Gun Violence News Information Retrieval using BERT as Sequence Tagging Task

Hung-Yeh Lin*, Teng-Sheng Moh†, Bryce Westlake‡
*†Department of Computer Science, ‡Department of Justice Studies
San Jose State University
*hungyeh.lin@sjsu.edu, †teng.moh@sjsu.edu, ‡bryce.westlake@sjsu.edu,

***Abstract*—The growth in both frequency and severity of gun violence in the United States has necessitated increased research into prevention, despite the lack of funding. Comprising more than 60k gun violence media articles with a total data size of 520 MB, the gun violence database (GVDB) was developed to assist natural language processing researchers in developing and testing prevention methods. Original research based on the GVDB utilized a span-selection model to extract shooter and victim information, but their works might potentially trim out important span candidates. We proposed a new approach to improve identification accuracy and recognize every token in a sentence using a sequence tagging technique. We implemented a BIO sequence tagging model at the token-level using BERT, then further classified each token using LSTM, BiLSTM, and CRF. We found that utilizing BERT as an embedding layer, and decoding word representation as a sequence tagging task, improved shooter/victim identification compared to a span-selection model. We believe that if this improved model is combined with gun violence related keywords, automated techniques could be implemented to identify precursors/risks to gun violence on social media, allowing for intervention by law enforcement or community agencies before escalation to deaths.**

***Index Terms*—NLP, natural language processing, BERT, transformer, sequence tagging, BiLSTM, CRF, gun violence**

I. Introduction

There were 14,414 firearm homicides in the United States in 2018, growing to 19,141 in 2019 [1]. While the increase in deaths is problematic, news media also report that escalating gun violence has had a significant impact on mental health [2]. As a result, the growing gun violence is not only a social safety issues, but also a public health issue.

The rise of social media has amplified gang violence [3][4], which has always been a key factor in fears and occurrences of firearm homicides in the United States. This is because there is a growing trend of gang-involved youth using social networking websites to engage in 'internet banging' [5]. Internet banging involves using social media and chat rooms to publicly invalidate rival gangs' strength, toughness, and masculinity [6], and gain notoriety, by bragging about participation in violent acts, communicating about impending threats of violent acts, and/or broadcasting gang affiliations [7][8]. These online actions can be, and have been, precursors to real-world violence [9][10].

Social and violence outreach workers have turned to social media posts by gang members to identify potential risks for escalation to real-world violence [11]. However, the substantial number of posts appearing on social media makes this manual process impractical. As a result, automated techniques for collecting relevant information, using natural language processing techniques, have been proposed [12][13]. However, nuances to the context and sentiment of language, including gang names versus given names, racial profiling, and interpreting imagery, highlight the challenges of using automated techniques to interpret text [14][15][16][17][18].

As internet banging surges in the United States, we propose a new model to more accurately identify names of potential shooters and targets/victims from social media posts. By improving accuracy, we believe that this can later be combined with other methods [12], that focus on gun violence keywords and the sentiment of keywords, to better flag posts that may include threats and be precursors to gun violence. This information can then be used by community gun violence prevention organizations and/or law enforcement to prevent escalations to violence.

In 2016, Pavlick et al. [19] developed the gun violence database (GVDB), which consolidated news reports/articles about gun violence, using natural language processing (NLP) annotations. Ebner et al. [20] proposed new baseline models that predicted the value (argument) for each slot (role) to retrieve target (e.g., names of people) information. In this paper, we continue previous work using the GVDB to build novel models to more accurately identify targets from a sentence or a document than benchmark, baseline, models [20]. Regarded as the 'sequence tagging problem', we sought to tag each word or token with a different label, similar to Part-of-Speech (PoS) and Named Entity Recognition (NER). For example, in NER there is a PERSON tag that indicates whether a given token is a person's name or a person. Instead, we sought to use a BIO tagging method, short for beginning, inside, and outside [21], marking each word with their position of being targets or non-targets. We analyze the characteristics of each word and evaluate their combination.

Furthermore, we utilize Bidirectional Encoder Representations from Transformers (BERT) [22] at the word embedding layer and compare the implementation of Long Short-Term Memory (LSTM) [23], Bidirectional Long Short-Term Memory (BiLSTM), and Conditional Random Field (CRF) [24] to classify each token. Our main contribution was to present a new approach to better identify shooters/victims by converting span-selection problem into sequence tagging ones

978-1-6654-3902-2/21/$31.00 ©2021 IEEE

using BERT and deep learning neural networks.

II. Related Works

In this section, we will introduce GVDB, BERT, the evaluation benchmark from Multi-Sentence Argument Linking [20], and the BIO tagging method.

A. Gun Violence Database

GVDB was developed through crowd-sourced annotations of local news and television reports of gun violence throughout the United States [19]. Articles were automatically categorized, using a high-recall text classifier, and then vetted by humans to filter out false positives. The database consists of 5,394 annotated articles with shooter/victim information. Using an off-the-shelf information extraction system, [19] obtained a 4.7% precision and recall score (F1) on shooter's information, which is calculated using the following (1).

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{1}$$

Specifically, they provided the string of the target information and the span of the information. For example, in an article with the following: "... Lamesa police are investigating the shooting death of Dominique Adams, who died Tuesday afternoon on his way to UMC...", the target is annotated with the index from the 346th character to the 361st character, and the value of 'Dominique Adams' (see Fig. 2).

B. BERT

Bidirectional Encoder Representations from Transformers (BERT) [22] pre-trained using unlabeled corpus with two different pipelines, masked language model and next sentence prediction. Masked language model is used to predict a token from a sentence, while next sentence prediction is used to classify the relationship between two sentences. In Fig. 1, these pre-training processes rely on the attention mechanism to encode input features [25] which learn the essence of human languages. BERT can be fine-tuned with downstream NLP tasks. In our case, we utilized this characteristic to run sequence tagging on token-level classification. In other words, we used BERT as our embedding layer and further implemented token-classifier as the downstream task.

C. Multi-Sentence Argument Linking

Ebner et al [20] proposed a multi-sentence argument linking model using slot filling, to extract information at the document-level. First, their model consisted of BERT at the embedding layer, to extract the representative text spans. Text spans are slices of a document or a sentence and are usually represented with the start position and the end position from a sentence as in Fig. 2. This was used to calculate the representation of spans to convert human language into math vectors that a machine could understand. Second, they pruned extra span collections, to leave the most possible spans. Third, they introduced a new link scoring function, which took the distance between the "event" and "candidate" spans into consideration within the feed-forward neural network.

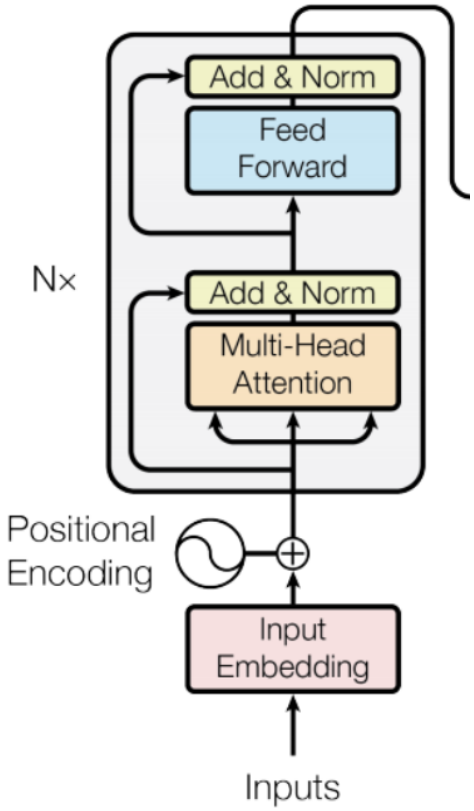

Fig. 1. Model Architecture of Encoder from Transformer [22]

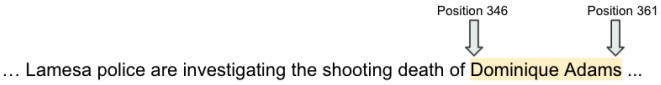

Fig. 2. Example of Text Span

Fourth, learning and then decoding steps were incorporated, to maximize the probability of one candidate as the predicted result.

The analysis by [20] consisted of a training set of 5,056 articles, a development set of 400 articles, and a test set of 500 articles. Articles from GVDB without a reliable publication date or lacking annotated spans were excluded from the training set, as were 100 articles spanning the week between the development and test datasets, to limit the possibility of data occurring in both datasets.

D. BiLSTM-CRF for Sequence Tagging

Inspired by [26], we also implemented BiLSTM architecture, but with BERT at the embedding layer. This was implemented because it can efficiently take input features forward and backward at the same time and feed the output into the CRF layer. In the CRF layer, we first compute log likelihood with tagged information and then decode the most probable sequence of tags. CRF utilize neighbor tag

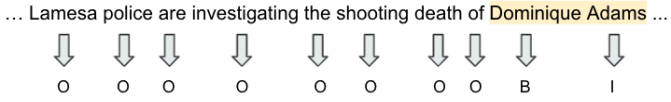

Fig. 3. Example of Beginning-Inside-Outside Tagging

information to predict the current tag, and in general it has been shown that CRF can produce better accuracy.

III. Methodology

In this section, we discuss the methodology for pre-processing, word embedding, sequence tagging classifying, and evaluation.

Instead of calculating the probabilities of spans to extract target information [20], we classify each of the tokens, through BIO tagging, to see if that token is part of the target. Hence, the main difference we implement, compared to [20], is how we predict the target. Specifically, we use the combination of LSTM, BiLSTM, and CRF to classify tokens after getting the word representation from BERT.

A. Pre-process

In order to do BIO tagging, we tagged every token, or every word, with either B, I, or O. B marks the beginning of the target token, I marks the rest of the target tokens, and O means null. Illustrated in Fig. 3, "Dominique" is marked as the beginning of the target, and "Adams" is tagged as the inside of the target. Also, because there might be multiple targets mentioned in an article, all targets were tagged with "B" and "I" for training. Moreover, to make the sequence length of each sample in the batch consistent, we appended empty strings tagged as O (null) at the end of the sequence. For example, when the max sequence length was 256, and one sample only contained 200 tokens, that data sample was padded with 56 "O" at the end.

B. Embedding Layer

We utilized BERT as a word embedding layer. BERT is a way to convert the human language to a mathematical vector that a computer can understand. Each word has its own vector that represents a meaning for computers. We took the last hidden state from BERT with the default dimension (768) and fed it into the following pipeline.

C. Sequence Tagging Classifier

We utilized Long Short-Term Memory (LSTM), BiLSTM, and Conditional Random Field (CRF) to further classify each word or each token. Hence, each token was pre-labeled as B, I, O which are frequently used in sequence tagging NLP tasks. We implemented 6 different models: BERT (Fig. 4), BERT-LSTM (Fig. 5), BERT-BiLSTM (Fig. 6), BERT-CRF, BERT-LSTM-CRF, and BERT-BiLSTM-CRF (Fig. 7). We did not show the model architectures of BERT-CRF and

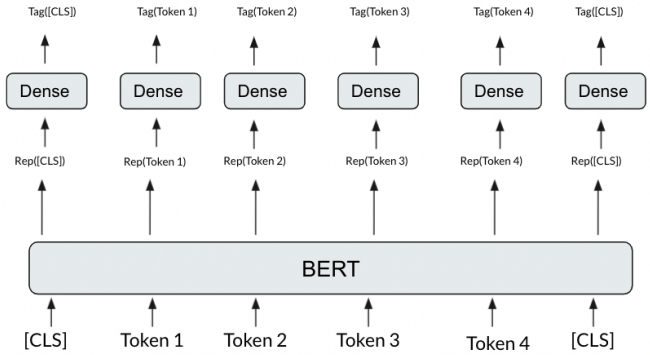

Fig. 4. Model Architecture of using BERT for Sequence Tagging

BERT-LSTM-CRF because the difference between BERT-Linear/BERT-LSTM/BERT-BiLSTM and BERT-CRF/BERT-LSTM-CRF/BERT-BiLSTM-CRF is that we replaced a feed-forward layer with a CRF layer. The final output was sent to a softmax layer used to normalize the probabilities of three labels (BIO).

In Fig. 4, the architecture is the same as sequence tagging models (POS or NER) in [22]. In Fig. 5, the output from BERT was passed to an LSTM layer and another feed-forward layer. In Fig. 6, the output from BERT was passed to a BiLSTM layer, and internally, the representations were passed to a forward LSTM layer and a backward LSTM layer at the same time. We found that summing up forward LSTM and backward LSTM outperformed concatenating them. Hence, we chose to sum up forward LSTM and backward LSTM for BERT-BiLSTM and BERT-BiLSTM-CRF. For simplicity, the dimension of the LSTM's hidden states we used was the same as the output dimension of BERT - 768. Because BERT-CRF, BERT-LSTM-CRF, and BERT-BiLSTM-CRF were similar to the first three models, we only show Fig. 7 as an example for CRF. Inspired by [27] and [28], we implemented BERT-LSTM-CRF and BERT-BiLSTM-CRF on sequence labelling tasks. However, [27] used BERT-LSTM-CRF for NER on information extraction of municipal solid waste crisis, and [28] utilized BERT-BiLSTM-CRF with BIO tagging for NER on Chinese electronic medical records.

D. Evaluation

To evaluate the performance of the models, we first extracted predicted targets from BIO tagging method and compared predicted results with actual targets using exact match or strict match. From the output of the neural network, there were 1-dimension vectors with 3 float values as the probabilities of "B", "I", and "O", and each token was classified based on probabilities. After each of the tokens were classified as "B", "I", "O", we looked for the words that were classified as the combination of B and I. Take an example in Fig. 8, "Dominique" was classified as "B", and "Adams" was classified as "I", then "Dominique Adams" was classified as the target. When there were multiple targets that were classified, we evaluated the probability of the token that was

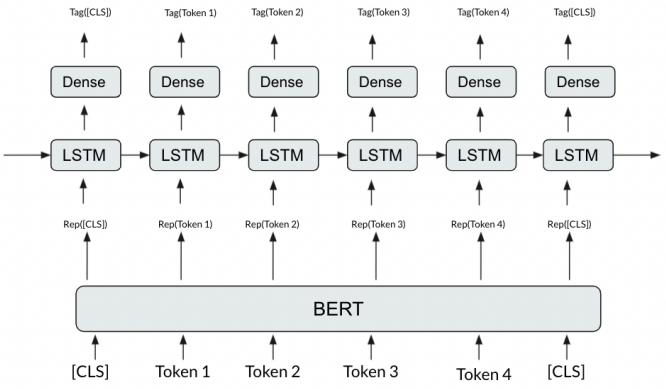

Fig. 5. Model Architecture of using BERT-LSTM for Sequence Tagging

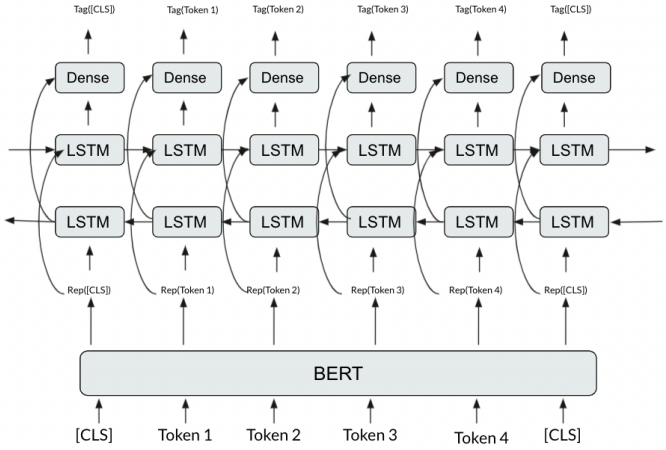

Fig. 6. Model Architecture of using BERT-BiLSTM for Sequence Tagging

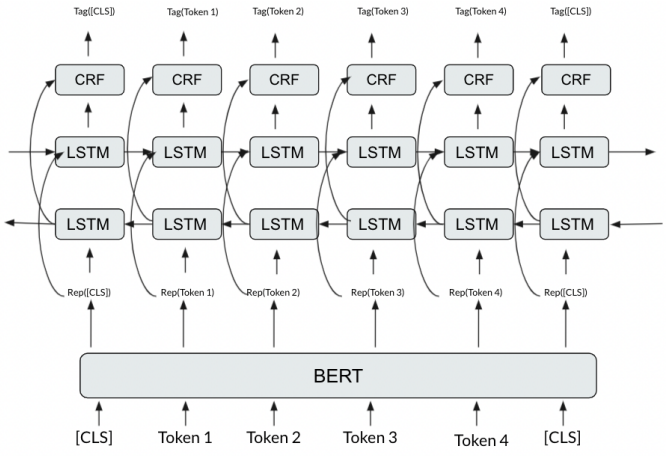

Fig. 7. Model Architecture of using BERT-BiLSTM-CRF for Sequence Tagging

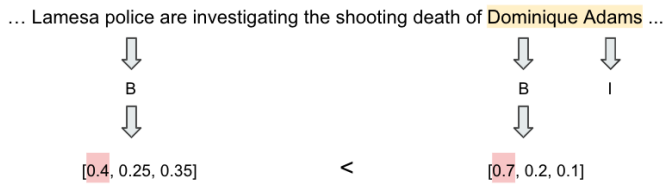

Fig. 8. Evaluation when Multiple Targets

marked as "B" (0.4 and 0.7 in Fig. 8), and the target with the highest probability was treated as the 'true' target. "Police" and "Dominique" were both classified as "B" because the probability of being "B" on "police" (0.4) was the highest compared to the probability of being "I" (0.25) and the probability of being "O" (0.35). But "Dominique Adams" was predicted as the true target because "Dominique" gave the highest probability as a B tag (0.7 is greater than 0.4). We considered the prediction as a true positive only if the predicted result was exactly the same as the original target value. We used the testing set separated in [20] to evaluate our model.

IV. Experiment

In this section, we discuss the structure of the GVDB dataset and how we set up the experiment in terms of software and hardware.

A. Dataset

In GVDB, they collected news articles related to gun violence and annotated each article with shooters' and victims' information such as name, age, and gender. We used the separated dataset in [20] that was composed of almost 18 MB texts and 6,292 samples. We only focused on extracting the name of the target, so we extracted the full text of the article and the targets' names. Furthermore, we tagged each word using BIO sequence tagging technique.

Due to the nature of BERT, BERT's tokenizer [29] will convert some tokens into multiple subtokens. In order to solve the inconsistency between the number of tokens and the number of tags, we tagged subtokens based on the previous tokens. For example, "Alvin" might be converted to "Al" and "##vin", and in this case, "Alvin" was originally tagged with B, so after tokenization, we tagged "Al" as B and "##vin" as I. If the original token was tagged as O, then all the subtokens were O as well. The idea was borrowed from [30].

B. Experiment Setup

We used Pytorch and Huggingface's transformers [29] to build our models. And we did experiments using the setup parameters below in Table 1.

Note that, although we included 128 as one of max sequence length, 128 is still too small to cover most of the dataset during training and evaluation; hence, we suggest 256 or more than 256.

The parameters in Table 1 were based on our experiments starting from smaller values of each, and the chosen values showed conspicuous results.

TABLE I
SOFTWARE VERSION

Hyperparameter	Value
Nvidia Driver	470.57.02
CUDA	11.4
Pytorch	1.8.0
GPU	8 GB GeForce RTX 3070
learning rate	1e-05, 5e-05, 2e-06, 5e-06
batch size	2, 4, 6 , 8, 10
max sequence length	128, 256, 512
LSTM hidden layers	768
baseline	0.0001
patience	10

Also, we implemented early stopping on our models, and we stopped training if a model was not improving for a certain amount of times that was the definition of "patience". We defined two scenarios in which a model did not improve: the loss was going down each epoch or the loss was lower than a baseline value. First, we tried fixed epochs and observed that most of the models converged after the loss was under roughly 0.001, so we set a baseline value of 0.001 with a epoch number of 50. Furthermore, we found that the best patience value was 10 as to even out the occurrences of two scenarios.

Algorithm 1 Early Stopping Algorithm

Input: baseline, patience

Initialization :
$lowest \leftarrow$ NULL
$count \leftarrow 0$
for $epoch = i$ to N **do**
  for $batch$ in $batches$ **do**
    Train model
    Calculate loss
  end for
  if $lowest$ is $NULL$ **then**
    $lowest \leftarrow loss$
  else if $loss \leq lowest$ **then**
    $lowest \leftarrow loss$
    $count \leftarrow count$ + 1
  else if $loss \leq baseline$ **then**
    $count \leftarrow count$ + 1
  end if
  if $count \geq patience$ **then**
    Stop training
  end if
end for

We performed a grid search on hyperparameters to find the optimal result for each model. In general, when the learning rate was relatively lower than others(Table 1), the result took longer epochs to converge. Using larger batch sizes also required more epochs. As a result, the best combination often fell on a learning rate of 5e-06, a batch size of 2, a number of epoch of 20, and a max sequence length of 256. Our experiments and models have been published [31].

V. RESULTS AND DISCUSSION

In this section, we provide our results and discuss our model performance.

We compared the performance of our sequence labelling models for shooter (Table 2) and victim (Table 3) name identification for precision, recall, and F1 (combination of precision and recall), to those from the original GVDB analysis [19] and the span-selection baseline models [20]. We found that our models outperformed comparison models for shooter identification. Specifically, that BERT-LSTM was highest on precision and F1, while BERT-BiLSTM-CRF was highest on recall. Comparing CRF models to non-CRF models, we see that results are similar. The idea of CRF that utilizes neighbor tag information does not lead to prominent results, given BERT is present. This is because BERT and BiLSTM take the entire document or sentence into account during training.

TABLE II
PERFORMANCE COMPARISON ON PREDICTING SHOOTER NAME

Chosen Models	Metrics		
	Precision	***Recall***	***F1***
GVDB	5.8%	3.9%	4.7%
Baseline	55.3%	51.1%	53.1%
BERT	49.4%	52.0%	50.6%
BERT-LSTM	**60.6%**	53.1%	**56.6%**
BERT-BiLSTM	54.3%	55.7%	55.0%
BERT-CRF	47.3%	58.9%	52.4%
BERT-LSTM-CRF	54.8%	51.9%	53.3%
BERT-BiLSTM-CRF	46.5%	**67.1%**	55.0%

For victim name identification, we found that BERT can produce similar, or better, F1 scores when compared to the baseline span-selection model, with BERT-LSTM-CRF and BERT-CRF performing the best of all models for recall and F1 respectively.

TABLE III
PERFORMANCE COMPARISON ON PREDICTING VICTIM NAME

Chosen Models	Metrics		
	Precision	***Recall***	***F1***
GVDB	10.2%	8.5%	9.3%
Baseline	**61.2%**	63.3%	62.2%
BERT	57.1%	82.0%	67.3%
BERT-LSTM	54.9%	78.7%	64.7%
BERT-BiLSTM	51.0%	82.9%	63.2%
BERT-CRF	57.7%	83.2%	**68.1%**
BERT-LSTM-CRF	54.3%	**89.7%**	67.7%
BERT-BiLSTM-CRF	54.9%	87.0%	67.3%

Recalling that accuracy was higher in Table 3, when compared to Table 2, it is important to note that recall is easily affected by data distribution. In the GVDB, there is an unbalanced distribution of shooter (Fig. 9) and victim (Fig. 10) information. In 431 of the 500 data samples in the testing set, the victim information was present. In comparison, in only 247 of the 500 were shooter information present. If the model regards that a sentence or a document does not contain a shooter/victim but it actually does, then that data sample is a false negative. On the other hand, if the model

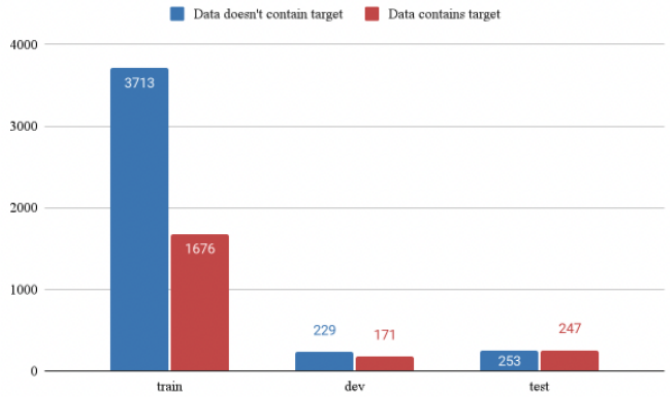

Fig. 9. Shooter Data Distribution

Fig. 10. Victim Data Distribution

regards that a sentence or a document contains a shooter/victim but it actually does not, then that data sample is a false positive. Because there are few non-victim data points that can be evaluated as false negatives, the false negatives in victim evaluation are less. In eq. (2) and eq. (3), we see that precision is affected by false positives and recall is affected by false negatives. As the number of false positives goes up, precision goes down. As the number of false negatives goes down, recall goes up. Hence, recall is higher, but precision is not. In addition, in [19], although they did not reveal the implementation of the text classifier during collection phase, GVDB was collected using a high-recall text classifier, and it might affect recalls, especially on CRF models, if they used similar graph-based models.

$$precision = \frac{True\ Positive}{True\ Positive\ +\ False\ Positive} \quad (2)$$

$$recall = \frac{True\ Positive}{True\ Positive\ +\ False\ Negative} \quad (3)$$

Although BERT-LSTM had the best precision for predicting shooter names, the baseline model had the best precision for predicting victim names. Sequence tagging has an inconsistency issue that can make it difficult to see the combination of "B" and "I". For example, if there is a sequence of "B O I", only the first token that is seen as "B" will be treated as a target, but it is possible that these three tokens form an actual target. Nevertheless, experimentally, sequence tagging using BERT generates better F1 scores on GVDB.

We assume that BERT already keeps the information from forward and backward sentences, so including backward information in BiLSTM during fine-tuning phase does not change performance compared to LSTM. Although [27] and [28] both implemented BERT-LSTM-CRF and BERT-BiLSTM-CRF, they did not compare the performance between BERT and CRF; hence, our future work will examine this assumption. We encourage others to use the same model architectures on more and different sequence tagging benchmarks. However, we propose a new technique on GVDB to extract target information with better F1 scores.

VI. Conclusion

In this paper, we present a new model for extracting shooter and victim names from news media articles. While span selection has been used previously [20], there is a potential to drop candidates during the pruning step. This can be improved by utilizing the fine-tuning step in BERT and then converting the original problem into a sequence tagging task. By tagging each token in a document, we are able to classify each's tag and further evaluate results with a BIO technique. Furthermore, this method better deals with the issue of multiple targets in one sentence, so as to only take the most probable data as our predicted target. As a result, our new model outperforms previous models in F1 scores.

There are a multitude of actions on social media that can lead to real-world gang violence, even expressions of grief and loss [32][33]. However, effective responses from crisis intervention workers on social media can lead to de-escalation of violent situations offline [34]. As a result, it is important that the accuracy of automated natural language processing methods be improved for them to be useful for deployment by social workers. By reducing the number of false positive and negative name identifications, and combining with techniques for detecting loss and aggression [18][35][36], violence prevention workers can be correctly alerted to posts which may escalate to real-world violence, correctly identify the persons (shooters and victims) who may be involved in said violence, and apply de-escalation tactics.

We believe that, by continuously accumulating training data and process, we can improve this model and better predict the gun violence information accurately. Combining our model with searching keyword or classifying news, a detector can be placed on social media to capture suspicious gun violence activities across online platforms and aid law enforcement investigators with preventing victimization and/or identifying perpetrators of offline violence.

Acknowledgment

We thank California Association of Criminalists McLaughlin Endowment for their support in our research funding.

REFERENCES

[1] CDC. (2021, Sep. 13). *Faststats* [Online]. Available: https://www.cdc.gov/nchs/fastats/homicide.htm

[2] E. E. McGinty, D.W. Webster, M. Jarlenski, and C. L. Barry, "News media framing of serious mental illness and gun violence in the United States, 1997-2012," in *Am J Public Health*, vol. 104(3), pp. 406-413, March 2014, doi:10.2105/AJPH.2013.301557.

[3] J. Densley, R. Deuchar, and S. Harding, "An Introduction to Gangs and Serious Youth Violence in the United Kingdom," Youth Justice 20, no. 1–2, April 2020, pp. 3–10. doi.org/10.1177/1473225420902848.

[4] D. C. Pyrooz and R. K. Moule Jr., "Gangs and Social Media," Nov. 2019. [Online]. Available: https://oxfordre.com/criminology/view/10.1093/acrefore/9780190264079.001.0001/acrefore-9780190264079-e-439

[5] D. U. Patton, R. D. Eschmann, and D. A. Butler, "Internet banging: New trends in social media, gang violence, masculinity and hip hop," in *Computers in Human Behavior*, vol 29(5), pp. A54–A59, Jan. 2013, doi.org/10.1016/j.chb.2012.12.035

[6] F. Stuart, "Code of the Tweet: Urban Gang Violence in the Social Media Age," in *Social Problems*, vol 67(2), pp 191-207, May 2020, doi.org/10.1093/socpro/spz010

[7] R. K. Moule Jr, D. C. Pyrooz, and S. H. Decker, "From What the F#@% is a Facebook?' to 'Who Doesn't Use Facebook?': The role of criminal lifestyles in the adoption and use of the Internet," in *Social Science Research*, vol 42(6), pp. 1411-1421, Nov. 2013, doi: 10.1016/j.ssresearch.2013.06.008

[8] D. U. Patton, R. D. Eschmann, C. Elsaesser, and E. Bocanegra, "Sticks, stones and Facebook accounts: What violence outreach workers know about social media and urban-based gang violence in Chicago," in *Computers in Human Behavior*, vol 65, pp. 591-600, doi.org/10.1016/j.chb.2016.05.052

[9] D. U. Patton, J. Lane, P. Leonard, J. Macbeth, and J. R. Smith-Lee, "Gang violence on the digital street: Case study of a South Side Chicago gang member's Twitter communication," in *New Media and Society*, vol 19(7), pp. 1000-1018, doi: 10.1177/1461444815625949

[10] D. U. Patton, "Social media as a vector for youth violence: A review of the literature," in *Computers in Human Behavior* vol 35, pp. 548-553, Jun. 2014, doi.org/10.1016/j.chb.2014.02.043

[11] J. M. Hyatt, J. A. Densley, and C. G. Roman, "Social Media and the Variable Impact of Violence Reduction Interventions: Re-Examining Focused Deterrence in Philadelphia," in *Research on Gang-Related Violence in the 21st Century* vol 10(5):147, 2021, doi.org/10.3390/socsci10050147

[12] S. Chang et. al., "Detecting Gang-Involved Escalation on Social Media Using Context" in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 46-56, doi.org/10.18653/v1/D18-1005

[13] D. U. Patton, K. McKeown, O. Rambow, and J. Macbeth, "Using Natural Language Processing and Qualitative Analysis to Intervene in Gang Violence: A Collaboration Between Social Work Researchers and Data Scientists," 2016, *arXiv:1609.08779*. [Online]. Available: https://arxiv.org/abs/1609.08779

[14] D. U. Patton, W. R. Frey, and M. Gaskell, "Guns on social media: complex interpretations of gun images posted by Chicago youth," in *Palgrave Commun*, vol 5(119), 2019, doi.org/10.1057/s41599-019-0330-x

[15] D. U. Patton, W. R. Frey, K. A. McGregor, F. T. Lee, K. McKeown, and E. Moss, "Contextual Analysis of Social Media: The Promise and Challenge of Eliciting Context in Social Media Posts with Natural Language Processing," in *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pp. 337-342, doi.org/10.1145/3375627.3375841

[16] S. L. Blodgett and B. O'Connor, "Racial Disparity in Natural Language Processing: A Case Study of Social Media African-American English," 2017, *arXiv:1707.00061* . [Online]. Available: https://arxiv.org/abs/1707.00061

[17] R. Zhong, Y. Chen, D. Patton, C. Selous, and K. McKeown, "Detecting and Reducing Bias in a High Stakes Domain," in *Proceedings of the 2019 EMNLP-IJCNLP*, pp. 4765-4775, doi.org/10.18653/v1/D19-1483

[18] M. Sap, D. Card, S. Gabriel Y. Choi, and N. A. Smith, "The Risk of Racial Bias in Hate Speech Detection," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1668-1678, Jul. 2019, doi.org/10.18653/v1/P19-1163

[19] E. Pavlick, H. Ji, X. Pan, and C. Callison-Burch, "The Gun Violence Database: A new task and data set for NLP," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Austin, Texas, 2016, pp. 1018–1024.

[20] S. Ebner, P. Xia, R. Culkin, K. Rawlins, and B. V. Durme, "Multi-Sentence Argument Linking," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online, 2020, pp. 8057–8077.

[21] L. A. Ramshaw and M. P. Marcus, "Text Chunking using Transformation-Based Learning", *Proceedings of the ACL Third Workshop on Very Large Corpora*, pp. 82-94, 1995.

[22] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL-HLT*, 2019, pp. 4171–4186.

[23] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," in *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 15 Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[24] J. Lafferty, A. McCallum, and F. C.N. Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data," in *Proceedings of the 18th International Conference on Machine Learning 2001 (ICML 2001)*, 2001, pp. 282–289.

[25] A. Vaswani et al., "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS)*, vol. abs/1706.03762.

[26] Z. Huang, W. Xu, and K. Yu, "Bidirectional LSTM-CRF models for sequence tagging," 2015, *arXiv:1508.01991*. [Online]. Available: http://arxiv.org/abs/1508.01991

[27] T. Wan, W. Wang, H. Zhou, "Research on Information Extraction of Municipal Solid Waste Crisis using BERT-LSTM-CRF," in *NLPIR 2020: Proceedings of the 4th International Conference on Natural Language Processing and Information Retrieval*, pp. 205–209, December 2020, doi.org/10.1145/3443279.3443314.

[28] W. Zhang, S. Jiang, S. Zhao, K. Hou, Y. Liu and L. Zhang, "A BERT-BiLSTM-CRF Model for Chinese Electronic Medical Records Named Entity Recognition," 2019 12th International Conference on Intelligent Computation Technology and Automation (ICICTA), 2019, pp. 166-169, doi: 10.1109/ICICTA49267.2019.00043.

[29] T. Wolf et al., "HuggingFace's Transformers: State-of-the-art Natural Language Processing," 2020, *arXiv:1910.03771*. [Online]. Available: https://arxiv.org/abs/1910.03771

[30] C. Sun, L. Huang, and X. Qiu, "Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence," Mar. 2019, *arXiv:1903.09588*. [Online]. Available: https://arxiv.org/abs/1903.09588

[31] H. Y. Lin, "Gun-Violence-Information-Retrieval-Using-BERT-as-Sequence-Tagging-Task," Oct. 2021, [Source code]. Available: https://github.com/bubblemans/Gun-Violence-Information-Retrieval-Using-BERT-as-Sequence-Tagging-Task

[32] D. U. Patton, J. MacBeth, S. Schoenebeck, K. Shear, and K. McKeown, "Accommodating Grief on Twitter: An Analysis of Expressions of Grief Among Gang Involved Youth on Twitter Using Qualitative Analysis and Natural Language Processing," in *Proceedings from the Digital Mental Health Conference - London, 2017*, vol 10, doi.org/10.1177/1178222618763155

[33] D. U. Patton, O. Rambow, J. Auerbach, K. Li, and W. Frey, "Expressions of loss predict aggressive comments on Twitter among gang-involved youth in Chicago," in *NPJ digital medicine* vol1(11), doi.org/10.1038/s41746-018-0020-x

[34] D. U. Patton, R. D. Eschmann, C. Elsaesser, and E. Bocanegra, "Sticks, stones and Facebook accounts: What violence outreach workers know about social media and urban-based gang violence in Chicago," in *Computers in Human Behavior*, vol 65, pp 591-600, Dec. 2016, doi.org/10.1016/j.chb.2016.05.052

[35] T. Blevins, R. Kwiatkowski, J. MacBeth, K. McKeown, D. Patton, and O. Rambow, "Automatically Processing Tweets from Gang-Involved Youth: Towards Detecting Loss and Aggression," in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 2196-2206. Available: https://aclanthology.org/C16-1207

[36] S. L. Fowler, A. Stylianou, R. Mousavi, S. Reid, and D. Zhang, "Detecting Violent Crime with Gang Social Media Postings," Dec. 2020. [Online] Available: https://aisel.aisnet.org/icis2020/social_media/social_media/16/