Expert Systems With Applications 199 (2022) 116720

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Federated learning for violence incident prediction in a simulated cross-institutional psychiatric setting

Thomas Borger [a,b], Pablo Mosteiro [a,*], Heysem Kaya [a], Emil Rijcken [c,a], Albert Ali Salah [a,f], Floortje Scheepers [d], Marco Spruit [e,g,a]

[a] *Department of Information and Computing Sciences, Utrecht University, Utrecht, The Netherlands*
[b] *KPMG N.V., Amstelveen, The Netherlands*
[c] *Jheronimus Academy of Data Science, Eindhoven University of Technology, 's-Hertogenbosch, The Netherlands*
[d] *Department of Psychiatry, University Medical Center Utrecht, Utrecht, The Netherlands*
[e] *Department of Public Health & Primary Care, Leiden University Medical Centre, Leiden, The Netherlands*
[f] *Department of Computer Engineering, Boğaziçi University, Istanbul, Turkey*
[g] *Leiden Institute of Advanced Computer Science, Leiden University, Leiden, The Netherlands*

ARTICLE INFO

Keywords:
Federated learning
Violence prediction
Neural networks
Psychiatry
Clinical notes

ABSTRACT

Inpatient violence is a common and severe problem within psychiatry. Knowing who might become violent can influence staffing levels and mitigate severity. Predictive machine learning models can assess each patient's likelihood of becoming violent based on clinical notes. Yet, while machine learning models benefit from having more data, data availability is limited as hospitals typically do not share their data for privacy preservation. Federated Learning (FL) can overcome the problem of data limitation by training models in a decentralised manner, without disclosing data between collaborators. However, although several FL approaches exist, none of these train Natural Language Processing models on clinical notes. In this work, we investigate the application of Federated Learning to clinical Natural Language Processing, applied to the task of Violence Risk Assessment by simulating a cross-institutional psychiatric setting. We train and compare four models: two local models, a federated model and a data-centralised model. Our results indicate that the federated model outperforms the local models and has similar performance as the data-centralised model. These findings suggest that Federated Learning can be used successfully in a cross-institutional setting and is a step towards new applications of Federated Learning based on clinical notes.

1. Introduction

Inpatient violence is a serious problem in clinical psychiatry, causing short- and long-term damage to property as well as people (Havaei, MacPhee, & Lee, 2019; Inoue, Tsukano, Muraoka, Kaneko, & Okamura, 2006; van Leeuwen & Harte, 2017; Nijman, Bowers, Oud, & Jansen, 2005). Violence Risk Assessment (VRA) has been used in mental healthcare to inform medical decisions and mitigation strategies (Conroy & Murrie, 2012; Singh et al., 2014). Several manual VRA methods have been proposed and evaluated (Almvik, Woods, & Rasmussen, 2000; Douglas et al., 2014; Ogloff & Daffern, 2006), yet these methods are time-consuming and subjective, and some of them require advanced training to use (Webster, Nicholls, Martin, Desmarais, & Brink, 2006). Machine Learning (ML) methods promise to address these limitations, developing fast and objective predictions based on patient data present in Electronic Health Records (EHR).

In the psychiatry domain, a particularly promising ML approach is Natural Language Processing (NLP), since EHRs contain large amounts of unstructured clinical notes written by nurses and psychiatrists. The information in these notes could be employed in decision-support systems to aid psychiatrists in predicting aggression, diagnosing patients, predicting side-effects from medication, and predicting suicide attempts, among others. The information is reported in subtle and nuanced ways, and often includes typographical errors, abbreviations and technical terms. Not surprisingly, a common problem encountered by ML researchers in the clinical domain are datasets that are small (Pestian, Nasrallah, Matykiewicz, Bennett, & Leenaars, 2010) or too specific (Suchting, Green, Glazier, & Lane, 2018). Thus, increasing

* Corresponding author.
E-mail addresses: p.mosteiro@uu.nl (P. Mosteiro), thomasborger2@gmail.com (T. Borger), h.kaya@uu.nl (H. Kaya), e.f.g.rijcken@tue.nl (E. Rijcken), a.a.salah@uu.nl (A.A. Salah), F.E.Scheepers-2@umcutrecht.nl (F. Scheepers), M.R.Spruit@lumc.nl (M. Spruit).

https://doi.org/10.1016/j.eswa.2022.116720
Received 8 July 2021; Received in revised form 26 November 2021; Accepted 21 February 2022
Available online 10 March 2022

dataset size and diversity is desirable for performance of ML models, in particular NLP models used in psychiatry.

Combining datasets from multiple departments and institutions would be a natural way to enlarge datasets for various tasks. Yet, medical institutions are usually not allowed to combine their data (Flikweert et al., 2020). Thus, instead of sharing data, machine learning models can be shared amongst institutions, using local data for training and/or fine-tuning.

This is the basis of Federated Learning (FL) (McMahan, Moore, Ramage, & y Arcas, 2016). Through FL, multiple parties collaborate in solving an ML task under the coordination of a central server, where data are never allowed to leave a party's device (Kairouz, McMahan, Avent, Bellet, et al., 2021). Though some losses are expected with respect to a data-central approach, it has been shown that these could be quite small and acceptable given the gain in privacy (Sheller, Reina, Edwards, Martin, & Bakas, 2019). FL has been gaining traction in recent years, and applications within the medical domain are slowly emerging (Deist et al., 2020; Kairouz et al., 2021). However, none of the clinical applications of FL so far employ clinical texts.

In this work, we employ clinical texts for FL, examining violence risk assessment. We seek to find how FL compares to centrally- and locally trained models. For this comparison, we use free texts in EHRs. Since we do not have access to data from multiple institutions, we use "mock" institutions, created from the data of a single location using nursing-ward-based partitioning. We train four machine learning models: a federated model, a data-centralised model and two local models (A and B). Here, A and B are the names of the mock institutions we created. Then, we compare the performance of these four models on institutions A and B separately and on the combined test dataset.

Our main contributions are:

- We demonstrate that FL applied to NLP models and trained on clinical texts has similar performance as a centralised model, and better than locally trained models.
- We highlight the potential of FL for clinical psychiatry.

The remainder of this paper is structured as follows. Section 2 discusses related work regarding FL in the medical domain. Section 3 describes the dataset, and explains the method for obtaining the empirical results. Sections 4 and 5 state and discuss the empirical results. Finally, Section 6 provides the conclusions drawn from the results.

2. Related work and background

Multiple Machine Learning (ML) methods have been proposed to tackle the problem of Violence Risk Assessment (VRA). Bader and Evans (2015) attempted to differentiate between patients perpetrating severe and repeated aggression and non-aggressive patients, using common risk factors as predictor variables. In a retrospective study, Raja and Azzoni (2005) found some factors that seemed to correlate with inpatient violence. Menger, Spruit, van Est, Nap, and Scheepers (2019), Le, Montgomery, Kirkby, and Scanlan (2018), and Cook et al. (2016) exploited the abundant free text in EHRs to employ Natural Language Processing (NLP) to this task. Beyond VRA, Pestian et al. (2010) used NLP to classify suicide notes as legitimate or elicited.

Two limiting factors in the development of fair and accurate ML models for the healthcare domain are dataset size and diversity. Of the studies mentioned above, only one had more than a few thousand data points (Le et al., 2018). Its limitation, however, was that they predicted existing VRA instrument scores, not real violence incidents. Suchting et al. (2018) had nearly 30 thousand data points, yet they report being limited both by dataset diversity (due to the nature of their facility) and by dataset size (due to the imbalanced nature of the dataset, as most patients do not engage in violence). Aggregating data from multiple institutions would tackle both problems. However, as medical data often resides in secure data silos across institutions (Lehne, Sass, Essenwanger, Schepers, & Thun, 2019), aggregating these data is not possible.

Federated Learning (FL) is a novel technique for training ML models on decentralised data (Konečný et al., 2016). It began with the question of how one can train an ML model in a setting where data is unevenly distributed across a large number of devices, and the data cannot be shared among devices or with the central server. FL provides a solution to this question through decentralised training, orchestrated by a central server. The server initialises and sends a model to each participating institution or data silo. Each institution trains the model on their own data, and shares the updated model's parameters with the central server. The server then aggregates all models and creates a new global model. A widely used algorithm for creating a new model is FedAvg (McMahan et al., 2016), which performs a weighted average over the parameters of all models to create a new model. Other algorithms have been proposed to allow the use of adaptive optimisers, such as FedAdagrad, FedYogi, and FedAdam (Reddi et al., 2021).

FL has brought promising results in recent literature, where federated models perform nearly on par with data-centralised models for medical classification tasks, such as brain tumour segmentation (Li et al., 2019; Sheller et al., 2019) and in-hospital mortality prediction (Choudhury et al., 2019). The technique has been applied on private medical data as well by utilising the Personal Health Train (PHT), for classifying post-treatment survival chances in lung cancer patients, by collaborating with eight medical institutions (Deist et al., 2020). PHT is a platform aiming to provide healthcare data from various sources to researchers while ensuring privacy protection. FL has also been used to predict suicidal ideation in online social care texts (Ji et al., 2019). During the literature search conducted at time of study, no applications of FL on models employing clinical texts were identified. Table 1 shows all the aforementioned methods, together with their goals and limitations.

To bring FL to the psychiatric domain, a Natural Language Processing (NLP) task is chosen for this study. Clinical notes have been written about admitted patients on a daily basis for many years across medical institutions. This means that local datasets are available for research at these institutions. Issues with these clinical texts are that they are semi-structured, and sometimes contain thousands up till tens of thousands of words for a single admission period, making feature extraction a difficult task. Menger, Scheepers, and Spruit (2018) compared various methods to convert texts into vectorial representations, including bag-of-words, TF–IDF, Word2Vec and Doc2Vec. Following previous work (Mosteiro et al., 2021), in this paper we use Doc2Vec (Le & Mikolov, 2014), which generates a fixed-length vector for a piece of text of arbitrary length. In this study's context, a document is the collection of notes of one admission period of a patient. Through this method, the vector representations aims to keep the semantics within each document intact. The representations can then be fed into an ML model such as a neural network for a classification task.

3. Method

In this section, we outline the method for conducting the FL experiment for predicting inpatient violence. First the data and the processing steps are described in Sections 3.1 and 3.2, respectively. Then the setup and training procedure are described in Section 3.3, and the method for validation of the classification models is given in Section 3.4. Thereafter, more detail is provided regarding the implementation of FL in the experiment in Section 3.5.

3.1. Data

The data made available by the psychiatry ward of UMC Utrecht for this study is the violence incident dataset prepared for violence risk assessment within admitted patients by Mosteiro et al. (2020, 2021). Each data point corresponds to an admission period of a patient, and contains

Table 1
Previous studies described in Section 2. None of the studies focused on clinical texts employ FL, or vice versa. Limitations are listed, where applicable. *Generalisability* means that the performance reported has been shown or is expected not to generalise to data from new institutions.

Paper	FL	NLP	Clinical texts	Limitations
Violence Risk Assessment				
Bader and Evans (2015)				Small cohort
Raja and Azzoni (2005)				Retrospective study
Menger et al. (2019)		X	X	Generalisability
Le et al. (2018)		X	X	Predicts other instruments
Suchting et al. (2018)				Generalisability
Suicide Prediction				
Ji et al. (2019)	X	X		
Cook et al. (2016)		X		
Pestian et al. (2010)		X	X	Small cohort
Other work				
Sheller et al. (2019)	X			
Li et al. (2019)	X			
Choudhury et al. (2019)	X			
Deist et al. (2020)	X			

Table 2
Dataset characteristics. Each data point is an *admission period*, i.e., the period that a patient spends while admitted to a given nursing ward of the psychiatry department. *Age* refers to the age of the patients in the nursing ward. *Positive* and *Negative* data points are defined by whether the patient is involved in a violence incident during the first 27 days after the first day of the admission period.

Nursing ward	*Age*	*Description*	*Positive*	*Negative*	*Total*
A1	> 40	*Affective & psychotic disorders*	25	734	759
A3	15–35	*Diagnosis & early psychosis*	130	696	826
A2V	> 18	*Acute & intensive care*	167	1710	1877
A2J	12–18	*Acute & intensive care*	103	715	818
Total	–	–	425	3855	4280

the concatenation of clinical notes of a maximum of 28 days before up until and including the 1st day after admission. Based on the next 27 days following the first day of admission, the data points are labelled by whether a violence incident took place or not (positive/negative outcome). The clinical notes, which are written in Dutch, have been vectorised using Doc2Vec (Le & Mikolov, 2014), with a feature vector dimensionality of 300. No structured features such as gender or age were used, as they did not provide significant discriminatory power in previous work (Mosteiro et al., 2020). There are four nursing wards in the psychiatry department at the UMC Utrecht, and each data point belongs to one nursing ward. The characteristics of the dataset are shown on Table 2.

3.2. Data processing

To simulate two institutions (A & B) based on one dataset, and to allow for hyper-parameter tuning, a data processing procedure was designed to ensure the split-up datasets meet the following requirements. First, each of the four nursing wards is assigned to either institution A or B, in such a way that makes the numbers of data points in A and B as even as possible. Second, both datasets are split up into a train/validation and test set. The train/validation set is split up into 5 folds for cross-validation (CV). Third, between cross-validation folds themselves, and between the train/validation set and the testing set, no patient IDs may overlap; overlapping patient IDs could result in validating/testing on training data. This overlap sometimes occurs when a patient is moved to a different nursing ward, and the new nursing ward copies the notes taken from the previous nursing ward. Fourth, it should be possible to combine the folds between institutions to form patient-independent folds for federated and data-centralised training. Fifth, both testing sets may only include new data based on the admission timestamp, to ensure we test the final models on new data points exclusively. These requirements are visualised as a top-down procedure illustrated in Fig. 1.

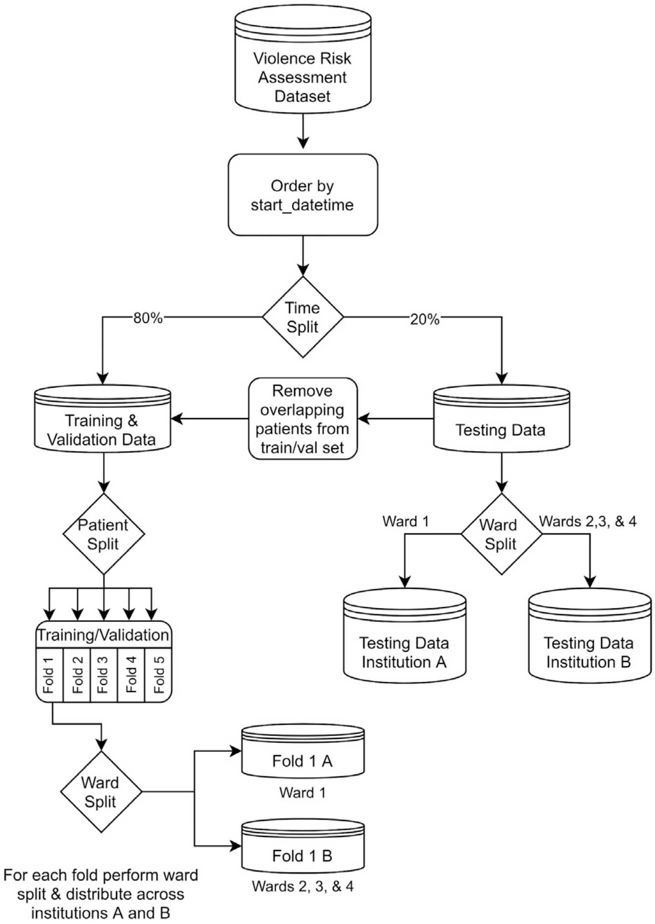

Violence Risk Assessment Dataset
Order by start_datetime
Time Split
80%
20%
Training & Validation Data
Remove overlapping patients from train/val set
Testing Data
Patient Split
Ward 1
Ward Split
Wards 2,3, & 4
Training/Validation
Fold 1
Fold 2
Fold 3
Fold 4
Fold 5
Testing Data Institution A
Testing Data Institution B
Ward Split
Fold 1 A
Ward 1
Fold 1 B
Wards 2, 3, & 4
For each fold perform ward split & distribute across institutions A and B

Fig. 1. The data processing procedure of the violence risk assessment dataset.

3.3. Treatment design

In this study, four treatments are designed and compared to test all scenarios derived from the research goals mentioned in Section 1, based on Wieringa's design cycle (Wieringa, 2014). Each treatment performs a grid search with 5-fold cross-validation (CV) to search for the best hyper-parameters for training a neural network on their respective dataset. Based on this outcome, each treatment delivers a final model by training on the data from all five folds, and is tested against a held-out testing set. These four final models are compared as part of the statistical difference-making experiment.

The difference between each treatment lies within the data it is applied on, and the training method. Two treatments are trained on data from the two simulated institutions A and B. The other two treatments, data-centralised and federated, train on data from both institutions. The data-centralised treatment trains on all data without restrictions, to show how performance would be if privacy regulations could be ignored. Therefore, it acts as a gold standard in terms of performance, as we expect the nonrestrictive training environment to deliver the best performance. The federated treatment trains a neural network on both institutional data sets through FL.

3.3.1. Classification model

The classification model used across treatments is a feed-forward neural network, consisting of an input layer, one hidden layer, and

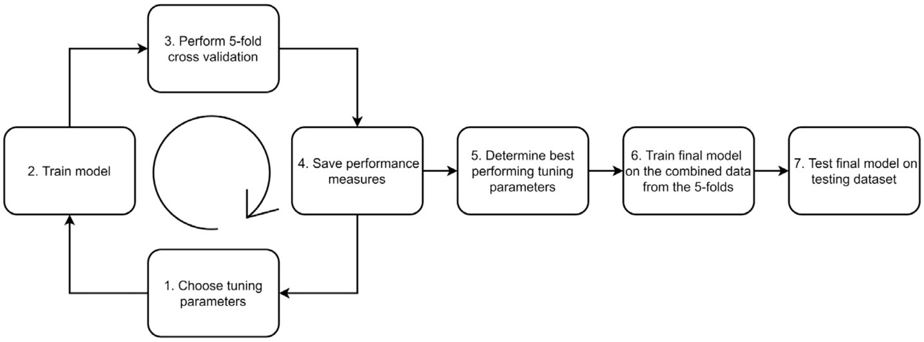

3. Perform 5-fold cross validation
2. Train model
4. Save performance measures
5. Determine best performing tuning parameters
6. Train final model on the combined data from the 5-folds
7. Test final model on testing dataset
1. Choose tuning parameters

Fig. 2. The hyper-parameter tuning cycle of each treatment.

output layer. The size of the input layer corresponds to the number of elements in the Doc2Vec vectors in the dataset (300). The hidden layer size is given by the variable h, whose values are optimised through hyper-parameter tuning. Furthermore, the hidden layer uses the Rectified Linear Unit (ReLU) activation function, chosen for its fast computation time. The output layer has a single neuron with a sigmoid activation function for providing the classification.

The model uses the Binary Cross Entropy (BCE) with logit loss function to compute its gradients. We use mini batch gradient descent. Eq. (1) is the average loss per data point for a single batch n, given input x and outcomes y.

$$l(x, y)_n = \frac{1}{T_n} \sum_{i=1}^{T_n} -[p\, y_i \cdot \log(\sigma(x_i)) + (1 - y_i) \cdot \log(1 - \sigma(x_i))] \tag{1}$$

Batch n contains T_n data points. For each data point i, we apply a sigmoid activation function σ to the input x_i. To mitigate the issue of class imbalance, when the binary outcome y_i is positive, we multiply it by a weight p equal to the ratio of negative to positive samples in the dataset.

An exponential learning rate scheduler is used for training, which updates the learning rate through $lr = lr_0 \cdot \gamma^{n_e}$, where lr_0 is the starting learning rate, γ is the amount of decay, and n_e is the number of the current epoch. A γ of 0.975 is used for the experiment. This value causes the learning rate to approximately be divided by 10 at epoch 100. It is a relatively quick drop, but as there is a computational constraint in the amount of epochs we can use, we aim for models with an initial quick convergence, and use the remaining epochs for more fine-grained model updates. The maximum number of epochs is 120.

An early stopping mechanism keeps track of the validation loss of the model at each epoch. The mechanism saves a checkpoint of the model whenever the validation loss decreases since the last overall decrease in validation loss. It means that if the validation loss has not decreased in the last seven epochs, the early stopping mechanism kicks in and stops the training. It will then load the model checkpoint which has the lowest validation loss. This checkpoint model is then used for model evaluation.

3.3.2. *Hyper-parameter tuning*

Each treatment follows the hyper-parameter tuning cycle and testing procedure as shown in Fig. 2. This aims to result in hyper-parameter values optimal for training a treatment's final model. The tuning happens through a process known as grid search in steps 1 through 4 in the Figure, where for each possible combination from a fixed set of hyper-parameters, a model is trained to reveal its respective performance. To ensure a good error estimate, the grid search is performed through 5-fold CV. Thus, for each hyper-parameter combination, five models are trained. To compute a performance measure of a combination, the ground truth labels and the predicted labels from the five models are concatenated and used as input for performance measure calculations.

The following set of hyper-parameters are used during the grid search, resulting in 36 unique combinations. These values were chosen during data exploration.

- Hidden Layer Size: [64, 128, 256, 512]
- Learning rate: [0.005, 0.001, 0.0005]
- Weight decay: [1e-3, 1e-4, 1e-5]

The performance measure used for fine-tuning is the F1-score, which assigns importance to correctly classifying the positive class. As the violence risk assessment dataset is strongly imbalanced and as correctly classifying patients exhibiting violence is deemed to be more important than correctly classifying patients who do not exhibit such behaviour, being able to accurately evaluate true positives among the positive predictions is key. When the combination with the best F1-score is determined in step 5, the final model can be trained. It uses the best hyper-parameters to train on the data from all 5-folds in step 6. After training, it is ready to be evaluated on the held out testing data at step 7.

3.4. *Treatment validation*

Each treatment is validated by testing its final model on the held out testing data containing only new data points based on the admission timestamp. It corresponds to the final step in the hyper-parameter tuning cycle (Fig. 2). This is done by feeding the testing data from institutions A, B, and the combination of the two sets into each treatment's final model. From here, performance measures per treatment are computed. As each treatment uses the same set of testing datasets, and each data point is fed in an identical order to each treatment, the performance measures as well as the raw predictions can directly be compared.

For each performance measure, confidence intervals are calculated by bootstrapping the test set for 10 000 bootstraps. Bootstrapping is a method for estimating the sample distribution for a certain statistic. It is implemented by sampling the test set with replacement to produce a new test set with a different distribution of the same size, this is done 10 000 times. From here a bootstrapped mean and confidence intervals can be calculated. The 95% confidence intervals are computed for each performance measure, treatment, and test set combination. These intervals are computed through percentiles, which is known as the percentile bootstrap method, and are compared between treatments and test sets, to provide insights into relative performance.

Each treatment's bootstrapped performance measures are compared against the federated measures by calculating the difference between the measure scores for each bootstrapped sample. Computing this difference is a method adapted from (Cumming & Finch, 2005). This too will yield a distribution with 95% confidence intervals. If the confidence interval whiskers exclude zero, then the difference is statistically significant. An important side-note for this method is that bootstrapping is ideally performed on the training set. Due to computational limitations, we performed it on the test set to see how much our specific trained models vary in their performance, when the test set is modified slightly through bootstrapping.

3.5. Federated learning implementation

The Python library PySyft is used for simulating a federated setting on a single device. We simulate two institutional devices and a central server. The training/validation algorithm implemented using PySyft is shown below in Algorithm 1. First, a central model is initialised on the server and a copy is sent to both institutions. Each institution trains the model in batches of their full local dataset. For each batch passing through the institutional models, the models are updated accordingly. After a single pass over the full dataset, the resulting institutional models are sent to the central server. Here they are aggregated by averaging the weights and biases of each layer in the neural network. The resulting averaged model is validated at each epoch to track the validation loss and performance measures. This happens by sending a copy of the averaged model back to the institutions and by feeding it the local validation sets. This results in a set of predictions and ground truth values from both institutions, which are in turn shared with the central server. The central server then computes the performance measures over the concatenation of the predicted and ground truth values from both institutions.

Algorithm 1: Federated Learning implementation

```
Result: Model, performance

model = initialise_model()
for $n_e$ in n_epochs do
    for inst in institutions do
        updated_model_inst = local_train_model(model, inst)
    model = average({updated_model_inst})
    (performance, loss) = validate(model)
    should_stop = early_stopping(loss)          // Section 3.3.1

    if should_stop then
        break
return model, performance

Function validate(model):
    for inst in institutions do
        predictions_inst, labels_inst = local_validate_model(model,
         inst)
    predictions = concatenate({predictions_inst})
    labels = concatenate({labels_inst})
    performance, loss = get_performance_and_loss(predictions,
     labels)
    return (performance, loss)

Function local_train_model(model, inst):
    updated_model = model
    for $n_b$ in n_batches do
        updated_model = train(updated_model, dataset(inst))
    return updated_model

Function local_validate_model(model, inst):
    predictions, labels = model(dataset(inst))
    return (predictions, labels)
```

4. Experimental results

4.1. Data splitting

The violence risk assessment dataset contains a total of 4005 data points after removing overlapping patients from the training/validation set. The first split assigns 856 points to the testing set, and 3149 to the training/validation set. The test set is distributed to institutions A & B based on nursing wards, giving institution A 347 and institution B 509 testing data points. Distributing the training/validation set based on the same nursing ward split assigns 1410 (of which 114 positive) and 1739 (of which 184 positive) data points to institutions A and B, respectively. Table 3 displays the distribution across the cross-validation folds illustrating the class imbalance based on the training/validation set. Nursing ward A2V is appointed to institution A, and nursing wards A1, A2J, and A3 to institution B.

Table 3
The distribution of positive and negative data points across institutional cross-validation folds.

Treatment	*Institution A*		*Institution B*		*Combined*	
Label	*Negative*	*Positive*	*Negative*	*Positive*	*Negative*	*Positive*
Fold 1	257	26	313	34	570	60
Fold 2	277	22	295	36	572	58
Fold 3	265	18	311	36	576	54
Fold 4	263	17	307	43	570	60
Fold 5	234	31	329	35	563	66
Total	1296	114	1555	184	2851	298

Table 4
Grid search cross-validation F1-scores compared to the F1-score on a treatment's own testing set.

Treatment	*CV Min F1*	*CV Mean F1*	*CV Max F1*	*F1 own Test Set*
Institution A	0.220	0.288	0.320	0.351
Institution B	0.351	0.374	0.388	0.335
Federated	0.334	0.346	0.362	0.388
Data-centralised	0.324	0.343	0.359	0.396

4.2. Grid search cross validation

Table 4 displays the relationship between cross-validation F1-scores and the F1-scores of applying the treatments to the held-out test set. The goal of the grid search is to find the combination giving the highest F1-score (CV Max F1), and in this way it aims to find a combination which has a comparably high F1-score on the held out testing set. For the final models of the data-centralised, federated, and institution A treatments, the F1-score on their own test set is higher than the CV Max F1-score. This indicates that the hyper-parameters picked during cross validation provides a decent F1-score on the held out test set. Only for institution B the opposite was true as the F1-score on its own test set is lower. In an ideal situation, a similar F1-score is preferred as the cross-validation would then provide the most realistic carry-over value.

4.3. Performance measures

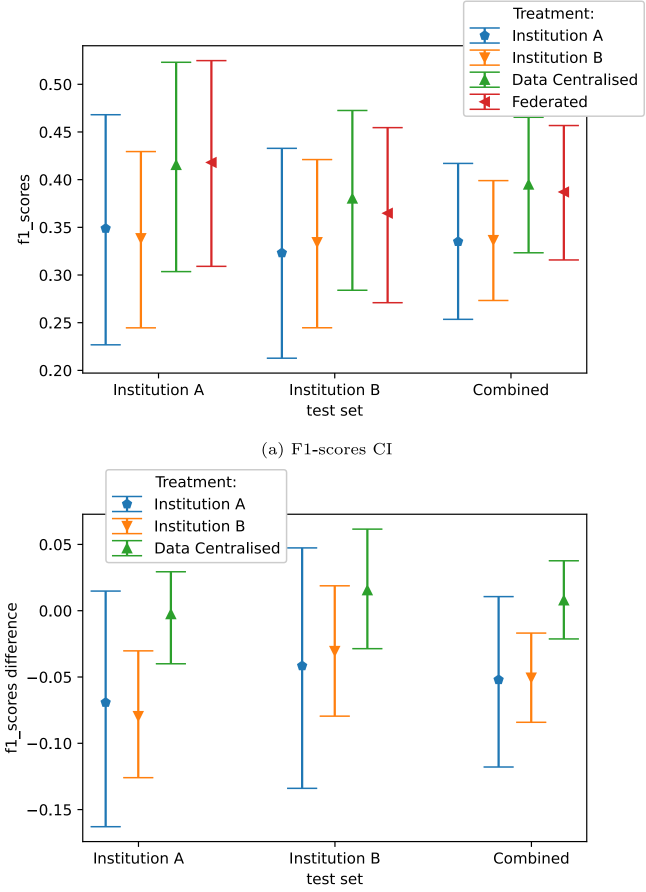
We observe in Table 5 that the federated and data-centralised models perform much alike regardless of the performance measure on our testing set. This indicates that the FL process has had no large impact on the performance of the model. When comparing the federated model to the local institutional models, we see a large gap in terms of the F1-score for all testing sets; the federated model's F1-score was remarkably higher for each set. In addition, the federated model achieves higher scores for most performance measures regardless of the testing set compared to the local models, while achieving similar scores compared to the data-centralised model. The data-centralised ROC-AUC score on the combined test set is consistent with that found in previous work (Mosteiro et al., 2021).

4.4. Bootstrapped F1-scores comparison

For a given performance measure, the confidence intervals are calculated in two ways. Both methods rely on bootstrapping of the

Table 5
Performance measures for each treatment tested on each testing set. Values in bold are the highest among a measure given a specific test set across the four treatments. F1-score, recall, and precision use a classification threshold of 0.5.

Test set	*Test set Institution A*				*Test set Institution B*				*Test set Combined*			
Treatment	*Inst A*	*Inst B*	*Fed*	*DC*	*Inst A*	*Inst B*	*Fed*	*DC*	*Inst A*	*Inst B*	*Fed*	*DC*
ROC-AUC	**0.803**	0.744	0.777	0.774	0.740	0.755	0.759	**0.762**	0.764	0.742	**0.765**	**0.765**
PR-AUC	0.278	0.274	**0.293**	0.281	0.276	0.293	0.292	**0.296**	0.270	0.281	**0.288**	**0.288**
F1	0.351	0.339	**0.419**	0.417	0.325	0.335	0.366	**0.382**	0.336	0.337	0.388	**0.396**
Recall	0.459	**0.757**	0.703	0.676	0.306	0.516	0.516	**0.532**	0.364	**0.606**	0.586	0.586
Precision	0.283	0.219	0.299	**0.301**	**0.345**	0.248	0.283	0.297	**0.313**	0.233	0.290	0.299



(a) F1-scores CI

(b) F1-scores differences compared to Federated

Fig. 3. Comparison of Confidence Intervals (95%) based on bootstrapped F1-scores. The x-axis refers to the test set. The centre points represent the mean of the bootstrapped F1-scores.

ground truth labels and the predictions based on 10 000 resamplings of the testing sets. The first method computes a performance measure for each bootstrapped sample. This results in a distribution of a given measure's scores with 10 000 data points. Then the two-tailed confidence intervals are calculated using percentiles (CI: 95%). The confidence intervals of this method using the F1-score as a performance measure, are illustrated in Fig. 3(a). The second method compares the bootstrapped distributions of the F1-scores between all non-federated treatments compared to the federated treatment. Given two treatments, the difference in a performance measure for each bootstrapped sample is calculated. These differences for a given measure provide a new distribution for which the confidence intervals are calculated (CI: 95%). This method is illustrated in Fig. 3(b).

Table 6
Confusion matrices of each treatment applied to the combined test set.

		Predicted	
		Neg	Pos
Actual	Neg	678	79
	Pos	63	36

(a) Institution A

		Predicted	
		Neg	Pos
Actual	Neg	560	197
	Pos	39	60

(b) Institution B

		Predicted	
		Neg	Pos
Actual	Neg	615	142
	Pos	41	58

(c) Federated

		Predicted	
		Neg	Pos
Actual	Neg	621	136
	Pos	41	58

(d) Data-centralised

4.5. Prediction comparisons

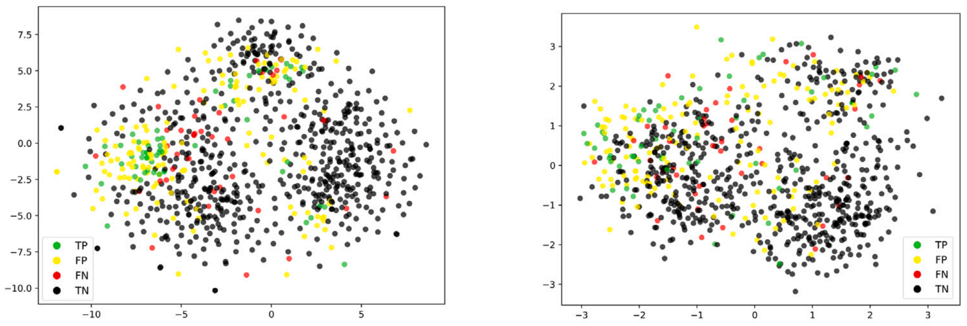
To provide a more in-depth comparison between each treatment's predictions, the confusion matrices and contingency tables are displayed in Tables 6 and 7, respectively. We observe significant differences between the predictions of the two local models and the data-centralised and federated model. Comparing the predictions of the data-centralised and federated models alone, reveals highly similar predictions. Table 8 provides the extent to which all models agree on each test sample. We observe that all models often agree with one another on the same test samples. An interesting observation is that the models also commonly misclassify a significant number of similar test samples.

To provide more insights into the data structure and the classifications, t-SNE and PCA analyses were performed on the testing dataset of the federated model. Fig. 4 shows the result of these analyses coloured by the classification of the federated model. It reveals the difficulty of the classification task as both the t-SNE and PCA show that the true positive samples are scattered across the figures.

5. Discussion

5.1. Statistical significance

When tested on the combined testing data, the federated model achieved an F1-score of 0.388 and the data-centralised achieved 0.397. This is in line with our expectations that both models would perform on par with one another. We also report on Table 5 the areas under



(a) t-SNE

(b) First two principal components

Fig. 4. PCA and t-SNE visualisations on the combined test set. The data points are labelled by True Positives (TP), False Positives (FP), False Negatives (FN), and True Negatives (TN), as predicted by the federated model.

Table 7
Contingency tables of each treatment applied to the combined test set compared to the federated model.

		Institution A	
		Correct	Incorrect
Fed	Correct	647	26
	Incorrect	67	116

(a) Institution A compared to federated

		Institution B	
		Correct	Incorrect
Fed	Correct	608	65
	Incorrect	12	171

(b) Institution B compared to federated

		Data-centralised	
		Correct	Incorrect
Fed	Correct	658	15
	Incorrect	21	162

(c) Data-centralised compared to federated

Table 8
Commonly correct and commonly misclassified predictions. Common is defined as having all models agreeing upon a single outcome.

Ground Truth	Misclassified	Correct
Neg (total nr: 757)	72	545
Pos (total nr: 99)	34	33

the Receiver Operating Characteristic (ROC-AUC) and precision–recall curve (PR-AUC). These measures indicate performance without choosing a classification threshold. On the combined test set, both models had the same ROC-AUC (0.765) and PR-AUC (0.288). Furthermore, there was not enough evidence to reject the null hypothesis that the data-centralised and federated model were similar, with a significance threshold of 0.05. While we cannot conclusively state that the federated and data-centralised model perform on par, they did so on our test dataset.

The F1-score of institution A, as observed on its own test set (0.351), was much lower than the federated model (0.419). Even though the observed differences from the experiment were large, the bootstrapped confidence intervals could not prove a statistically significant difference, given a significance threshold of 0.05. However, we can state that the federated model was better on our testing data. Analysing the F1-scores of institution B on its own test set (0.335) compared to the federated model (0.366), we see a smaller difference. For institution B, we observed a statistically significant difference when tested on the testing set from institution A and the combined set, but not when tested on the set from institution B. Again, based on the bootstrapped mean F1-scores we observe that the federated model outperformed the local model, but we cannot state with statistical significance that federated model is better.

5.2. Model differences

The confusion matrices of all models as displayed in Table 6 reveal the differences in correct and incorrect predictions between each model. Institution A delivered the most accurate model giving the lowest number of incorrect predictions (79+63 = 142). However, the model also yielded the highest number of false negatives (63) and lowest number of true positives (36), which is why it had the lowest F1-score out of all models. The data-centralised model and federated model give an identical number of false negatives (41) and true positives (58), while differing only by six samples for true negatives and false positives. The model of institution B yielded the highest number of true positives (60) at the expense of the highest number of false positives (197). The differences between institutions A and B remind us that the F1-score is not the end of the story. It will also depend on whether a practitioner favours a high number of false negatives over a high number of false positives, or vice versa.

Contingency tables reveal the relative performance and agreement of each model compared to the federated model on the combined data. Tables 7a & 7b show that there is a significant disagreement between the two local models compared to the federated model. Table 7c shows that the highest level of agreement is between the data-centralised and federated model, disagreeing only on 36 (21 + 15) data points.

To see the extent to which all models agree with one another, Table 8 displays the number of commonly correctly classified and the commonly misclassified test samples. Common, within this context, is defined as a prediction for which all models agree with one another on a given test sample. Out of all test samples (757 + 99 = 856), overall common agreement (72 + 33 + 545 + 34 = 684) between models was as high as 79.9%. A worrying detail is that all models misclassified 34 real positive samples. This high number of common false negatives begs the question of whether these data points have anything in common as to be classified incorrectly.

5.3. Limitations

The first limitation of this study is concerning the Doc2Vec model. The model was trained on all clinical notes present in the violence risk assessment dataset. There are two issues with this approach. First, it would have been more realistic to train this Doc2Vec model using FL; currently it was trained with a data-centralised approach to create the best Doc2Vec model we could with the limited size of the data set. For a real life scenario, adding federated Doc2Vec training to the pipeline is a prerequisite, for the same reasons data-centralised training is not allowed cross-institutionally. However, Doc2Vec is not compatible with PySyft at the moment. The second issue is that the Doc2Vec model has also been trained on clinical notes occurring in the testing set. When making real life predictions, it is unusual to retrain the Doc2Vec model to include the testing clinical notes and retrain the classification model. Rather, one would use the existing Doc2Vec model to immediately vectorise the newly acquired clinical notes to predict for violence. This limitation will be addressed in future work.

Another limitation is that other privacy preserving technologies were not investigated in this study. Models trained through FL can be attacked like other machine learning models, and an attacker might be able to infer details about the training data of the model. FL can be combined with other privacy preserving techniques such as differential privacy (Dwork, 2006), homomorphic encryption (Gentry, 2009), and secure multiparty computation (Yao, 1982). These techniques might alleviate additional privacy concerns, but could also negatively impact model performance. To guarantee a high level of privacy to admitted patients, combining FL with these techniques might be required.

Lastly, we observed a large variance in the performance measures and believe this can be attributed to the small test set with a low number of positive samples. Because of this, there is a high probability that a bootstrapped sample contains a skewed class distribution, which has a high impact on the variance of F1-scores.

6. Conclusions

Violence Risk Assessment (VRA), like many other clinical tasks, can be tackled with Machine Learning methods. In the psychiatry domain, NLP methods are particularly interesting thanks to the abundance of clinical notes containing valuable information. NLP models benefit enormously from bigger and more diverse datasets, as can be acquired by working across multiple institutions. Since data sharing among institutions is not possible, we have developed a Federated Learning (FL) pipeline for training an algorithm for VRA. We found no performance loss from using FL, as opposed to a data-centralised approach. Also, FL seems to improve the performance of locally trained models tested on a different dataset. To the best of our knowledge, this is the first application of FL and NLP on clinical texts.

The results suggest that there are benefits to using federated models and this should be investigated further with cross-institutional datasets. Not only would this provide insights into real-life deployment, it would also lead to more data points for training and testing and could help to decrease performance measure variance.

In future work, we plan to train document embeddings in a federated environment. Furthermore, we will investigate how FL can help solve other clinical tasks, such as text de-identification. Finally, we plan to investigate the possibility of adding other additional privacy-preserving technologies, such as differential privacy.

CRediT authorship contribution statement

Thomas Borger: Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Visualization. **Pablo Mosteiro:** Methodology, Software, Investigation, Data curation, Supervision, Project administration, Writing – review & editing. **Heysem Kaya:** Methodology, Supervision, Writing – review & editing. **Emil Rijcken:** Software, Investigation, Writing – review & editing. **Albert Ali Salah:** Methodology, Supervision, Writing – review & editing. **Floortje Scheepers:** Investigation, Resources, Funding acquisition. **Marco Spruit:** Conceptualization, Supervision, Project administration, Funding acquisition, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by Utrecht University, The Netherlands, the PsyData team from UMC Utrecht, The Netherlands, and KPMG, The Netherlands n.v. Useful discussions were held with Sebastian Mildiner and Laurens Breij.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.eswa.2022.116720. For each treatment's final model, the receiver operator characteristic (ROC) curve has been computed. The supplementary data provides these curves for each test set.

References

Almvik, R., Woods, P., & Rasmussen, K. (2000). The brøset violence checklist: Sensitivity, specificity, and interrater reliability. *Journal of Interpersonal Violence*, *15*, 1284–1296. http://dx.doi.org/10.1177/088626000015012003.

Bader, S. M., & Evans, S. E. (2015). Predictors of severe and repeated aggression in a maximum-security forensic psychiatric hospital. *International Journal of Forensic Mental Health*, *14*, 110–119. http://dx.doi.org/10.1080/14999013.2015.1045633.

Choudhury, O., Gkoulalas-Divanis, A., Salonidis, T., Sylla, I., Park, Y., Hsu, G., et al. (2019). Differential privacy-enabled federated learning for sensitive health data. CoRR, abs/1910.02578. arXiv:1910.02578.

Conroy, M. A., & Murrie, D. C. (2012). From risk assessment to risk management. In *Forensic assessment of violence risk chapter, Vol. 8* (pp. 135–152). John Wiley & Sons, Ltd., http://dx.doi.org/10.1002/9781118269671.ch8.

Cook, B. L., Progovac, A. M., Chen, P., Mullin, B., Hou, S., & Baca-Garcia, E. (2016). Novel use of natural language processing (NLP) to predict suicidal ideation and psychiatric symptoms in a text-based mental health intervention in madrid. *Computational and Mathematical Methods in Medicine*, http://dx.doi.org/10.1155/2016/8708434, Article ID 8708434, 8.

Cumming, G., & Finch, S. (2005). Inference by eye: Confidence intervals and how to read pictures of data. *The American Psychologist*, *60*, 170–180. http://dx.doi.org/10.1037/0003-066X.60.2.170.

Deist, T. M., Dankers, F. J., Ojha, P., Scott Marshall, T., Faivre-Finn, C., Masciocchi, C., et al. (2020). Distributed learning on 20 000+ lung cancer patients – The personal health train. *Radiotherapy and Oncology*, *144*, 189–200. http://dx.doi.org/10.1016/j.radonc.2019.11.019.

Douglas, K. S., Hart, S. D., Webster, C. D., Belfrage, H., Guy, L. S., & Wilson, C. M. (2014). Historical-clinical-risk management-20, version 3 (hcr-20v3): Development and overview. *International Journal of Forensic Mental Health*, *13*, 93–108. http://dx.doi.org/10.1080/14999013.2014.906519.

Dwork, C. (2006). Differential privacy. In *Proceedings of the 33rd international conference on automata, languages and programming - Volume Part II* (pp. 1–12). Berlin, Heidelberg: Springer-Verlag, http://dx.doi.org/10.1007/11787006_1.

Flikweert, L., Niessen, W., Geleijnse, G., Dekker, A., Lustberg, T., Houterman, S., et al. (2020). De personal health train in de zorg. https://pht.health-ri.nl/sites/healthtrain/files/2020-07/PHT%20in%20de%20zorgpraktijk.pdf.

Gentry, C. (2009). Fully homomorphic encryption using ideal lattices. In *Proceedings of the forty-first annual ACM symposium on theory of computing* (pp. 169–178). New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/1536414.1536440.

Havaei, F., MacPhee, M., & Lee, S. E. (2019). The effect of violence prevention strategies on perceptions of workplace safety: A study of medical-surgical and mental health nurses. *Journal of Advanced Nursing*, *75*, 1657–1666. http://dx.doi.org/10.1111/jan.13950.

Inoue, M., Tsukano, K., Muraoka, M., Kaneko, F., & Okamura, H. (2006). Psychological impact of verbal abuse and violence by patients on nurses working in psychiatric departments. *Psychiatry and Clinical Neurosciences*, *60*, 29–36. http://dx.doi.org/10.1111/j.1440-1819.2006.01457.x.

Ji, S., Long, G., Pan, S., Zhu, T., Jiang, J., Wang, S., et al. (2019). Knowledge transferring via model aggregation for online social care. arXiv:1905.07665.

Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., et al. (2021). Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, *14*, 1–210. http://dx.doi.org/10.1561/2200000083.

Konečný, J., McMahan, H. B., Yu, F. X., Richtarik, P., Suresh, A. T., & Bacon, D. (2016). Federated learning: Strategies for improving communication efficiency. In *NIPS workshop on private multi-party machine learning*. https://arxiv.org/abs/1610.05492.

Le, Q., & Mikolov, T. (2014). Distributed representations of sentences and documents. In E. P. Xing, & T. Jebara (Eds.), *Proceedings of machine learning research: vol. 32*, *Proceedings of the 31st international conference on machine learning* (pp. 1188–1196). Bejing, China: PMLR, http://proceedings.mlr.press/v32/le14.html.

Le, D. V., Montgomery, J., Kirkby, K. C., & Scanlan, J. (2018). Risk prediction using natural language processing of electronic mental health records in an inpatient forensic psychiatry setting. *Journal of Biomedical Informatics*, *86*, 49–58. http://dx.doi.org/10.1016/j.jbi.2018.08.007.

van Leeuwen, M., & Harte, J. (2017). Violence against mental health care professionals: prevalence, nature and consequences. *Journal of Forensic Psychiatry and Psychology*, *vol. 28*, 581–598. http://dx.doi.org/10.1080/14789949.2015.1012533.

Lehne, M., Sass, J., Essenwanger, A., Schepers, J., & Thun, S. (2019). Why digital medicine depends on interoperability. *Npj Digital Medicine*, *2*, 79. http://dx.doi.org/10.1038/s41746-019-0158-1.

Li, W., Milletarì, D., Rieke, N., Hancox, J., Zhu, W., Baust, M., et al. (2019). Privacy-preserving federated brain tumour segmentation. In H.-I. Suk, M. Liu, P. Yan, C. Lian (Eds.), *Machine learning in medical imaging* (pp. 133–141). Cham: http://dx.doi.org/10.1007/978-3-030-32692-0_16.

McMahan, H. B., Moore, E., Ramage, D., & y Arcas, B. A. (2016). Federated learning of deep networks using model averaging. CoRR, abs/1602.05629. arXiv:1602.05629.

Menger, V., Scheepers, F., & Spruit, M. (2018). Comparing deep learning and classical machine learning approaches for predicting inpatient violence incidents from clinical text. *Applied Sciences*, *vol. 8*(981), http://dx.doi.org/10.3390/app8060981.

Menger, V., Spruit, M., van Est, R., Nap, E., & Scheepers, F. (2019). Machine learning approach to inpatient violence risk assessment using routinely collected clinical notes in electronic health records. *JAMA Network Open*, *vol. 2*, Article e196709–e196709. http://dx.doi.org/10.1001/jamanetworkopen.2019.6709.

Mosteiro, P., Rijcken, E., Zervanou, K., Kaymak, U., Scheepers, F., & Spruit, M. (2020). Making sense of violence risk predictions using clinical notes. In Z. Huang, S. Siuly, H. Wang, R. Zhou, & Y. Zhang (Eds.), *Health information science* (pp. 3–14). Cham: Springer International Publishing, http://dx.doi.org/10.1007/978-3-030-61951-0_1.

Mosteiro, P., Rijcken, E., Zervanou, K., Kaymak, U., Scheepers, F., & Spruit, M. (2021). Machine learning for violence risk assessment using dutch clinical notes. *Journal of Artificial Intelligence for Medical Sciences*, *vol. 2*, 44–54. http://dx.doi.org/10.2991/jaims.d.210225.001.

Nijman, H., Bowers, L., Oud, N., & Jansen, G. (2005). Psychiatric nurses' experiences with inpatient aggression. *Aggressive Behaviour*, *vol. 31*, 217–227. http://dx.doi.org/10.1002/ab.20038.

Ogloff, J. R. P., & Daffern, M. (2006). The dynamic appraisal of situational aggression. *Behavioral Sciences & the Law*, *vol. 24*, 799–813. http://dx.doi.org/10.1002/bsl.741.

Pestian, J., Nasrallah, H., Matykiewicz, P., Bennett, A., & Leenaars, A. (2010). Suicide note classification using natural language processing: A content analysis. *Biomedical Informatics Insights*, *vol. 3*, http://dx.doi.org/10.4137/BII.S4706, BII.S4706, PMID: 21643548.

Raja, M., & Azzoni, A. (2005). Hostility and violence of acute psychiatric inpatients. *Clinical Practice and Epidemiology in Mental Health*, *vol. 1*, 11. http://dx.doi.org/10.1186/1745-0179-1-11.

Reddi, S. J., Charles, Z., Zaheer, M., Garrett, Z., Rush, K., Konečný, J., et al. (2021). Adaptive federated optimization. In *International conference on learning representations*.

Sheller, M. J., Reina, G. A., Edwards, B., Martin, J., & Bakas, S. (2019). Multi-institutional deep learning modeling without sharing patient data: A feasibility study on brain tumor segmentation. In A. Crimi, S. Bakas, H. Kuijf, F. Keyvan, M. Reyes, & T. van Walsum (Eds.), *Brainlesion: Glioma, multiple sclerosis, stroke and traumatic brain injuries* (pp. 92–104). Cham: http://dx.doi.org/10.1007/978-3-030-11723-8_9.

Singh, J. P., Desmarais, S. L., Hurducas, C., Arbach-Lucioni, K., Condemarin, C., Dean, K., et al. (2014). International perspectives on the practical application of violence risk assessment. *International Journal of Forensic Mental Health*, *vol. 13*, 193–206. http://dx.doi.org/10.1080/14999013.2014.922141.

Suchting, R., Green, C. E., Glazier, S. M., & Lane, S. D. (2018). A data science approach to predicting patient aggressive events in a psychiatric hospital. *Psychiatry Research*, *vol. 268*, 217–222. http://dx.doi.org/10.1016/j.psychres.2018.07.004.

Webster, C. D., Nicholls, T. L., Martin, M.-L., Desmarais, S. L., & Brink, J. (2006). Short-term assessment of risk and treatability (start): the case for a new structured professional judgment scheme. *Behavioral Sciences & the Law*, *vol. 24*, 747–766. http://dx.doi.org/10.1002/bsl.737.

Wieringa, R. J. (2014). Design science methodology: For information systems and software engineering. http://dx.doi.org/10.1007/978-3-662-43839-8.

Yao, A. C. (1982). Protocols for secure computations. In *23rd Annual symposium on foundations of computer science* (pp. 160–164). http://dx.doi.org/10.1109/SFCS.1982.38.