DOI: 10.15514/ISPRAS-2024-36(1)-3

CC BY

Виды атак на федеративные нейросети и способы защиты

В.А. Костенко, ORCID: 0000-0002-7895-2322 <*kostmsu@gmail.com*>
А.Е. Селезнева, ORCID: 0009-0005-8480-8182 <*alice.in.moskow@gmail.com*>

Московский государственный университет имени М.В. Ломоносова,
Россия, 119991, Москва, Ленинские горы, д. 1.

Аннотация. Федеративное обучение — это технология обучения с сохранением конфиденциальности в распределенных системах хранения данных. Такое обучение позволяет создать общую модель прогнозирования, сохраняя все данные в своих системах хранения. В обучении общей модели учувствуют несколько устройств, при этом каждое устройство имеет свои уникальные данные, на которых обучается нейросеть. Взаимодействие устройств происходит только для корректировки весов общей модели. После чего, обновленная модель передается на все устройства. Обучение на нескольких устройствах рождает множество возможностей для атак на этот тип сетей.

Ключевые слова: нейронные сети; федеративные классификаторы; безопасность нейронных сетей; атаки на нейросети; защита нейросетей; атаки отравлением; атаки уклонения; атаки логического вывода; атаки восстановления данных.

Для цитирования: Костенко В.А., Селезнева А.Е. Виды атак на федеративные нейросети и способы защиты. Труды ИСП РАН, том 36, вып. 1, 2024 г., стр. 35–44. DOI: 10.15514/ISPRAS–2024–36(1)–3.

Types of Attacks on Federated Neural Networks and Methods of Protection

V.A. Kostenko, ORCID: 0000-0002-7895-2322 <kostmsu@gmail.com>
A.E. Selezneva, ORCID: 0009-0005-8480-8182 <alice.in.moskow@gmail.com>

Lomonosov Moscow State University,
GSP-1, Leninskie Gory, Moscow, 119991, Russia.

Abstract. Federated learning is a technology for privacy-preserving learning in distributed storage systems. This training allows you to create a general forecasting model, storing all the data in your storage systems. Several devices take part in training the general model, and each device has its own unique data on which the neural network is trained. The interaction of devices occurs only to adjust the weights of the general model. After which, the updated model is transmitted to all devices. Training on multiple devices creates many attack opportunities against this type of network. After training on a local device, model data is sent via some type of communication to a central server or global model. Therefore, vulnerabilities in a federated network are possible not only at the training stage on a separate device, but also at the data exchange stage. All this together increases the number of possible vulnerabilities of federated neural networks. As is known, not only neural networks, but also other models can be used to build federated classifiers. Therefore, the types of attacks directly on the network also depend on the type of model used. Federated neural networks are a rather complex design, different from neural networks and other classifiers, which can be vulnerable to various types of attacks because training occurs on different devices, and both neural networks and simpler algorithms can be used. In addition, it is necessary to ensure data transfer between devices. All attacks come down to several main types that exploit

classifier vulnerabilities. It is possible to implement protection against attacks by improving the architecture of the classifier itself and paying attention to data encryption.

Keywords: machine learning; federal neural networks; neural network attacks; neural network protections; poisoning attacks; evasion attacks; logical inference attacks; data recovery attacks.

For citation: Kostenko V.A., Selezneva A.E. Types of attacks on federated neural networks and methods of protection. *Trudy ISP RAN/Proc. ISP RAS*, vol. 36, issue 1, 2024. pp. 35-44 (in Russian). DOI: 10.15514/ISPRAS-2024-36(1)-3.

1. Введение

Федеративные нейросети обучаются распределенно и позволяют множеству участников вместе обучать единую модель, не раскрывая своих уникальных наборов данных.

Существует несколько подходов для создания федеративных сетей. Один из подходов – это обучение под управлением центрального сервера. Такая архитектура нейронной сети подразумевает, что данные обучения хранятся на каждом устройстве только локально. Эти данные не передаются ни другим устройствам, ни центральному серверу. Центральный сервер принимает на вход только обновленные параметры модели машинного обучения, тем самым, не взаимодействуя с данными, на которых обучались локальные модели [1].

Другой вид федеративного обучения – децентрализованное обучение. Так же обучается глобальная модель, но взаимодействие между узлами сети происходит непосредственно друг с другом, без участия центрального сервера. Данные обучения каждого устройства так же остаются локальными, передаются только параметры модели машинного обучения [1].

Один из примеров федеративных нейросетей – Google клавиатура на устройствах Android. По умолчанию на устройствах Android установлена клавиатура со стандартными подсказками при вводе слов и при составлении предложений. Со временем, для каждого пользователя формируется собственный набор предсказания слов. Этот набор становится уникальным и зависит только от того, насколько часто конкретный пользователь использовал слова в определенном контексте. Клавиатура обучается локально на каждом устройстве, сервер получает информацию только о данных модели обучения, без каких-либо обучающих данных [2].

Федеративные нейросети обучаются на множестве устройств, на каждом из которых происходит собственное обучение. После обучения на локальном устройстве, данные модели пересылаются по каким-либо видам связи на центральный сервер или глобальную модель. Поэтому уязвимости федеративной сети возможны не только на этапе обучения на отдельном устройстве, но и на этапе обмена данными. Все это в совокупности увеличивает количество возможных уязвимостей федеративных нейросетей.

Как известно, для построения федеративных классификаторов могут быть использованы не только нейросети, но и другие модели и методы. Поэтому виды атак непосредственно на сеть также зависят от типа используемых методов и моделей. В данной работе будут рассматриваться федеративные классификаторы, построенные на основе нейросетей.

2. Виды атак и способы защиты

Атаки на федеративные классификаторы могут иметь различные цели – атаки непосредственно на локальное устройство с целью его захвата, получение данных о самой модели или же целью может выступать сам классификатор. Уязвимы могут быть каналы передачи данных и большинство локальных устройств вместе с центральным сервером, если он присутствует у классификатора. Под классификатором понимается обученная модель, управляемая центральным или распределенным сервером, которая обновляется за счет локальных устройств с децентрализованными данными обучения [1].

На рис. 1 схематично отображены уязвимые моменты в работе федеративного классификатора. Можно увидеть, что атаки бывают нацелены на локальные устройства, например, целью может выступать – отравление данных одного или нескольких устройств. Далее атаки переходят на попытки отравления модели или моделей – как локальных устройств, так и центрального классификатора. При взаимодействии локальных устройств и центрального классификатора, возможно, как прослушивание любых передаваемых данных, так и попытки сделать выводы о самих данных и (возможны) нарушения конфиденциальных данных [3].

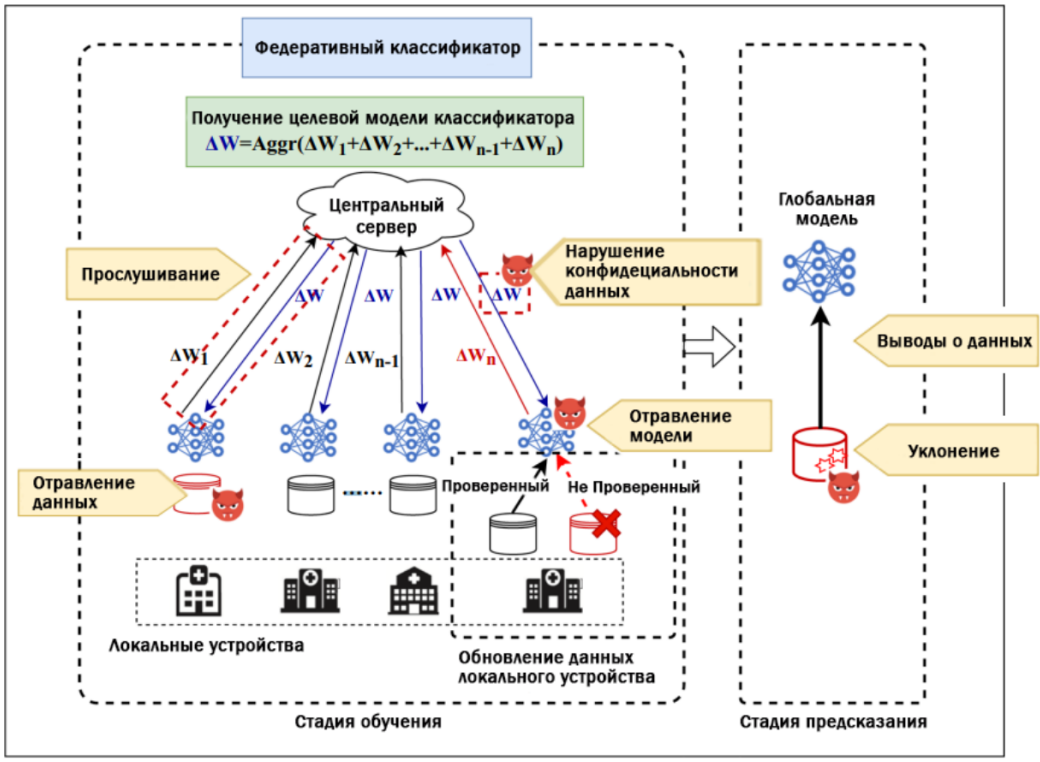

Рис. 1. Схема федеративного классификатора, обозначение уязвимостей и возможных атак
Fig. 1. Scheme of federated neural networks, designation of vulnerabilities and possible attacks

2.1 Атаки отравлением

Под отравлением данных понимается злонамеренное изменение или добавление данных в обучающую выборку, что в конечном итоге, приводит к захвату локального устройства или сервера. Атаки могут быть случайными или целенаправленными, как пример, можно привести бэкдор атаки [3]. Бэкдор атаки – это форма состязательных атак на нейронные сети, во время которых злоумышленник использует зараженные данные. Зараженные данные – это данные, которые способны повлиять на целостность и качество модели или классификатора [1]. Зараженные данные используются злоумышленником для обучения модели. Сначала подкладываются зараженные данные в процесс обучения модели, а затем, злоумышленник в любой момент может активировать атаку, с помощью определенного шаблона-триггера. Активировать атаку – значит передать на вход классификатору определенные данные, то есть триггер. Триггер связан с заранее зараженными данными. Зараженные данные заставляют классификатор обучаться неправильно, при этом в результате, злоумышленник получает ожидаемый результат [4].

Отравление глобальной модели осуществляется злоумышленником, который контролирует небольшое количество вредоносных агентов (локальных устройств). Отравление происходит путем передачи глобальной модели параметров локальных сетей, которые обучены на отравленных данных. Цель такой атаки -заставить глобальную модель ошибочно классифицировать набор выбранных входных данных с высокой степенью достоверности. Атаки отравлением можно поделить на чистые и грязные [5].

Атаки чистого отравления предполагают, что злоумышленник не может изменить весовые коэффициенты модели с помощью обучающих данных. В модели существует процесс, с помощью которого, данные можно валидировать, как принадлежащие к какому-либо определенному классу. Атаки с помощью грязного отравления подразумевают введение в обучающую выборку нескольких копий примеров, которые злоумышленник хочет неправильно классифицировать с указанием желаемого класса [6].

Возможно отравление как самих данных, например, добавление шума, так и отравление целой модели. В работе [5] отмечалось, что атаки грязного отравления данных не особенно эффективны для отравления всего федеративного классификатора, в отличие от отравления локального классификатора. Отравленный классификатор будет продолжать функционировать и посылать невалидные веса в центральную модель.

В федеративном классификаторе захват нескольких устройств для отравления модели центрального сервера так же является одной из возможных атак [7].

Возможны другие виды атак с помощью отравления данных. Рассмотрим атаки отравления под видом возможных ошибок классификатора. Цель злоумышленника – реализовать неправильную классификацию. Этот вид атаки возможен только для многоклассовой классификации. Атака может быть как целенаправленной, так и случайной. Особенность этих атак заключается в том, что они имитируют ошибки при нормальном функционировании сети [3].

Еще один вид атаки с помощью отравления данных – это атака на основе отравления градиента. Эта атака может быть весьма неэффективна. Однако, если использовать обратный градиент, то процедуру обучения можно проследить в обратном направлении, а затем вычислить значения градиента в обратном порядке. Это дает возможность осуществить атаку для большинства градиентных методов, таких как – градиентный спуск с фиксированным шагом [6].

2.2 Способы защиты от отравляющих атак

Первый способ борьбы с атаками отравления для федеративного классификатора – это независимая проверка центральным сервером точности глобальной модели с помощью тестовой выборки. Центральный сервер также может выполнять статистические проверки, сравнивая отличия обновлений локальных устройств между собой, тем самым он способен обнаружить вредоносное устройство [5].

Второй способ описан в работе [8], где было показано, что для атак отравления могут быть эффективны методы защиты с помощью дифференциальной конфиденциальности. Защита заключается в добавлении случайного шума к градиентам модели каждого локального устройства, причем устройства сами могут контролировать эти помехи. Это позволяет в федеративном классификаторе добиться полной уникальности добавления шума на каждом устройстве [9].

Третий способ – это добавление валидации на центральном сервере. В этом случае, отравление центрального сервера возможно только с захватом более 50% работающих устройств [10].

2.3 Атака уклонения

Атака уклонения – это тип атаки, при котором злоумышленник пытается обмануть целевую модель классификатора путем создания конкретных образцов, называемых состязательными примерами. Обычно небольшой шум, добавленный к входным данным у локальной модели, не может быть обнаружен людьми. Это приводит к тому, что такая модель получает неправильные результаты классификации [11]. Обычно это происходит на этапе прогнозирования, когда модель закончила обучение. Результат такого рода атаки – неправильная классификация у глобального классификатора. Одна из главных особенностей этой атаки – широкий разброс возможных опасностей. В качестве примера, можно рассмотреть неправильное распознавание дорожных знаков беспилотными автомобилями, некорректное распознавание лиц, неверная работа системы распознавания речи и прочее. Атака уклонения – это атака на целостность за счет подмены модели [12].

2.4 Способы защиты от атак уклонения

Первый способ защиты – эмпирическая защита, при которой, например, предполагается предварительная обработка исходных данных и преобразование признаков. Этот способ эффективен только если злоумышленнику не будет известно об этом механизме защиты. Другой способ эмпирической защиты – сокрытие полной информации о локальной модели. Используется слияние локальных моделей, при передаче данных глобальной модели, при наложении градиентной маски и другие методы [13].

Второй способ защиты – добавление случайного сглаживания исходного изображения с помощью Гауссовского шума на этапе классификации. Принуждение локального классификатора правильно классифицировать входные данные с учетом добавленного шума – тоже улучшит защищенность модели. При применении этого способа защиты – наблюдается устойчивость к атакам уклонения [14].

2.5 Атаки логического вывода с нарушением приватности

Можно выделить несколько типов логических атак с нарушением приватности.

Первый, нацелен на получение данных о том, является ли конкретный объект частью обучающей выборки. Злоумышленник обучает несколько теневых моделей для имитации поведения локального классификатора и обучает собственную модель на основе данных, полученных из выходных данных теневых моделей [15]. Он может использовать как черный ящик (злоумышленник ограничивается произвольным набором входных данных, пытаясь сделать на основе этого какие-либо выводы), так и белый ящик (злоумышленник получает доступ к самой модели, включая ее параметры, которые необходимы для классификации, поэтому, для любых входных данных он может получить помимо непосредственного результата, все промежуточные вычисления модели) [3]. Также злоумышленник может делать оценку принадлежности объекта к определённому классу, если классификатор осуществляет классификацию объектов [12].

Второй тип атаки реализуется, если злоумышленник нацелен на извлечение самой модели. Он пытается получить информацию о модели с помощью циклических запросов. С помощью атаки с извлечением параметров модели злоумышленник восстанавливает параметры модели за счет доступа к классификатору. Основная задача атакующего – построить состязательную модель, то есть модель, которая состоит из двух сетей: первая генерирует образцы (эту роль выполняет злоумышленник), вторая пытается их классифицировать (атакуемый классификатор) [16].

Третий тип атак – прослушивающие атаки на вывод свойств. Одна из уязвимостей федеративного классификатора – передача данных модели между устройствами. Возможны атаки с инверсией модели, которые используют, в основном, некоторые прикладные

системы, если они используются федеративным классификатором для получения или обмена данными о модели. С помощью этой информации злоумышленники могут проанализировать модель, чтобы получить соответствующую информацию об исходных данных, например, получить свойства модели [17]. Злоумышленники также могут прослушивать канал связи, чтобы получить любую информацию о самом федеративном классификаторе или о локальной модели [12].

Прослушивающие атаки можно разделить по вмешательству в работу классификатора на пассивные и активные. Во время пассивных атак злоумышленник только наблюдает за обновлением существующей модели и параллельно обучает собственный классификатор. На основе полученной модели он может сделать выводы о существующей модели. Во время активного способа атаки злоумышленник пытается обмануть модель, чтобы лучше получить целевые данные атакуемого объекта [18].

2.6 Способы защиты от логических атак

В качестве защиты рассматриваются методы защиты непосредственно передаваемых данных. Предполагается, что злоумышленник может каким-то образом получить доступ к каналу связи. Обеспечить безопасность передачи данных можно за счет, например, выбора безопасных протоколов с шифрованием. Мы рассмотрим только способы защиты непосредственно классификатора, без защиты канала передачи информации.

Первая возможная защита – это защита структуры модели, то есть возможно снижение чувствительности модели к обучающим выборкам и к переобучению модели. Следующий вариант – это защита от обфускации данных классификатора, способ добиться безопасности через неясность, например, введением в заблуждение с помощью изменения выходных данных модели [3].

Еще один способ добиться конфиденциальности данных – обфускация любых данных путем добавления случайного шума при обучении как к самим данным, так и к целевой функции, градиентам, параметрам и выходным данным. Этот метод снижает производительность, но повышает безопасность классификатора. Предыдущий способ защиты является частным случаем данного [19].

2.7 Атаки восстановления данных

Атаки восстановления данных могут позволить получить исходную информацию об обучающей выборке путем сбора любой доступной информации о классификаторе, например, предсказываемых значений, параметров и градиентов модели. Один из возможных сценариев атаки – это использование генеративно-состязательных сетей, за счет которых может быть получен доступ к данным других участников. При такой атаке злоумышленнику не обязательно иметь полное представление о федеративном классификаторе [20].

Другой тип атаки предполагает возможность восстановления исходных данных на основе информации о градиенте модели [21]. Градиент вычисляется через обратное распространение ошибки от последнего слоя к первому. Градиент конкретного слоя получается с использованием функции активации этого слоя и полученной ошибки от его верхнего слоя. Если следить за градиентами, можно вывести значения признаков, которые получаются непосредственно из исходных обучающих данных [18]. Во время использования градиента может происходить синтез пар фиктивных входных данных и меток путем сопоставления их фиктивных градиентов, близких к реальным через задачу оптимизации:

$$\arg\min_{x} || \nabla_\theta L_\theta(x, y) - \nabla_\theta L_\theta(x^*, y)||^2$$

где (x, y) – фиктивные данные и метки, (x*, y) – реальные данные и метки переданного градиента, $\nabla_\theta L_\theta(x^*, y)$ – передаваемый градиент, $\nabla_\theta L_\theta(x, y)$ – градиент фиктивных данных и

меток, $L_\theta(x, y)$ и $L_\theta(x^*, y)$ – некоторые функции потерь. Таким образом, можно восстановить исходные данные за счет перебора фиктивных данных и меток [3].

Еще один тип атаки восстановления данных нацелен на инвертирование градиентов. Эта атака подразумевает наличие полного доступа злоумышленника к данным модели. Инвертирование градиентов порождает противоположные значения, за счет чего модель перестает корректно работать [22].

2.8 Виды защиты от атак, нацеленных на восстановление данных

Один из способов защиты данных от атак, нацеленных на восстановление данных – использование возможности сжатия или разрежения градиентов при их передаче. Такой способ защиты позволяет защитить информацию при передаче градиентов моделей между устройствами. Другой способ защиты – отбрасывание случайных значений градиентов. Поскольку злоумышленник не знает, какой параметр отброшен и отброшен ли вообще, то задача восстановления данных для него сильно осложняется, поскольку ему придется использовать неполные данные. Недостаток такой защиты заключается в том, что небольшое изменение градиента ухудшает производительность классификаторов, как по времени, так и возможно по памяти [18].

Еще один способ защиты – это шифрование градиентов. Способы шифрования можно разделить на гомоморфное шифрование и безопасные многосторонние вычисления. Гомоморфное шифрование позволяет кодировать и обрабатывать зашифрованные данные так, что при этом расшифрованный результат будет эквивалентен результату, полученному на исходных данных. Данный алгоритм не изменяет исходную информацию, поэтому гарантируется отсутствие потери точности [23]. Безопасные многосторонние вычисления позволяют отдельным устройствам выполнять совместные вычисления на основе своих исходных данных, не раскрывая собственной информации другим участникам. Таким образом, они обеспечивают высокую степень конфиденциальности. Но получается, что каждое устройство должно согласовывать свои действие с соседними, что может негативно сказываться на производительности всего классификатора [24].

Еще один способ защиты был описан выше – добавление случайного шума к любым данным [25].

3. Заключение

Федеративные нейросети – это довольно сложная конструкция, отличающаяся от нейронных сетей и других классификаторов, которые могут быть уязвимы для различного рода атак, потому что обучение происходит на различных устройствах. Могут использоваться как нейросети, так и более простые алгоритмы. Помимо этого, необходимо обеспечивать передачу данных между устройствами.

Все атаки сводятся к нескольким основным типам, которые используют уязвимости классификатора. Можно реализовать защиту от атак с помощью усовершенствования архитектуры самого классификатора или уделить внимание шифрованию данных.

Список литературы / References

[1]. Kairouz, Peter; Brendan McMahan, H.; Avent, Brendan; Bellet, Aurélien; Bennis, Mehdi; Arjun Nitin Bhagoji; Bonawitz, Keith; Charles, Zachary; Cormode, Graham; Cummings, Rachel; D'Oliveira, Rafael G. L.; Salim El Rouayheb; Evans, David; Gardner, Josh; Garrett, Zachary; Gascón, Adrià; Ghazi, Badih; Gibbons, Phillip B.; Gruteser, Marco; Harchaoui, Zaid; He, Chaoyang; He, Lie; Huo, Zhouyuan; Hutchinson, Ben; Hsu, Justin; Jaggi, Martin; Javidi, Tara; Joshi, Gauri; Khodak, Mikhail; et al. (10 December 2019). "Advances and Open Problems in Federated Learning". arXiv:1912.04977, DOI: 10.48550/arXiv.1912.04977.

[2]. Federated Learning: Collaborative Machine Learning without Centralized Training Data (online) https://ai.googleblog.com/2017/04/federated-learning-collaborative.html — 01.12.2023.
[3]. Threats, attacks and defenses to federated learning: issues, taxonomy and perspectives - Peng Liu, Xiangru Xu, Wen Wang, Cybersecurity 5, 4 (2022), DOI: 10.1186/s42400-021-00105-6.
[4]. Aniruddha Saha; Akshayvarun Subramanya; Hamed Pirsiavash; (2019), - "Hidden Trigger Backdoor Attacks" - arXiv:1910.00033v2, DOI: 10.48550/arXiv.1910.00033.
[5]. Bhagoji AN, Chakraborty S, Mittal P, Calo SB - Analyzing federated learning through an adversarial lens. In: Chaudhuri K, Salakhutdinov R (eds) Proceedings of the 36th international conference on machine learning, ICML 2019, 9–15 June 2019, Long Beach, California, USA, volume 97 of proceedings of machine learning research, pp 634–643, PMLR 97:634-643, 2019.
[6]. Chen, X., Liu, C., Li, B., Lu, K., and Song, D. Targeted backdoor attacks on deep learning systems using data poisoning, arXiv:1712.05526, 2017a, DOI: 10.48550/arXiv.1712.05526.
[7]. Clement Fung, Chris J. M. Yoon, and Ivan Beschastnikh. The limitations of federated learning in sybil settings. In 23rd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2020), pp. 301–316, San Sebastian, October 2020. USENIX Association. ISBN 978-1-939133-18-2.
[8]. Naseri M, Hayes J, De Cristofaro E (2020) Toward robustness and privacy in federated learning: experimenting with local and central differential privacy. CoRR arXiv:2009.03561, DOI: 10.48550/arXiv.2009.03561
[9]. Abadi M, Chu A, Goodfellow I, McMahan HB, Mironov I, Talwar K, Zhang L (2016) Deep learning with differential privacy. In: Proceedings of the 2016 ACM SIGSAC conference on computer and communications security, pp 308–318, DOI: 10.48550/arXiv.1607.00133.
[10]. Kostenko V.A., Tankaev I.R., Federated Learning Using Simple Voting Scheme; 2022 - ISSN 1060-992X
[11]. Szegedy C, Zaremba W, Sutskever I, Bruna J, Erhan D, Goodfellow IJ, Fergus R (2014) Intriguing properties of neural networks. In: Bengio Y, LeCun Y (eds) 2nd international conference on learning representations, ICLR 2014, Banff, AB, Canada, April 14–16, 2014, conference track proceedings, arXiv:1312.6199, DOI: 10.48550/arXiv.1312.6199.
[12]. Yao Chen1, Yijie Gui1, Hong Lin1, Wensheng Gan1,2∗, Yongdong Wu; Federated Learning Attacks and Defenses: A Survey – 2022; arXiv:2211.14952v1, DOI: 10.48550/arXiv.2211.14952.
[13]. JI Shou-Ling, Du Tian-Yu, Li Jin-Feng, Shen Chao, Li Bo - Security and privacy of machine learning models: a survey. Ruan Jian Xue Bao/J Softw 32(1):41–67, 2021, DOI: 10.13328/j.cnki.jos.006131.
[14]. Cohen JM, Rosenfeld E, Kolter JZ - Certified adversarial robustness via randomized smoothing. In: Chaudhuri K, Salakhutdinov R (eds) Proceedings of the 36th international conference on machine learning, ICML 2019, 9–15 June 2019, Long Beach, California, USA, volume 97 of proceedings of machine learning research. PMLR, pp 1310–1320, DOI: 10.48550/arXiv.1902.02918.
[15]. Nasr M, Shokri R, Houmansadr A (2019) Comprehensive privacy analysis of deep learning: passive and active white-box inference attacks against centralized and federated learning. In: 2019 IEEE symposium on security and privacy (SP). IEEE, pp 739–753, DOI: 10.1109/SP.2019.00065.
[16]. Ren K, Meng QR, Yan SK - Survey of artificial intelligence data security and privacy protection. Chin J Netw Inf Secur 7(1):1–10, 2021, DOI: 10.11959/j.issn.2096-109x.2021001.
[17]. Jayaraman B, Evans D (2019) Evaluating differentially private machine learning in practice. In: Heninger N, Traynor P (eds) 28th USENIX security symposium, USENIX security 2019, Santa Clara, CA, USA, August 14–16, 2019. USENIX Association, pp 1895–1912, DOI: 10.48550/arXiv.1902.08874.
[18]. Melis L, Song C, De Cristofaro E, Shmatikov V (2019) Exploiting unintended feature leakage in collaborative learning. In: 2019 IEEE symposium on security and privacy (SP). IEEE, pp 691–706. DOI: 10.48550/arXiv.1805.04049.
[19]. Papernot N, McDaniel PD, Sinha A, Wellman MP (2018) Sok: security and privacy in machine learning. In: 2018 IEEE European symposium on security and privacy, EuroS&P 2018, London, United Kingdom, April 24–26, 2018. IEEE, pp 399–414. https://doi.org/10.1109/EuroSP.2018.00035
[20]. B. Hitaj, G. Ateniese, and F. Perez-Cruz, "Deep models under the gan: information leakage from collaborative deep learning," in ACM SIGSAC Conference on Computer and Communications Security, 2017, pp. 603–618.
[21]. L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," Advances in Neural Information Processing Systems, vol. 32, 2019.
[22]. Geiping J, Bauermeister H, Dröge H, Moeller M (2020) Inverting gradients—how easy is it to break privacy in federated learning? arXiv preprint arXiv:2003.14053, DOI: 10.48550/arXiv.2003.14053.
[23]. Fang H, Qian Q (2021) Privacy preserving machine learning with homomorphic encryption and federated learning. Future Internet 13(4):94, DOI: 10.3390/fi13040094.

[24]. Li Y, Zhou Y, Jolfaei A, Dongjin Y, Gaochao X, Zheng X (2020) Privacy-preserving federated learning framework based on chained secure multi-party computing. IEEE Internet Things J. DOI: 10.1109/JIOT.2020.3022911.

[25]. Lyu L, Yu H, Ma X, Sun L, Zhao J, Yang Q, Yu PS (2020) Privacy and robustness in federated learning: attacks and defenses. arXiv preprint arXiv:2012.06337, DOI: 10.48550/arXiv.2012.06337.

Информация об авторах / Information about authors

Валерий Алексеевич КОСТЕНКО – кандидат технических наук, доцент кафедры автоматизации систем вычислительных комплексов (АСВК) факультета Вычислительной математики и кибернетики (ВМК), Московский государственный университет имени М.В. Ломоносова. Сфера научных интересов: теория расписаний, методы машинного обучения, вычислительные системы реального времени, планирование вычислений в центре обработки данных.

Valery Alekseevich KOSTENKO – Cand. Sci. (Tech.), Associate Professor of the Department of Automation of Computer Complex Systems, Faculty of Computational Mathematics and Cybernetics, Lomonosov Moscow State University. Research interests: scheduling theory, machine learning methods, real-time computing systems, scheduling calculations in a data center.

Алиса Евгеньевна СЕЛЕЗНЕВА – студентка факультета Вычислительной математики и кибернетики (ВМК), Московский государственный университет имени М.В. Ломоносова. Сфера научных интересов: модели и методы машинного обучения, федеративные нейросети, планирование вычислений в центре обработки данных.

Alisa Evgenievna SELEZNEVA – student of the Faculty of Computational Mathematics and Cybernetics, Lomonosov Moscow State University. Research interests: machine learning models and methods, federated neural networks, scheduling calculations in a data center.