Federated Multilingual Models for Medical Transcript Analysis

Andre Manoel[*] ANDRE.MANOEL@MICROSOFT.COM
Mirian del Carmen Hipólito Garcia[*] MIRIANH@MICROSOFT.COM
Tal Baumel[*] TAL.BAUMEL@MICROSOFT.COM
Shize Su SHIZE.SU@MICROSOFT.COM
Jialei Chen JIALEICHEN@MICROSOFT.COM
Robert Sim RSIM@MICROSOFT.COM
Microsoft Corporation

Dan Miller[†] DANIEL.KEEGAN.MILLER@GMAIL.COM
Airbnb

Danny Karmon[†] DANNYKARMON@GOOGLE.COM
Google

Dimitrios Dimitriadis[†] DBDIM@AMAZON.COM
Amazon

Abstract

Federated Learning (FL) is a machine learning approach that allows the model trainer to access more data samples by training across multiple decentralized data sources while enforcing data access constraints. Such trained models can achieve significantly higher performance beyond what can be done when trained on a single data source. In a FL setting, none of the training data is ever transmitted to any central location; i.e. sensitive data remains local and private. These characteristics make FL perfectly suited for applications in healthcare, where a variety of compliance constraints restrict how data may be handled. Despite these apparent benefits in compliance and privacy, certain scenarios such as heterogeneity of the local data distributions pose significant challenges for FL. Such challenges are even more pronounced in the case of a multilingual setting. This paper presents a FL system for pre-training a large-scale multilingual model suitable for fine-tuning on downstream tasks such as medical entity tagging. Our work represents one of the first such production-scale systems, capable of training across multiple highly heterogeneous data providers, and achieving levels of accuracy that could not be otherwise achieved by using central training with public data only. We also show that the global model performance can be further improved by a local training step.

Data and Code Availability Most of the data used in this paper, apart from proprietary clinical notes and non-English annotated texts, is publicly available; review Table 1 for unannotated texts used for training and evaluating the language models and Table 4 for NER datasets. All texts were tokenized using NLTK (Bird et al., 2009) and custom scraping tools (wikidata query tools (Wikidata.org, 2022), Selenium (Software Freedom Conservancy, 2022), Beautiful Soup (Richardson, 2022), E-utilities (Sayers, 2018)) were developed to obtain the unannotated texts. Proofs of concept were built using the FLUTE library (Dimitriadis et al., 2022). For the experiments on the AzureML platform, FL pipelines were built using the open source Shrike library (Microsoft Corp., 2022c). The model code is proprietary and cannot be shared, but is similar in architecture to XLM-K (Jiang et al., 2022).

Institutional Review Board (IRB) This work did not require IRB approval.

1. Introduction

Federated learning is a machine learning paradigm for training models on decentralized data found in

[*] Authors contributed equally. [†]Work done while at Microsoft.

segregated sources (silos), as described in (McMahan et al.; Kairouz and et al, 2019). The models are trained locally for a number of steps, and then combined together on a central server, ultimately creating a global model that contains information from all data sources. A basic assumption for most of these FL methods is the data sources are independent and identically distributed (i.i.d.) or at least with "overlapping" distributions. As such, these algorithms are not designed to adequately handle multilingual data, that is non-i.i.d. by definition, with severely skewed local distributions. In such scenarios, the data in each silo is expected to be locally homogeneous and monolingual, but the overall global data distribution is non-i.i.d., making it challenging to aggregate all the local models. Consider, for example, the task of combining models independently trained on English and German corpora. Furthermore, the volume of available data per language can also be severely skewed – high-resource language data can be much more abundant, making the trained model heavily "focused" on the high-resource languages and failing to adequately model the low-resource ones.

"*Natural Language Understanding*" (NLU) refers to machine extraction of knowledge from unstructured human communications, mainly text-based sources. Although NLU tasks have been focused mostly on documents written in a single language, the joint analysis of multilingual documents is attracting increasing attention. It is shown that NLU models, when trained on multilingual datasets, can extract knowledge from such vastly different corpora, improving the overall model performance (Huang et al., 2020). In this context, the goal of multilingual NLU is to create a single model for all languages, exploiting any correlations and underlying relationships that span beyond the language barrier. Most existing work on multilingual NLU is focused on scenarios where the data is centrally stored. However, multilingual data found in real-life scenarios, especially in the space of healthcare, is most often distributed across multiple providers, e.g., as in (Wang et al., 2022). The participating data providers possess text data that is segregated and stored locally, and the NLU objective is to collectively process the documents without sharing any of the raw data. In this paper we present one of the first commercial FL applications training a single global model in a multi-lingual healthcare setting (supporting nine data-segregated languages).

Statistical NLU systems have been designed by extracting features from corpora using statistical and machine learning algorithms and they have gradually replaced traditional rule-based systems because of their superiority in generalization and robustness. In healthcare, NLU is most usually applied to process medical-related text, such as clinical notes and other related text data. Clinical notes come from all medical scenarios and mainly consist of unstructured text stored in "*electronic health record*" (EHR) systems, including medical notes, diagnostic reports, electronic prescriptions, *et cetera*. Other text data may derive from other healthcare scenarios, e.g., clinical trial protocols, medical publications, surveys in population screening and articles for evidence-based reference. Besides these notes, an emerging source of data is based on transcriptions of patient-doctor communications combined with machine translations and user-facing conversational bots (Hb et al., 2020). Research on applications of NLU for smart healthcare has received intensive attention in recent years, with some systems reaching maturity (in terms of productization) (Bhirud et al., 2019) among others.

As mentioned above, the promise of federated learning is a shared global model is trained under the coordination of a central server while keeping the user data segregated on local silos (aka clients). Federated Learning has been applied to problems in NLU since its inception (Liu et al., 2021; Lin et al., 2021), and in particular for language modeling tasks (Yang et al., 2018; Hard et al., 2018; Stremmel and Singh, 2020). However, the multilingual NLU setting appears far more challenging. In such a case, FL-based optimization suffers from training instability, slower convergence rates and lack of fairness for the smaller clients. On the other hand, since FL techniques can now provide theoretical warranties for convergence in the case of non-i.i.d. data distributions, as in (Li et al., 2020), the value of its application to multilingual tasks takes on greater interest, where privacy and legal constraints are also of concern. Most of the legal constraints are based on the data sovereignty principle – each data provider maintains ownership and control of their data and, as such, it's not possible for the data to be mixed. The multilingual setup in healthcare falls under this constraint where the data silos are held by different providers, and even located in different countries with differing regulatory regimes. Herein, we present a production-ready system able to overcome such geopolitical barriers, using techniques from FL optimization. Data in each silo is in a different language, and cannot be shared due to being sensitive, to the laws in different countries etc.

Concurrent with the growing interest in Federated Learning, NLU has rapidly shifted towards the use of "*foundation language models*" (fLMs), for foundation models see (Bommasani and et al, 2022) and extensions to fLMs in (Devlin et al., 2019a), GPT-3 (Brown et al., 2020). These fLMs are used as a starting point for learning other downstream NLU tasks. Such training strategies have become the golden standard for most of the concurrent applications. Lately, multilingual versions of these models have been also proposed and are often used along with few-shot and/or transfer learning techniques to increase performance for tasks where the available target-language training data is limited. This state-of-the-art setup exploits the strong few-shot learning capabilities of large transformer models and fLMs generally. The scenario presented in this paper is one occurrence of several FL approaches for fLMs. Other papers in this space focus on the performance gap between federated learning and centralized training, evaluating on a wide variety of English NLU tasks (Liu and Miller, 2020; Lin et al., 2021). On the contrary, we differ from such work by studying the federation of fLMs in a highly imbalanced and non-i.i.d. setup, with performance constraints across all participating languages.

In this work, we explore multilingual Federated Learning across 9 languages, each with various amounts of available training data, i.e. Table 1, while leveraging a pretrained foundation model as the initial seed model. Our results show that, by applying continued pre-training with FL, such models can perform similarly to centralized methods (that is, the case when no data accessibility constraints are in place), despite having completely non-i.i.d. data distributions among the participating silos (each with monolingual data). We show that training fLMs this way provides an effective and generalizable way for processing multilingual data all while benefiting from the accessibility features of FL at little or no cost to the final/downstream task performance. As part of the proposed solution, accessing and sampling the individual silos based on the available resources per language can ensure a more fair knowledge representation.

In addition to the sampling strategy, we investigate the merits of personalization in the overall model performance. We show that under-resourced languages can benefit either from simple fine-tuning or from interpolation between a locally fine-tuned model and the globally trained mode (Deng et al., 2020).

The contributions of the presented system are:

1. We present one of the first production-ready FL-based systems for medical-related NLU where the models are trained on real-life data.

2. An end-to-end FL system for training base multi-language models for medical text understanding, when the participating clients (in different languages) have different amounts of training data. Herein, we present different sampling strategies for the proposed system.

3. An investigation into when personalization in the form of model interpolation can benefit performance, especially for under-represented languages.

4. Experimental validation of the proposed approach: we present comprehensive experimental results supporting the proposed design and algorithmic decisions.

In summary, this work presents the federation of an NLU foundation model trained on multilingual data for medical text analytics as the down-stream task. The challenges for such systems mostly lie in the extreme non-iidness of the multilingual data, the skewed training data distributions in terms of volume and the data constraints driven by the geopolitical legal framework. The deployed system federates a single multilingual model without mixing data from different silos and/or locales, all while improving performance for all locales in tandem. The data sources and algorithms implemented in this system are detailed in Section 2, including the design decisions for client sampling, optimization, etc. The experimental findings are presented in Section 3. We focus mainly on the performance of the trained models on the downstream task, i.e. named-entity recognition (NER) in the healthcare setting, as well as the impact of local personalization, i.e. fine-tuning the global model on each language. Finally, Section 4, summarizes our findings and open challenges.

The framework we develop in this paper has been used in the real-world, to train production-oriented NER models in a federated way. More information can be found at (Bitran, 2022).

2. Methods

2.1. Task and Model Description

Pretrained transformer-based language models (Vaswani et al., 2017; Devlin et al., 2018) achieve state-of-the-art performance on a variety of NLU tasks

(Wang et al., 2018, 2019), especially when pretrained on domains similar to the target task (Gururangan et al., 2020). Such models also show great promise achieving zero-shot classification (Liang et al., 2020) when trained across multiple languages (Conneau et al., 2019; Jiang et al., 2022).

The proposed approach to achieve state-of-the-art performance on multilingual medical domain NLU tasks is to use FL to further pretrain an already pretrained model, i.e. an XLM-K (Jiang et al., 2022) model, on the medical domain, with a specific focus on multilingual clinical notes. At the time of writing, we have only been able to find such models in English (Beltagy et al., 2019; Lee et al., 2020; Gu et al., 2020). In fact, one of the key problems we faced when trying to build this model is the availability of public medical text in other languages, especially clinical notes. The training task we chose for continued pretraining is the masked language model (MLM) task (Devlin et al., 2019b), where some of the tokens in the input text are randomly masked and the model aims to accurately predict them based only on the surrounding context. As such, no additional labels are required for the training phase. The standard evaluation metric for MLM training is to measure the mean *perplexity* PP over the test set, where $PP(\cdot)$ is defined as

$$PP(W) = \sqrt[N]{\prod P(w_1, w_2, \dots, w_N)^{-1}}$$

where $W = w_1, w_2, \dots, w_N$ is the set of tokens in a training example and $P(\cdot)$ is the probability the model assigns to the sequence.

Once this new model is obtained, we can fine-tune the model to perform other specialized downstream tasks. Our experiments focus on the "*named entity recognition*" (NER) task, i.e., identifying whether there are words or expressions on the notes that belong to one of a few categories such as symptoms, diseases, medical codes, and so on.

2.2. Baseline Model

The central learning (CL) baseline is obtained based on the conventional optimization process for continued pre-training using the data collected from different sources, cf. Table 1. Hyperparameters were initally selected according to past experiments with similar models and datasets, as detailed in Table 2. These initial settings were further tuned for the FL case. The CL results shown in Table 3 and Figure 2 refer to this baseline.

Roughly 10.4% of the training data is used for central training, since the volume of all the available data is too large, and using all of it was deemed prohibitive. Batches are created by sampling uniformly at random from examples from all languages, and are not stratified per language. Evaluation is performed every 2048 batches of training, using 65,536 samples picked at random from the test set. These validation perplexities are reported in Figure 2. In contrast, the perplexities reported in Table 3 are computed over the full test set.

In parallel, we trained a single model per silo (without either FL or mixing data from other languages) with the corresponding results shown in Table 3, using the same hyperparameters and data volume. Next, we provide more details about the training of the federated model.

2.3. Federated Learning

Available medical data is not fully exploited by researchers and medical institutions given the constraints on transmitting sensitive information from private silos to a centralized location. Federated Learning provides a collaborative learning environment with privacy guarantees under the coordination of a central server. The proposed production-scale system can train models across hundreds of silos without sharing raw data, allowing partners across different cloud storage and compute providers to train a single model.

For performing the federated training, we use a hierarchical optimization approach (Dimitriadis et al., 2021), instantiating a persistent optimizer on the server, $f^s(\cdot)$; and intermittent stateless ones $f_i^c(\cdot)$, on each silo i, which are re-instantiated at each training round. This approach has been shown to improve the convergence rate, allowing better control of the learning process. Specifically, the global model at round r, $\theta^{(r)}$ is communicated to the participating silos i and trained on their data, with the the successive local gradients $g_i^{(r,b)}$, $b = 1, \dots, B$. The silo-side optimizers then update the local weights $\theta_i^{(r,b+1)} = f_i^c(\theta_i^{(r,b)}, g_i^{(b)})$, with $\theta_i^{(r,0)} \equiv \theta^{(r)}$. Once the maximum number of local batches or the end of local data is reached, the silo estimates a local "pseudo-gradient" $g_i^{(r)}$,

$$g_i^{(r)} \overset{\text{def}}{=} \theta^{(r)} - \theta_i^{(r,B)}, \tag{1}$$

and transmits it back to the server. Once all silos have transmitted their local gradient, the updated

Silo ID	Language	# Train samples	# Test samples	Sources

1	Italian	412,437	26,192	e3c(Magnini et al., 2020)
				EMA(Gøtzsche and Jørgensen, 2011)
				Wikipedia clinical articles(Wikipedia, 2004)

2	French	477,323	31,167	PMC(Roberts, 2001)
				QUAERO(Névéol et al., 2014)
				EMA, Wikipedia clinical articles

3	Spanish	502,479	42,225	SciELO(Packer, 2009), CODEIESP(Miranda-Escalada and Gonzalez-Agirre, 2020) e3c, EMA, PMC, Wikipedia clinical articles
4	Portuguese	400,175	26,192	PMC, EMA, Wikipedia clinical articles
5	English	132,593,658	4,400,378	ClinicalTrials.gov(Zarin et al., 2011) MIMIC-III(Johnson et al., 2016) MIMIC CXR(Johnson et al., 2019) i2b2(Uzuner et al., 2006), Wikipedia clinical articles, Proprietary clinical notes
6	German	575,559	26,192	OPUS(Tiedemann, 2012) PMC, Wikipedia clinical articles
7	Arabic	83,963	26,192	Wikipedia clinical articles
8	Hebrew	34,104	26,192	IMA(Association), MyTrials(Trial), Wikipedia clinical articles
9	Russian	100,762	26,192	PMC, Wikipedia clinical articles
		135,280,160	**4,630,922**	

Table 1: Public data used to validate the FL model. Data is severely unbalanced across silos: the one containing English data has many more samples.

CL/per-silo	FL
# samples seen Per batch: (up to) 2048	**# samples seen** Per round on silo i: $\max(500, 0.8 \cdot 10^{-4} N_i)$ Per batch: (up to) 2048

optimizer
`AdamW`
`lr` (γ_0) `= 3e-5`
`eps = 1e-6`
`weight_decay = 0.01`
	optimizer
	Client/Silo: `SGD`
	`lr = 1e-4`
	Server: `Adam`
	`lr` (γ_0^s) `= 3e-4`

scheduler none	**scheduler** At round r: $\gamma^s = (1 - 10^{-3} r)\, \gamma_0^s$

Table 2: Hyper-parameters used for pre-training.

global model is given by

$$\theta^{(r+1)} = f^s\big(\theta^{(r)}, \sum_i w_i g_i^{(r)}\big). \tag{2}$$

The weights w_i that can be set in different ways (Dimitriadis et al., 2021) – in most experiments, we have used $w_i = N_i / \sum_i N_i$, as in FedAvg (McMahan et al.).

A brief hyperparameter search was executed manually, varying batch size, samples seen per iteration (on the FL case), optimizer and scheduler parameters. The ones in Table 2 have been picked base on the best perplexity results during validation[1].

At each silo i, only a fixed number $\max(500, 0.8 \cdot 10^{-4} N_i)$ of samples is used per iteration, out of the total N_i samples available in that silo. In practice, this choice makes the sample size proportional to N_i for large silos, and fixed at 500 for smaller ones—for the data in Table 1, only the English silo will have the sample done proportionally, with $10.6k$ samples being processed in the form of 6 batches per round, while the remaining will have fixed sample sizes. These local training samples are picked uniformly at random, *with* replacement. This is a deviation from the sampling approach in the CL scenario, where samples are seen once per epoch.

Validation is performed every 5 FL rounds, with 10% of the total amount of test samples being picked at random. These validation results are shown in

[1] A side-comment here is that the final results are quite sensitive to the batch sizes.

Figure 2; the results on Table 3 are based on the final model and reported on the entire test set.

2.4. Personalization

The convergence of most Federated Learning optimization algorithms is theoretically proven when the client data distributions are iid. However, scenarios, such as the multilingual NLU ones, where the data distributions are non-iid, are far more challenging. One of the different approaches for addressing this issue is with convex interpolation between the global $\theta^{(r)}$ and locally fine-tuned models $\theta_i^{(r,B)}$, (Deng et al., 2020). The resulting model $\theta_{int}^{(r)}$ after interpolation is given by

$$\theta_{int}^{(r)} = \alpha_i \cdot \theta_i^{(r,B)} + (1 - \alpha_i) \cdot \theta^{(r)} \tag{3}$$

and the interpolation weights α_i for each client i are estimated as described in (Deng et al., 2020).

We have tried different strategies for training the local models depending on the initial checkpoint of the local model. We have found that starting the local model training later in the FL process allows for improved generalization of the models. Otherwise, the local models are quickly overfitting, degrading the overall performance. More details are discussed in Section 3.3.

2.5. Engineering System Architecture

The proposed production solution is based on Azure Arc-enabled Kubernetes clusters (Microsoft Corp., 2022b; Lin, 2022), enabling large-scale FL applications on the Cloud. The developed FL system offers templates for the target task and a public API allowing easy deployment of production FL tasks (Microsoft Corp., 2022d,a).

Following FL principles, the local data never leave the customer tenant, and are only processed on their own Arc-enabled Kubernetes compute cluster. Each silo iteratively trains a local version of the global model using its own data on its own compute environment. At each iteration, a silo might either use the silo's full data for the model training, or a randomly sampled subset (and different subsets across iterations) of the silo's data. Once these silos finish processing their data (or reach a maximum number of processed batches), the locally adapted model weights are transmitted back to the central orchestrator/server. On the server, the model weights from all silos are then aggregated appropriately and update the global model of the previous iteration, before moving to the next training iteration. The flowchart is shown in Figure 1.

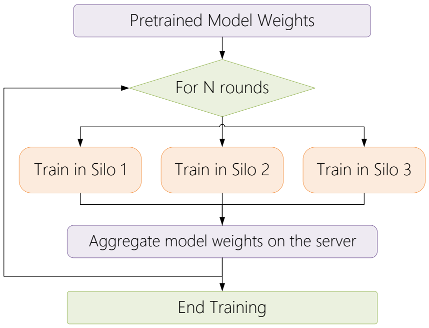

Figure 1: Cross-silo federated learning flowchart.

The FL task is set up according to the published recipes (Microsoft Corp., 2022a), with the required resources:

1. the central orchestrator in an AzureML workspace,
2. the compute environment for this central orchestrator,
3. the individual data silos with Azure Arc-enabled Kubernetes compute environments and
4. the Azure Data Factory compute for transferring intermediate model weights between silos.

Next, the user can create the FL pipeline by following the public API documentation, i.e., specifying global task configuration (specifying parameters like 'max_iterations', compute targets, etc.), and creating the appropriate component functions, as shown in Figure 1, e.g., `Train` for local model training at each silo, and `Aggregate` for the model weight aggregation at the server, etc. The available FL API also provides visualization tools for the AML workspace.

Finally, the FL API enables secure aggregation using cryptographically generated masks based on secure multi-party computation (MPC) (Canetti et al., 1996). This security feature ensures the server cannot link the communicated model weights to a particular client. The added masks are designed to cancel each other out during the aggregation step, and as such the aggregated model is unaffected.

3. Results

3.1. Validation on Public Data

We first use publicly available data to demonstrate the capacity for a multi-lingual fLM to pretrain in a federated setting with results competitive with central training. Starting with a base snapshot of XLM-K (Jiang et al., 2022), we federate the model across nine silos, each of which contains medical text in a distinct language as shown in Table 1.

As usual in FL, the pretraining procedure consisted of a sequence of iterations where, at each iteration, every silo produces a new model based on the latest global model, and sends it to the server, which then combines all models it received. Procedures for updating the models and for combining them are detailed in Section 2.3. The system infrastructure for the silos, server and the coordination, are described in Section 2.5.

Per-language model quality was assessed using the perplexity on test data, which for simplicity was made available to the server. The FL performance was compared to a baseline produced using central learning (CL), with the data from all silos pooled together; and also to language-specific baselines, trained using data from each individual silo. In addition, we report perplexity results for the zero-shot task (i.e. performance using the base model snapshot). Importantly, all models were trained using roughly 10% of the available, relevant training samples, to match the data volumes used for the FL and CL cases. The hyper-parameters for FL and CL were independently adjusted, and the per-silo models used the same parameters as CL.

The results in Table 3 show FL performance competitive with and sometimes outperforming central training. We also observe that English, German, and Spanish all benefit from training on all languages, rather than training per-language models. Figure 2 illustrates the convergence of the federated model compared to central training.

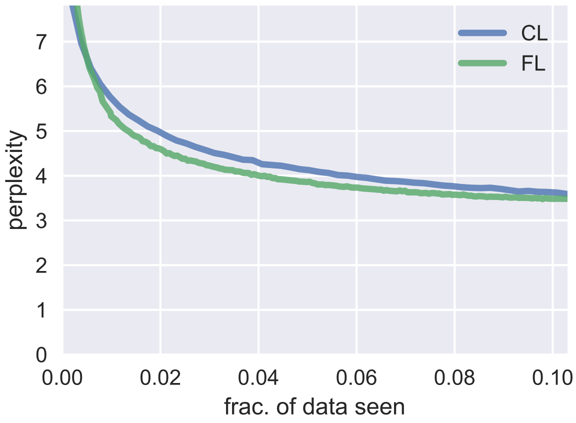

Figure 2: Convergence of FL vs CL in terms of the overall perplexity over the test set. After approximately 10% of the data has been seen, the two models report similar perplexities, of 3.49 (FL) and 3.59 (CL).

Finally, we used the FL and CL snapshots to evaluate performance on downstream tasks. Specifically, we have fine-tuned two models for performing named entity recognition (NER), each starting from one of the snapshots. This procedure was repeated for three different public datasets: BC5CDR(Li et al., 2016), NCBI-disease(Doğan et al., 2014), and i2b2_2009(Patrick and Li, 2010). As seen in Table 4, both give similar results in terms of achieved micro, macro and weighted f1-scores.

Together, Tables 3 and 4 and Figure 2 show that our federated learning setup can produce comparable, if not superior, models, without sharing raw data during the training process. Small differences in performance can be attributed to the batch stratification that happens in FL, as well as the choice of hyper-parameters, which were independently optimized for each model. This result has already been observed in multiple works across the literature, although most consist of simulations that cannot be easily generalized to real-world settings. Within our setup, this is just a matter of replacing the silos containing public data with real-world data silos, since they are already on isolated cloud environments. This is what we have done next.

3.2. Deployment and Validation on Real-World Data

The federated approach described in this paper was deployed in a real production environment to train a multi-lingual fLM with Hebrew support, considering the lack of publicly available clinical text in this language. Hebrew clinical notes were de-identified using HebSafeHarbor (8400 The Health Network, 2022) and used to tune XLM-K to the medical domain, using 2.5GB of proprietary Hebrew clinical notes in one silo, and English public clinical text in a separate silo.

Once pretraining has been performed using federated learning, the resulting model was attached to different classification heads to fine-tune on proprietary data, for solving 3 tasks: *”named entity recognition”* (NER), *”assertion detection”* (AD) and *”relation extraction”* (RE). This dataset consisted of annotated clinical notes in both English and Hebrew. Note that the de-identified clinical notes could not be accessed in a centralized training setting.

language	FL	CL	It	Fr	Es	Pt	En	De	Ar	He	Ru	Base
Italian	**5.63**	5.80	6.32	8.12	10.18	8.51	22.12	8.11	9.22	10.76	9.25	16.10
French	4.97	**4.94**	7.16	5.53	7.62	7.36	18.09	6.98	8.01	9.35	7.72	13.13
Spanish	**5.65**	5.81	9.01	7.96	6.45	7.74	17.93	8.11	9.17	11.40	9.11	17.28
Portuguese	**5.85**	6.01	9.36	8.91	9.40	6.91	22.68	8.98	10.23	12.46	10.31	18.65
English	**3.41**	3.51	15.60	14.65	13.84	14.57	3.50	13.96	15.88	20.71	15.54	32.83
German	7.10	**7.04**	10.38	10.07	10.24	10.47	28.12	7.74	11.57	14.60	11.42	21.98
Arabic	**8.36**	10.48	12.07	12.30	12.20	12.66	33.79	11.25	10.81	13.78	12.22	19.82
Hebrew	**6.98**	9.44	10.44	10.14	10.36	11.18	32.08	9.73	10.80	11.43	10.65	19.30
Russian	**6.07**	7.17	7.95	7.66	7.59	8.12	25.95	7.56	8.65	10.39	7.34	14.77

Table 3: Perplexity per-language obtained on the test data for different models (lower is better): the one pretrained with federated learning (FL); another pretrained with centralized learning (CL), where all data is pooled together; and others obtained by using only data from specific silos. Bold results show the best result overall, and results highlighted in red show the best result per individual training language.

Dataset		FL	CL
BC5CDR	micro	0.8913	**0.8917**
	macro	0.8917	**0.8920**
	weighted	0.8915	**0.8919**
NCBI-dis.		0.9287	**0.9370**
i2b2_2009	micro	**0.9184**	0.9174
	macro	**0.8476**	0.8446
	weighted	**0.9170**	0.9155

Table 4: Best f1-score obtained for NER on three different datasets, presenting micro, macro and weighted averages over entities. For NCBI-disease a single number is presented, since there is a single entity.

Task	CL (public)	FL
NER (strict/type)	0.5415/0.6778	**0.6628/0.8118**
Relation Extraction	0.8144	**0.8178**

Table 5: Comparison of f1-score obtained in downstream tasks with proprietary data, using either a language model trained on public clinical text using CL, or a language model trained on Hebrew/English silos using FL. Note that, in contrast to previous experiments, CL here uses only public data. NER scores were computed in two different ways, requiring either exact entity boundaries to match (*strict*) or only the entity types (*type*)(Segura-Bedmar et al., 2013).

The average f1-score obtained in 2 of these 3 tasks is reported in Table 5. The AD task has a highly unbalanced label set and we found the classification scores to be uninformative, so they are not reported here. Note that for the reported tasks the FL model provided considerably higher accuracy in all cases, when compared to the previous model which used public data only.

3.3. Personalization

Silos can leverage their local data to obtain models that are better adjusted to the local data distribution. In Table 6, we introduce a *personalized* model, which starts from the global FL model at a given iteration, and only uses local data from that iteration onward. As shown in the table, the accuracy provided by this personalized model in the task of predicting masked tokens is often better than that of the FL model, especially for non-English silos. Moreover, if we interpolate between these two models, following the procedure described in Section 2.4, even better results can be obtained. Interpolating with the global model may have a regularizing effect, defending the overfitting of the local model against the local data distribution (Deng et al., 2020).

The experiment above was run as a proof of concept using the FLUTE simulation platform (Dimitriadis et al., 2022), with the same public data as in previous experiments. After confirming that personalization and interpolation can indeed improve the performance of the global FL model, we ran additional experiments

	English	French	Spanish	German	Russian	Portuguese	Italian
Federated	**0.6976**	0.6404	0.6183	0.5950	0.5876	0.6145	0.6194
Personalized	0.6945	0.6491	**0.6286**	0.6007	**0.6054**	0.6213	**0.6288**
Interpolated	0.6970	**0.6510**	0.6279	**0.6024**	0.6043	**0.6220**	0.6279

Table 6: Test accuracy obtained in predicting masked tokens, using different models. Personalized, silo-specific models typically provided a better performance than the global FL one; moreover, interpolating these two models can improve the performance even further.

in the production platform on AML. Specifically, we trained a personalized model starting from the final FL model, using only German data; this, already, provided better performance when evaluating on the German test set. Figure 3 shows that while the personalized model already has better performance than the global FL model on the German test data, interpolating can provide even better results, in this case with an α value of 0.9.

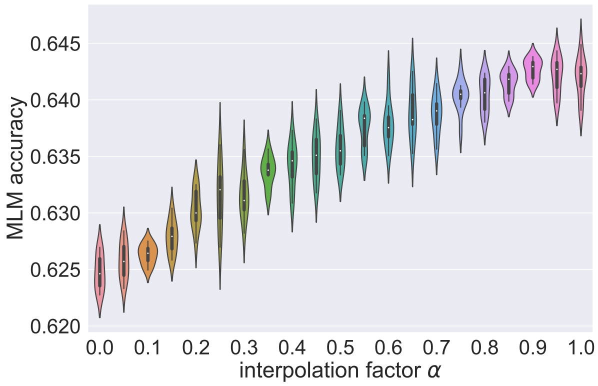

Figure 3: Test accuracy obtained on the MLM task interpolating between the global FL model ($\alpha = 0$) and a personalized model ($\alpha = 1$) trained post-hoc, starting from the global FL model and using German data. For each of 21 values of α partitioning the range $[0, 1]$, an evaluation over the whole German test set was performed.

4. Discussion

The current proposed approach navigates through two different sets of constraints and considerations: one set can be attributed to the legal and privacy framework and the Cloud business reality, and the second one attributed to algorithmic and task-related challenges.

4.1. Policy and Business Constraints

Policy Challenges Today many parts of the world require technology companies to treat user data, which is usually generated and stored in data silos (e.g., service providers' data centers), according to user-privacy laws. Examples of such laws include the "*European Union General Data Protection Regulation*" (GDPR)(European Union, 2016), the "*California Consumer Privacy Act*" (CCPA) (California Attorney General, 2022), and the "*Health Insurance Portability and Accountability Act*" (HIPAA) (Centers for Disease Control and Prevention, 2022). Such legal constraints pose a challenge to traditional centralized ML approaches, where all data is usually stored at a single, central location, typically a Cloud data center. Centralized data management and processing can impact transparency and data provenance tracking, which in turn could lead to the lack of trust from the end-users, as well as increased difficulty in compliance with the data governance laws like the GDPR and/or HIPAA. As a response to these challenges, Federated Learning is an evolving technology that is well positioned to address such policy challenges appropriately.

Federated Learning on Cloud ML Platforms Besides these legal constraints, more practical issues with centralized training are the need for specialized computing resources, the fact that large-scale data collection and processing held on a single server can be seen as a single point-of-failure and a non-negligible risk of data breaches. Such hardware requirements drive increased costs from the service provider for training and maintaining the proper infrastructure.

Regardless, a number of incumbent cloud-based ML training platforms has emerged, including Azure ML. These platforms, while highly capable for processing very large data volumes and machine learning tasks, are characterized by their dependence on centralized data for training. On the other hand, a few platforms (e.g., Flower (Adap GmbH, 2022) sherpa.ai (sherpa.ai, 2022), OpenMined (OpenMined, 2022), and Substra (Galtier and Marini, 2019)) have been developed to enable data collaboration across security boundaries. However, these solutions focus more on smaller-scale user (data scientist) interaction solutions without built-in integration with cloud providers. There are a few large scale federated learning platforms, some proprietary e.g., from Google, Apple and Meta, for federated learning using consumer, mobile, and advertising data, rather than cloud-based enterprise data. Our focus has been on building and deploying large-scale cross-silo federated learning in the Cloud, processing enterprise data across dozens or hundreds of data providers. Our implementation, based on bespoke public APIs for cross-silo job orchestration in AML (Microsoft Corp., 2022b), is one of the first to address how existing ML platforms can be repurposed for cross-silo federated learning at scale, while addressing the complexities of managing compute and authentication across distinct organizations.

4.2. Algorithmic Challenges

Cross-Silo FL with Unbalanced Data Our experiments show that FL-based model training, with no or very limited impact in performance (when compared with the centralized training)[2] is possible even in settings where the local datasets are highly heterogeneous. While similar tasks have been extensively explored in the literature, the current work presents some innovative design decisions, e.g., the use of multilingual text data combined with a multilingual fLM in a healthcare context, a sampling strategy for the clients, etc. The proposed deployment is supported by the Azure ML platform as a straightforward extension. As detailed in Section 4.3, some algorithmic challenges concerning the stability and robustness of the presented system are still pending. For example, we have noticed that the amount of data used at each silo on any given iteration can widely affect the final result, due to model under- or over-fitting across the different languages. Handling such instability requires additional innovation as part of the FL optimization.

Personalization Finally, we investigate how personalization can improve performance in the presence of extreme statistical heterogeneity, as such in the case of multilingual NLU model training. In addition to the statistical heterogeneity of the local data distributions, the investigated FL setup presents additional heterogeneity due to the training dataset imbalance, e.g., the number of English training examples is at least 2 orders of magnitude more than the rest of the multilingual data. As such, this imbalance of training examples skews the global models significantly, as shown in Table 6. Tackling these heterogeneity challenges, a straightforward solution is to maintain multiple models for the different local distributions, e.g., as proposed in “Clustered FL” (Sattler et al., 2020). Another strategy is for FL-based personalization (Tan et al., 2022), where a different personalized model is generated for each client by leveraging both global and local information. Herein, we capture the differences in the local distributions by interpolating two models, the global and a local one (one local model per language), as described in (Deng et al., 2020).

It is shown that standard FL algorithms fail to ensure fairness for most of the under-represented languages, as shown in Table 6. On the other hand, the local or the interpolated model are nearly always better than the global one. In other words, the global model alone cannot fairly model data distributions when they are different, and a certain degree of personalization is necessary. However, a global model enhanced with additional information from other languages can be better when the training examples are overwhelmingly abundant for a particular language. It is shown, for example, that neither the personalized nor the interpolated model can improve over the global model in the case of English, as in Table 6.

It is an open question why in some cases the interpolated model out-performed the personalized version. It maybe the case that our scheme for choosing the interpolation parameter is sub-optimal, or perhaps in some cases the personalized model is over-trained and interpolation provides some regularization with the more general model.

Finally, our personalization approach assumes a simple handoff from global model training to continued training with local data. One can imagine other schemes that leverage data from “nearby” silos, such as employing Spanish to support a Portuguese model, as the system refines the model towards its target language. This is a interesting avenue for future work.

4.3. Conclusion and Future Work

Federated learning should be considered an indispensable tool in supporting privacy-sensitive data applications, where the training data is distributed and direct access is severely constrained. The Healthcare set-

[2] Centralized training can provide an upper bound in performance but it’s infeasible in real-life scenarios where data accessing constraints are in place.

ting and scenarios where personalization is needed are among the most prominent FL applications. HIPAA and other legal frameworks can severely constrain the sharing of private information. and as such, healthcare is one of those industries that can benefit the most from FL.

Although our work addresses some of the technical challenges discussed for a production deployment, there is a number of open questions that still need addressing. Among these questions are the computing and bandwidth resources for local training and communication. Further, some participating nodes may be significantly slower, i.e. these nodes are called "stragglers", and solutions based on asynchronous updates are required. The extreme heterogeneity present in the participating clients in the form of data distributions, available training examples have severe impact on the model fairness. Finally, data-related challenges such the lack of good-quality labels can affect the overall performance. Concluding, the number of practical concerns that arises, especially related to quality control and smooth operation, requires additional innovation– despite the fact that the presented system is production-ready. As part of our future work, our plan is robustify our solution in all of the above technical challenges.

References

8400 The Health Network. HebSafeHarbor, 2022. URL https://github.com/8400TheHealthNetwork/HebSafeHarbor.

Adap GmbH. Flower a friendly federated learning framework, 2022. URL https://flower.dev/.

Israeli Medical Association. Israeli medical association. https://www.ima.org.il/.

Iz Beltagy, Kyle Lo, and Arman Cohan. Scibert: A pretrained language model for scientific text. *arXiv preprint arXiv:1903.10676*, 2019.

Nivedita Bhirud, Subhash Tatale, Sayali Randive, and Shubham Nahar. A literature review on chatbots in healthcare domain. *Int. J. of Scientific & Technology Research*, 8, July 2019.

Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit.* " O'Reilly Media, Inc.", 2009.

Hadas Bitran. Expanding ai technology for unstructured biomedical text beyond english, 2022. URL https://azure.microsoft.com/en-us/blog/expanding-ai-technology-for-unstructured-text-beyond-english/.

Rishi Bommasani and et al. On the opportunities and risks of foundation models, 2022.

Tom B Brown, Benjamin Mann, Nick Ryder, and Melanie Subbiah et al. Language models are few-shot learners, 2020.

California Attorney General. California Consumer Privacy Act, 2022. URL https://www.oag.ca.gov/privacy/ccpa.

Ran Canetti, Uri Feige, Oded Goldreich, and Moni Naor. Adaptively secure multi-party computation. In *In proc. of Symp. on Theory of Computing (STOC '96)*, 1996.

Centers for Disease Control and Prevention. Health Insurance Portability and Accountability Act of 1996, 2022. URL https://www.cdc.gov/phlp/publications/topic/hipaa.html.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1912.02116*, 2019.

Yuyang Deng, Mohammad Mahdi Kamani, and Mehrdad Mahdavi. Adaptive personalized federated learning, 2020.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of NAACL-HLT'19*. ACM, July 2019a.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019b.

Dimitrios Dimitriadis, Kenichi Kumatani, Robert Gmyr, Yashesh Gaur, and Sefik Emre Eskimez. Dynamic gradient aggregation for federated domain adaptation, 2021.

Dimitrios Dimitriadis, Mirian Hipolito Garcia, Daniel Madrigal Diaz, Andre Manoel, and Robert Sim. FLUTE: A scalable, extensible framework for high-performance federated learning simulations. *arXiv preprint arXiv:2203.13789*, 2022. URL https://www.github.com/microsoft/msrflute.

Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. Ncbi disease corpus: a resource for disease name recognition and concept normalization. *Journal of biomedical informatics*, 47:1–10, 2014.

European Union. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation), 2016. URL https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A02016R0679-20160504.

Mathieu N Galtier and Camille Marini. Substra: a framework for privacy-preserving, traceable and collaborative machine learning. *arXiv preprint arXiv:1910.11567*, 2019.

Peter C Gøtzsche and Anders W Jørgensen. Opening up data at the european medicines agency. *Bmj*, 342, 2011.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical natural language processing, 2020.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. Don't stop pretraining: adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*, 2020.

Andrew Hard, Kanishka Rao, Rajiv Mathews, Swaroop Ramaswamy, Françoise Beaufays, Sean Augenstein, Hubert Eichner, Chloé Kiddon, and Daniel Ramage. Federated learning for mobile keyboard prediction, 2018.

Barathi Ganesh Hb, U. Reshma, Soman Kp, and M. Kumar. *MedNLU: Natural Language Understander for Medical Texts*, pages 3–21. 01 2020. ISBN 978-3-030-33965-4. doi: 10.1007/978-3-030-33966-1_1.

Zhiqi Huang, Fenglin Liu, and Yuexian Zou. Federated learning for spoken language understanding. In *Proc. of ACL'20*, pages 3467–3478. ACL, December 2020.

Xiaoze Jiang, Yaobo Liang, Weizhu Chen, and Nan Duan. XLM-K: Improving cross-lingual language model pre-training with multilingual knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10840–10848, 2022.

AEWP Johnson, Tom Pollard, Roger Mark, Seth Berkowitz, and Steven Horng. Mimic-cxr database. *PhysioNet10*, 13026:C2JT1Q, 2019.

Alistair EW Johnson, Tom J Pollard, Lu Shen, Liwei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3 (1):1–9, 2016.

Peter Kairouz and et al. Advances and open problems in federated learning, 2019.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, 2020.

Jiao Li, Yueping Sun, Robin J. Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J. Mattingly, Thomas C. Wiegers, and Zhiyong Lu. Biocreative V CDR task corpus: a resource for chemical disease relation extraction. *Database J. Biol. Databases Curation*, 2016, 2016. doi: 10.1093/database/baw068. URL https://doi.org/10.1093/database/baw068.

Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. In *8th Inter. Conf. on Learning Representations, ICLR 2020*. OpenReview.net, 2020.

Yaobo Liang, Nan Duan, Yeyun Gong, Ning Wu, Fenfei Guo, Weizhen Qi, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, et al. Xglue: A new benchmark dataset for cross-lingual pre-training, understanding and generation. *arXiv preprint arXiv:2004.01401*, 2020.

Bill Yuchen Lin, Chaoyang He, ZiHang Zeng, Hulin Wang, Yufen Huang, M. Soltanolkotabi, Xiang Ren, and S. Avestimehr. Fednlp: A research platform for federated learning in natural language processing, 2021.

Bozhong Lin. Realizing machine learning anywhere with azure kubernetes service and arc-enabled machine learning, 2022. URL https://techcommunity.microsoft.com/t5/ai-machine-learning-blog/realizing-machine-learning-anywhere-with-azure-kubernetes/ba-p/3475298.

Dianbo Liu and Tim Miller. Federated pretraining and fine-tuning of bert using clinical notes from multiple silos, 2020.

Ming Liu, Stella Ho, Mengqi Wang, Longxiang Gao, Yuan Jin, and He Zhang. Federated learning meets natural language processing: A survey, 2021.

Bernardo Magnini, Begoña Altuna, Alberto Lavelli, Manuela Speranza, and Roberto Zanoli. The e3c project: Collection and annotation of a multilingual corpus of clinical cases. In *CLiC-it*, 2020.

H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data.

Microsoft Corp. Federated Learning in Azure ML, 2022a. URL https://github.com/Azure-Samples/azure-ml-federated-learning.

Microsoft Corp. Configure Kubernetes cluster for Azure Machine Learning, 2022b. URL https://github.com/Azure/AML-Kubernetes.

Microsoft Corp. Shrike: incubation for Azure ML, 2022c. URL https://github.com/azure/shrike.

Microsoft Corp. How to write a federated learning pipeline using shrike.pipeline.federated_learning, 2022d. URL https://shrike-docs.com/pipeline/federated-learning-doc/.

Antonio Miranda-Escalada and Aitor Gonzalez-Agirre. Codiesp: Clinical case coding in spanish shared task (ehealth clef 2020). In *eHealth CLEF 2020*, 2020.

Aurélie Névéol, Cyril Grouin, Jeremy Leixa, Sophie Rosset, and Pierre Zweigenbaum. The quaero french medical corpus: A ressource for medical entity recognition and normalization. In *In proc biotextm, reykjavik*. Citeseer, 2014.

OpenMined. A world where every good question is answered, 2022. URL https://www.openmined.org/.

Abel L Packer. The scielo open access: a gold way from the south. *Canadian Journal of Higher Education*, 39(3):111–126, 2009.

Jon Patrick and Min Li. High accuracy information extraction of medication information from clinical notes: 2009 i2b2 medication extraction challenge. *Journal of the American Medical Informatics Association*, 17(5):524–527, 2010.

Leonard Richardson. Beautiful Soup Documentation, 2022. URL https://www.crummy.com/software/BeautifulSoup/bs4/doc/.

Richard J Roberts. Pubmed central: The genbank of the published literature, 2001.

Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE Trans. on Neural Networks and Learning Systems*, pages 1–13, 08 2020.

Eric Sayers. E-utilities Quick Start, 2018. URL https://www.ncbi.nlm.nih.gov/books/NBK25500/.

Isabel Segura-Bedmar, Paloma Martínez Fernández, and María Herrero Zazo. Semeval-2013 task 9: Extraction of drug-drug interactions from biomedical texts (ddiextraction 2013). Association for Computational Linguistics, 2013.

sherpa.ai. Privacy-preserving artificial intelligence to advance humanity, 2022. URL https://www.sherpa.ai/.

Software Freedom Conservancy. Selenium, 2022. URL https://www.selenium.dev/.

Joel Stremmel and Arjun Singh. Pretraining federated text models for next word prediction, 2020.

Alysa Ziying Tan, Han Yu, Lizhen Cui, and Qiang Yang. Towards personalized federated learning. *IEEE Trans. on Neural Networks and Learning Systems*, pages 1–17, 03 2022.

Jörg Tiedemann. Parallel data, tools and interfaces in opus. In *Lrec*, volume 2012, pages 2214–2218, 2012.

Israeli Clinical Trial. Israeli clinical trial. https://www.gov.il/he/departments/general/clinical-trials-website.

Ozlem Uzuner, Peter Szolovits, and Isaac Kohane. i2b2 workshop on natural language processing challenges for clinical records. In *Proceedings of the Fall Symposium of the American Medical Informatics Association*. Citeseer, 2006.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer

Levy, and Samuel Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32, 2019.

Haoyu Wang, Handong Zhao, Yaqing Wang, Tong Yu, Jiuxiang Gu, and Jing Gao. FedKC: Federated knowledge composition for multilingual natural language understanding. In *Proc. of WWW'22*, page 1839–1850. ACM, April 2022.

Wikidata.org. Wikidata Query Service, 2022. URL https://query.wikidata.org/.

Wikipedia. *Wikipedia*. PediaPress, 2004.

Timothy Yang, Galen Andrew, Hubert Eichner, Haicheng Sun, Wei Li, Nicholas Kong, Daniel Ramage, , and Françoise Beaufays. Applied federated learning: Improving google keyboard query suggestions, 2018.

Deborah A Zarin, Tony Tse, Rebecca J Williams, Robert M Califf, and Nicholas C Ide. The clinicaltrials. gov results database—update and key issues. *New England Journal of Medicine*, 364(9):852–860, 2011.

Appendix A. Supplementary Material

A.1. Perplexity on Public Data – Split Dataset

For completeness, we recomputed the perplexities of FL and CL in Table 3 splitting the test data in 4 subsets, so that we could have a measure of uncertainty. Below, we provide the mean and standard deviation obtained in this new calculation. Note that means might differ from perplexities reported in Table 3, since perplexities here are computed over a smaller number of samples.

language	FL	CL
Italian	**5.61 ± 0.24**	5.98 ± 0.14
French	**5.15 ± 0.10**	5.32 ± 0.05
Spanish	**5.76 ± 0.16**	6.03 ± 0.20
Portuguese	**6.00 ± 0.07**	6.16 ± 0.07
English	**3.42 ± 0.01**	3.51 ± 0.02
German	7.31 ± 0.09	**7.22 ± 0.04**
Arabic	**8.57 ± 0.17**	10.60 ± 0.10
Hebrew	**6.88 ± 0.12**	9.32 ± 0.10
Russian	**6.21 ± 0.09**	7.05 ± 0.20

Table 7: Results similar to that of Table 3, in particular the two first columns. Here, the test data has been split in 4 subsets, and perplexity is computed for each of them; we report the mean and standard deviation of those 4 perplexities.