pure speed paste
4	magic mushroom	• lsd or magic mushrooms solo trip • ive done magic mushrooms 3 times and lsd 1 time 100mcg • if i would trip on lsd i would do 75mcg and with magic mushrooms 1 portion around 10g fresh
5	dnm coke	• just this one time the rest of the night i had a gram of high quality dnm coke for the night • i also havent had amazing euphoria from dnm coke in a while im talking older batches of icoke and thecandymanuk • ive had dnm coke before from a different vendor and barely felt it and ive had street coke which i wont go near again

Table 3: Case Studies of the false positives detected on the drug dataset. They are real examples from Reddit.

phemistic phrase candidates using word2vec before ranking by SpanBERT.[6]

ILM performs poorly for this task. ILM is designed for text infilling for a *document*, but not for a *sentence*. By inspecting the output of ILM, we find that many top ranked candidates contain a punctuation which separates one sentence from another. For instance, in the masked sentence "these products can sometimes be found in shitty and dangerous [MASK] [MASK] pills", ILM ranks "places ." as the best candidates to replace the masks. Though we limit the ranking candidates to be the pre-selected phrases generated in Section 2.2, we still find its ranking performance to be suboptimal. However, we do find that ILM produces reasonable results for single-word prediction, which is not the task we consider.

3.2 False Positive Analysis

We present the top 30 outputs generated by EPD in Table 2 and perform case study on the false positives in Table 3. A closer analysis of the false positives reveals that some of them are true euphemistic phrases for drugs that were not present in the ground truth list (i.e., cases 2-5 in Table 3). This is of significant utility in the context of Internet communities, where memes and slangs lead to rapidly evolving euphemistic vocabulary and new types of drugs may be invented. For instance, we discover "nitrous oxide" (commonly known as "laughing gas", popular among young people). Among other false positives, we find that many of them are strongly related to a drug, but they are not proper euphemisms such as "crystal meth" and "xtc pills" ("ecstasy pills").

[6]We also point out that the pre-selection step saves 62% of the run time in our experiment.

3.3 Generalizability to Other Datasets

Owing the limited availability or the nature of euphemisms in the dataset, we perform experiments on only one real-life dataset. We did not perform experiments on the weapon and the sexuality datasets used in Zhu et al. (2021), because most euphemisms used are single words rather than multi-word phrases. Neither did we perform experiments on the hate speech dataset collected by Magu and Luo (2018) since the dataset was not publicly available.

Despite the lack of empirical support, we believe our approach to be generalizable to other datasets or domains since the algorithm does not make any domain-specific assumptions. Besides, EPD shares a similar model architecture with the algorithm proposed by Zhu et al. (2021), shown to be robust across various datasets. However, we do admit that the generalizability of our approach needs to be justified empirically on multiple real-life datasets. We leave the dataset collection and empirical evaluation for future work.

4 Related Work

Euphemism detection and its related work has recently received increased attention from the natural language processing and security and privacy communities (Durrett et al., 2017; Portnoff et al., 2017;

Magu and Luo, 2018; Pei et al., 2019; Felt and Riloff, 2020; Zhao et al., 2016; Yang et al., 2017; Zhu et al., 2020; Yuan et al., 2018; Hada et al., 2020; Zhu et al., 2021). Existing euphemism detection work have established a number of models by supervised (Pei et al., 2019), semi-supervised (Durrett et al., 2017) and unsupervised learning schemes (Zhao et al., 2016; Magu and Luo, 2018), on diverse categories and platforms (Yang et al., 2017; Hada et al., 2020), with and without distant-supervision (Portnoff et al., 2017; Felt and Riloff, 2020).

Without requiring any online search services, one major line of existing work have relied on static word embeddings (e.g., word2vec) in combination with network analysis (Taylor et al., 2017; Magu and Luo, 2018), sentiment analysis (Felt and Riloff, 2020), and semantic comparison across corpora (Yuan et al., 2018). However, the use of static word embeddings provides a single representation for a given word without accounting for its polysemy, and yields limited benefits. Therefore, Zhu et al. (2021) propose to explicitly harness the contextual information, formulate the problem as an unsupervised fill-in-the-mask problem (Devlin et al., 2019; Donahue et al., 2020), and solve it by a masked language model with state-of-the-art results.

Though prior studies report excellent results, to the best of our knowledge, none of the available approaches is capable of detecting euphemistic phrases without human effort.[7] Therefore, policy evaders could simply invent euphemistic phrases to escape from the censorship. Our work bridges this gap by extending the state-of-the-art euphemism detection approach proposed by Zhu et al. (2021) and achieves holistic euphemism detection by enabling the detection of euphemistic phrases.

5 Conclusion

We have proposed a solution to address the problem of euphemistic phrase detection. By mining quality phrases from the text corpus, pre-selecting euphemistic phrase candidates, and ranking phrases by a masked language model, we, for the first time, achieve euphemistic phrase detection automatically.[8] Moreover, we discover new euphemisms that are not even on the ground truth list, which is valuable for content moderation on social media platforms.

[7] Felt and Riloff (2020) achieves euphemistic phrases detection, *with* additional manual filtering process.

[8] Our code is publicly available at https://github.com/WanzhengZhu/Euphemism.

Acknowledgements

We thank the anonymous reviewers for their helpful comments on earlier drafts that significantly helped improve this manuscript. This research was supported by the National Science Foundation award CNS-1720268.

Ethical Considerations

The data we use in this paper are from the previous years, were posted on publicly accessible websites, and do not contain any personal identifiable information (i.e., no real names, email addresses, IP addresses, etc.). Just like Zhu et al. (2021), our analyses relying on user-generated content do not constitute human subjects research, and are thus not within the purview of the IRB.[9]

References

Beryl L Bellman. 1981. The paradox of secrecy. *Human Studies*, pages 1–24.

Cambridge Consultants. 2019. Use of AI in online content moderation. Ofcom report.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Chris Donahue, Mina Lee, and Percy Liang. 2020. Enabling language models to fill in the blanks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Greg Durrett, Jonathan K Kummerfeld, Taylor Berg-Kirkpatrick, Rebecca Portnoff, Sadia Afroz, Damon McCoy, Kirill Levchenko, and Vern Paxson. 2017. Identifying products in online cybercrime marketplaces: A dataset for fine-grained domain adaptation. In *Proceedings of Empirical Methods in Natural Language Processing (EMNLP)*.

Christian Felt and Ellen Riloff. 2020. Recognizing euphemisms and dysphemisms using sentiment analysis. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 136–145.

Takuro Hada, Yuichi Sei, Yasuyuki Tahara, and Akihiko Ohsuga. 2020. Codewords detection in microblogs focusing on differences in word use between two corpora. In *Proceedings of International*

[9] Readers are referred to the *Ethics* Section in Zhu et al. (2021) for more detailed information.

Conference on Computing, Electronics & Communications Engineering (iCCECE), pages 103–108. IEEE.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Jialu Liu, Jingbo Shang, Chi Wang, Xiang Ren, and Jiawei Han. 2015. Mining quality phrases from massive text corpora. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, pages 1729–1744.

Rijul Magu and Jiebo Luo. 2018. Determining code words in euphemistic hate speech using word embedding networks. In *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3111–3119.

Zhengqi Pei, Zhewei Sun, and Yang Xu. 2019. Slang detection and identification. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 881–889.

Rebecca S Portnoff, Sadia Afroz, Greg Durrett, Jonathan K Kummerfeld, Taylor Berg-Kirkpatrick, Damon McCoy, Kirill Levchenko, and Vern Paxson. 2017. Tools for automated analysis of cybercriminal markets. In *Proceedings of the 26th International Conference on World Wide Web (WWW)*.

Jingbo Shang, Jialu Liu, Meng Jiang, Xiang Ren, Clare R Voss, and Jiawei Han. 2018. Automated phrase mining from massive text corpora. *IEEE Transactions on Knowledge and Data Engineering*, 30(10):1825–1837.

Jherez Taylor, Melvyn Peignon, and Yi-Shin Chen. 2017. Surfacing contextual hate speech words within social media. *arXiv preprint arXiv:1711.10093*.

Hao Yang, Xiulin Ma, Kun Du, Zhou Li, Haixin Duan, Xiaodong Su, Guang Liu, Zhifeng Geng, and Jianping Wu. 2017. How to learn klingon without a dictionary: Detection and measurement of black keywords used by the underground economy. In *2017 IEEE Symposium on Security and Privacy*. IEEE.

Kan Yuan, Haoran Lu, Xiaojing Liao, and XiaoFeng Wang. 2018. Reading thieves' cant: automatically identifying and understanding dark jargons from cybercrime marketplaces. In *27th USENIX Security Symposium*, pages 1027–1041.

Kangzhi Zhao, Yong Zhang, Chunxiao Xing, Weifeng Li, and Hsinchun Chen. 2016. Chinese underground market jargon analysis based on unsupervised learning. In *2016 IEEE Conference on Intelligence and Security Informatics (ISI)*, pages 97–102. IEEE.

Wanzheng Zhu, Hongyu Gong, Rohan Bansal, Zachary Weinberg, Nicolas Christin, Giulia Fanti, and Suma Bhat. 2021. Self-supervised euphemism detection and identification for content moderation. In *42nd IEEE Symposium on Security and Privacy*.

Wanzheng Zhu, Hongyu Gong, Jiaming Shen, Chao Zhang, Jingbo Shang, Suma Bhat, and Jiawei Han. 2020. FUSE: Multi-faceted set expansion by coherent clustering of skip-grams. In *The European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*.