

**گزارش مقاله**

Communication-Efficient Learning of Deep Networks from Decentralized Data



تاریخ: ۱۴/۱۲/۱۴۰۳

Abstract

Mobile Devices مدرن به **حجم زیادی از داده‌ها** دسترسی دارند که برای یادگیری مدل‌ها مناسب بوده و می‌توانند تجربه کاربر را بهبود بخشند. برای مثال، مدل‌های زبانی می‌توانند تشخیص گفتار و ورود متن را ارتقا دهند و مدل‌های تصویری می‌توانند به‌صورت خودکار عکس‌های خوب را انتخاب کنند.

با این حال، این داده‌های غنی اغلب حساس از نظر **حریم خصوصی**، **دارای حجم زیاد**، یا **هر دو** هستند، که ممکن است انتقال آن‌ها به مرکز داده و آموزش با استفاده از روش‌های متداول را غیرممکن سازد.

ما یک جایگزین را پیشنهاد می‌کنیم که در آن داده‌های آموزشی روی دستگاه‌های موبایل باقی می‌مانند و یک مدل مشترک با تجمیع به‌روزرسانی‌های محلی محاسبه‌شده یاد گرفته می‌شود.

**این آزمایش‌ها نشان می‌دهند که این روش نسبت به Unbalanced Data (داده نامتعادل) و هم‌توزیع (non-IID) که ویژگی اصلی این محیط هستند، مقاوم است.**

**هزینه‌های ارتباطی محدودیت اصلی محسوب می‌شوند**، و ما نشان می‌دهیم که می‌توان تعداد دورهای ارتباطی موردنیاز را در مقایسه با گرادیان نزولی تصادفی همگام تا ۱۰ تا ۱۰۰ برابر کاهش داد.

Introduction

ما یک تکنیک یادگیری را بررسی می‌کنیم که به کاربران اجازه می‌دهد بدون نیاز به ذخیره‌سازی مرکزی، از مزایای مدل‌های مشترک که بر اساس این داده‌های غنی آموزش دیده‌اند، بهره‌مند شوند. ما این رویکرد را Federated Learning می‌نامیم.

علت این نام‌گذاری این است که وظیفه یادگیری توسط یک فدراسیون آزاد از دستگاه‌های شرکت‌کننده (که از آن‌ها با عنوان **Client** یاد می‌کنیم – فارسی آن مشتری است-) حل می‌شود و یک **سرور مرکزی** هماهنگی را بر عهده دارد.

**هر Client دارای یک مجموعه داده آموزشی محلی است که هرگز به سرور ارسال نمی‌شود**. در عوض، هر Client یک **به‌روزرسانی** برای مدل جهانی فعلی که توسط سرور نگهداری می‌شود، محاسبه کرده و تنها این به‌روزرسانی منتقل می‌شود.

یکی از مزایای اصلی این رویکرد، **جدا کردن فرآیند آموزش مدل از نیاز به دسترسی مستقیم به داده‌های خام آموزشی** است. البته همچنان نیاز به اعتماد به سرور هماهنگ‌کننده وجود دارد.

اما برای کاربردهایی که هدف آموزش را می‌توان بر اساس داده‌های موجود در هر مشتری مشخص کرد، Federated Learning می‌تواند **ریسک‌های حریم خصوصی و امنیتی را به میزان قابل توجهی کاهش دهد زیرا سطح حمله تنها به دستگاه محدود می‌شود نه همزمان به دستگاه و فضای ابری.**

به‌طور مشخص، ما الگوریتم **FederatedAveraging** را معرفی می‌کنیم که ترکیبی از گرادیان نزولی تصادفی محلی (SGD) روی هر مشتری و یک سرور است که میانگین‌گیری مدل را انجام می‌دهد.

**Federated Learning**

**مسئله ایده‌آل برای یادگیری فدرال دارای ویژگی‌های زیر است:**

1. آموزش بر روی داده‌های واقعی از دستگاه‌های موبایل نسبت به آموزش بر روی داده‌های جایگزین که معمولاً در مرکز داده موجود هستند، مزیت قابل‌توجهی دارد..
2. این داده‌ها از نظر حریم خصوصی حساس یا دارای حجم زیادی هستند (نسبت به اندازه مدل)، بنابراین ترجیح داده می‌شود که صرفاً برای آموزش مدل به مرکز داده ارسال نشوند (در راستای اصل گردآوری متمرکز و حداقلی داده‌ها)
3. در وظایف یادگیری نظارت‌شده، برچسب‌های داده را می‌توان به‌طور طبیعی از تعاملات کاربر استنباط کرد.

**برای مثال برای دو Task دسته‌بندی تصاویر (مانند پیش‌بینی اینکه کدام عکس‌ها بیشتر دیده یا به اشتراک گذاشته خواهند شد) و مدل‌های زبانی (تشخیص گفتار و ...) برچسب‌گذاری داده‌ها به صورت زیر است:**

**متن واردشده توسط کاربر به‌طور خودکار برچسب‌گذاری شده و برای آموزش یک مدل زبانی مناسب است**، و **برچسب‌های عکس را می‌توان بر اساس تعامل طبیعی کاربر با برنامه گالری تعریف کرد**. (مانند عکس‌هایی که حذف، به اشتراک گذاشته یا مشاهده می‌شوند)

**Privacy**

نگهداری حتی یک مجموعه داده «ناشناس» نیز می‌تواند حریم خصوصی کاربران را به خطر بیندازد، زیرا ممکن است از طریق ترکیب با سایر داده‌ها قابل شناسایی باشد. در مقابل، اطلاعاتی که در یادگیری فدرال منتقل می‌شوند، حداقل به‌روزرسانی لازم برای بهبود یک مدل خاص هستند (البته میزان مزیت حریم خصوصی به محتوای این به‌روزرسانی‌ها بستگی دارد)

به‌عنوان مثال، اگر به‌روزرسانی شامل **گرادیان کلی تابع هزینه** بر روی تمام داده‌های محلی باشد و ویژگی‌ها به‌صورت **کیسه‌ای از واژه‌های پراکنده** (sparse bag-of-words) نمایش داده شوند، در این صورت **گرادیان‌های غیرصفر دقیقاً نشان می‌دهند که کاربر چه کلماتی را در دستگاه خود وارد کرده است. (احتمال رخداد یک کلمه برابر ۱ است پس آن کلمه به کار رفته است)**

در مقابل، مجموع گرادیان‌های بسیاری در یک مدل متراکم مانند **شبکه عصبی پیچشی (CNN)** هدف دشوارتری برای مهاجمانی است که به‌دنبال استخراج اطلاعات درباره نمونه‌های آموزشی منفرد هستند، هرچند حملات همچنان امکان‌پذیر هستند.

(هر چند خود McMahan میاد و یه مقاله دیگه با تمرکز بر Privacy توی همین حوزه میده که باید مطالعه شود)

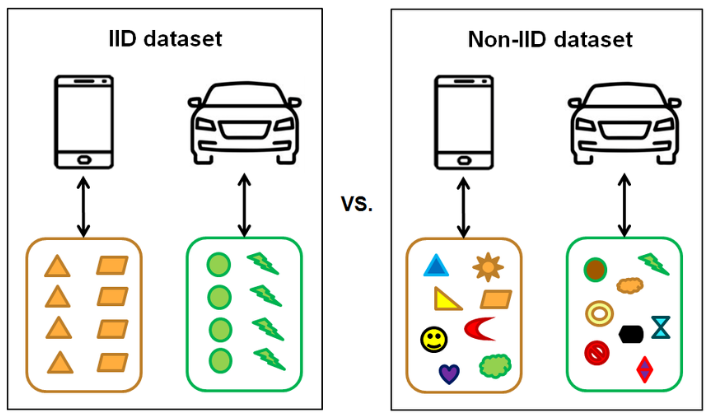
**این به‌روزرسانی‌ها می‌توانند (و باید) زودگذر باشند.** آن‌ها هرگز حاوی اطلاعاتی بیش از داده‌های آموزشی خام نخواهند بود (طبق اصل نابرابری پردازش داده‌ها) و معمولاً شامل اطلاعات بسیار کمتری هستند.

علاوه بر این، **منبع به‌روزرسانی‌ها برای الگوریتم تجمیع ضروری نیست.**

**Federated Optimization**

**بهینه‌سازی فدرال دارای چند ویژگی کلیدی است که شامل:**

* **داده‌های غیرمستقل و توزیع شده (Non-IID) (Non-Independent and Identically Distributed)**
* داده‌های آموزشی روی هر مشتری معمولاً بر اساس نحوه استفاده یک کاربر خاص از دستگاه موبایل است، بنابراین مجموعه داده محلی هر کاربر معمولاً نماینده توزیع کلی جمعیت نیست.



* **(Independent): مستقل هستند یعنی مقادیر داده‌ها به یکدیگر وابسته نیستند.**
* **(Identically Distributed): هم‌توزیع‌شده هستند یعنی از یک توزیع آماری یکسان نمونه‌برداری شده‌اند.**

**اما در یادگیری فدرال داده‌های اغلب Clientها Non-IID هستند زیرا** مجموعه داده‌های کاربران مختلف ممکن است وابستگی‌هایی داشته باشند و داده‌های هر کاربر از یک توزیع خاص خودش گرفته شده و با داده‌های کاربران دیگر متفاوت است.

* **نامتعادل بودن داده‌ها (Unbalanced Data)**
* برخی از کاربران از یک سرویس یا برنامه بسیار بیشتر از دیگران استفاده می‌کنند، که منجر به **مقادیر متفاوتی از داده‌های آموزشی محلی** بین کاربران می‌شود.
* **توزیع گسترده (Massively Distributed)**
* معمولاً تعداد زیادی از مشتریان در بهینه‌سازی شرکت می‌کنند، که تعداد آن‌ها **بسیار بیشتر از میانگین تعداد نمونه‌های آموزشی موجود در هر مشتری** است.
* **محدودیت ارتباطی (Limited Communication)**
* دستگاه‌های موبایل اغلب به‌صورت آفلاین هستند یا از اتصال‌های کند و پرهزینه استفاده می‌کنند، که باعث می‌شود ارسال داده بین مشتریان و سرور محدود باشد.

**تمرکز اصلی این پژوهش چیست؟**

**در این پژوهش تاکید بر Non-IID و Unbalanced داده‌ها و نیز Limited Communication هست.**

**در یک سیستم بهینه‌سازی یادگیری فدرال که به صورت عملی پیاده‌سازی شده است چالش‌های دیگری نیز دارد:**

1. تغییر مجموعه داده‌های مشتریان به دلیل **افزودن یا حذف داده‌ها**
2. در دسترس بودن مشتریان که ممکن است **با توزیع محلی داده‌ها به‌صورت پیچیده‌ای همبستگی داشته باشد**.

* (مثلاً کاربران انگلیسی‌زبان آمریکایی احتمالاً در زمان‌های متفاوتی نسبت به کاربران انگلیسی‌زبان بریتانیایی گوشی خود را به شارژ متصل می‌کنند)

1. مشتریانی که هرگز پاسخ نمی‌دهند یا به‌روزرسانی‌های مخدوش و نامعتبر ارسال می‌کنند.

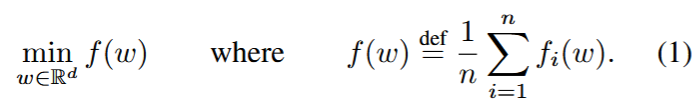
سه مورد گفته شده چالش اصلی این مقاله نیست ولی در عوض سه نکته گفته شده در بالا همچنان چالش است.

**Synchronous Update Schema**

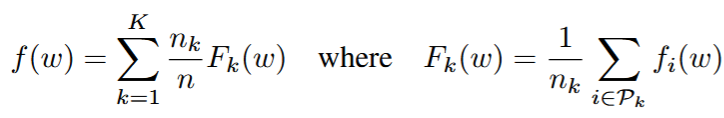
یک طرح به‌روزرسانی همگام (Synchronous Update Schema) که در دوره‌های ارتباطی (Communication Rounds) اجرا می‌شود.

* **تعداد ثابتی از K مشتری** در سیستم حضور دارند، که هرکدام **مجموعه داده محلی ثابتی** دارند.
* در **آغاز هر دور**، یک **نسبت تصادفی C از مشتریان انتخاب می‌شود**. سپس سرور **وضعیت فعلی مدل جهانی** (مانند پارامترهای مدل) را برای این مشتریان ارسال می‌کند.
* **فقط بخشی از مشتریان انتخاب می‌شوند** تا کارایی افزایش یابد، زیرا آزمایش‌های ما نشان می‌دهند که **افزودن تعداد بیشتری مشتری از یک حد مشخص، بازده کاهشی دارد.**
* **هر مشتری انتخاب‌شده، محاسبات محلی خود را انجام داده** و به‌روزرسانی‌های مدل را به سرور ارسال می‌کند.
* **سرور این به‌روزرسانی‌ها را روی مدل جهانی اعمال کرده و فرآیند تکرار می‌شود.**



****



****



****



**هزینه‌های ارتباطی در مقابل هزینه‌های محاسباتی**

* در **بهینه‌سازی مراکز داده**، هزینه‌های ارتباطی نسبتاً **کم** هستند و هزینه‌های **محاسباتی** بر فرآیند غالب‌اند. در این محیط، **استفاده از GPU** برای کاهش هزینه‌های محاسباتی اهمیت زیادی دارد.
* اما در **بهینه‌سازی فدرال**، **هزینه‌های ارتباطی غالب هستند**. معمولاً محدودیت **پهنای باند آپلود به کمتر از ۱ مگابایت بر ثانیه** باعث می‌شود که ارتباط میان مشتریان و سرور **بزرگ‌ترین مانع** برای آموزش مدل باشد.

همچنین، مشتریان فقط در شرایط خاصی **داوطلب شرکت در فرآیند آموزش می‌شوند**، مانند زمانی که:

* دستگاه آن‌ها به **شارژ متصل است.**
* به یک **شبکه Wi-Fi نامحدود** متصل هستند.
* مصرف باتری آن‌ها پایین است.

**علاوه بر این، هر مشتری معمولاً تنها در تعداد محدودی از دورهای به‌روزرسانی در روز شرکت می‌کند.**

**هدف: کاهش تعداد دوره‌های ارتباطی**

از آنجایی که **مجموعه داده محلی هر مشتری کوچک‌تر از کل مجموعه داده است** و گوشی‌های هوشمند مدرن دارای پردازنده‌های سریع و حتی شامل GPU هستند، هزینه محاسباتی **در مقایسه با هزینه ارتباطی تقریباً ناچیز** است.

هدف ما این است که با **افزایش مقدار محاسبات روی هر مشتری**، **تعداد دورهای ارتباطی موردنیاز برای آموزش مدل را کاهش دهیم.**

**دو راهکار برای افزایش محاسبات**

1. **افزایش موازی‌سازی (Parallelism):**

* استفاده از **تعداد بیشتری مشتری** که به‌طور مستقل بین هر دور ارتباطی کار می‌کنند.

1. **افزایش محاسبات در هر Client:**

* به‌جای انجام محاسبات ساده مانند یک گام گرادیان نزولی، هر مشتری محاسبات پیچیده‌تری را بین هر دور ارتباطی انجام دهد.

**افزایش محاسبات روی هر مشتری، پس از رسیدن به حداقل میزان موازی‌سازی، عامل اصلی بهبود سرعت آموزش بوده است.**

Related Works

1. **میانگین‌گیری مدل‌های محلی در محیط‌های توزیع شده**

* این پژوهش‌ها **فقط محیط‌های خوشه‌ای (Cluster) یا مراکز داده را در نظر گرفته‌اند**، که در آن‌ها تعداد Workers حداکثر ۱۶ بوده و **شبکه‌های پرسرعت** استفاده شده است.
* همچنین داده‌ها به صورت **Non-IID** در نظر گرفته نشده است.

1. **حفظ داده‌های حساس کاربران روی دستگاه‌های شخصی**

* داده‌ها به صورت **Non-IID** در نظر گرفته نشده است.
* ارزیابی تجربی محدود

1. **روش‌های بهینه‌سازی توزیع‌شده در حالت Convex**

* مسائل Convex مسائلی هستند که تابع هزینه آن‌ها فقط یک مینیمم یکتا دارد که همان بهنیه سراسری است.
* در حالی که در یادگیری فدرال شبکه‌های عصبی کار می‌شود که Convex نیستند.
* همچنین در این مطالعات فرض شده که تعداد مشتریان کم و داده هر مشتری زیاد است که در یادگیری فدرال درست برعکس آن فرض می‌شود یعنی تعدا مشتریان زیاد و داده هر مشتری کم است.
* همچنین داده‌ها به صورت **Non-IID** در نظر گرفته نشده است.

1. **روش‌های غیرهمگام (Async) برای بهینه‌سازی توزیع‌شده**

* این روش‌ها تعداد به‌روزرسانی زیادی دارند که برای یادگیری فدرال بهینه نیستند.

1. **روش‌های توافق توزیع‌شده (Distributed Consensus Algorithms)**

* این روش بهینه‌سازی به این شکل است که تمامی Nodeها روی یک مقدار مشخصی به توافق برسند.
* این روش برای بهینه‌سازی ارتباطی در حضور تعداد زیادی مشتری بهینه نیست.

1. **میانگین‌گیری یکباره (One-Shot Avg) در یادگیری فدرال**

* در یک روش ساده، **هر مشتری مدل خود را روی داده‌های محلی‌اش آموزش می‌دهد و سپس مدل‌های تمام مشتریان میانگین‌گیری شده و مدل نهایی ساخته می‌شود.**
* **فرق این روش با رویکرد یادگیری فدرال با مدل همگام که در بالا گفته شد این است که یکبار انجام می‌شود.**
* **همچنین فرض Non-IID نیز در نظر گرفته نشده است.**
* در بدترین حالت، مدل نهایی هیچ بهبودی نسبت به آموزش روی یک مشتری ندارد.
* همچنین مدل نهایی در صورتی که توزیع داده‌ها به صورت Non-IID باشد مدل کلی خیلی نزدیک به مشتریانی است که داده‌ها غالب دارند و داده دیگر مشتریان در نظر گرفته نمی‌شود.

The FederatedAveraging Algorithm

**SGD را می‌توان به‌طور ساده در مسئله بهینه‌سازی فدرال به کار برد، به این صورت که در هر دور ارتباطی، تنها یک محاسبه گرادیان روی یک دسته کوچک از داده‌ها (مثلاً روی یک مشتری انتخاب‌شده به‌صورت تصادفی) انجام شود.** این روش از نظر محاسباتی کارآمد است، اما برای تولید مدل‌های مناسب، به تعداد بسیار زیادی از دورهای آموزشی نیاز دارد.

به‌عنوان مثال، حتی با استفاده از یک روش پیشرفته مانند batch normalization برای آموزش مجموعه داده MNIST به 50000 مرحله آموزشی با batch size برابر با ۶۰ نیاز است. در این مقاله روی مجموعه داده CIFAR-10 آزمایش انجام می‌شود.

در تنظیمات یادگیری فدرال مشارکت تعداد زیادی از مشتریان هنزیه کمی از نظر زمان واقعی (Wall-Clock Time) دارد. بنابراین در روش پایه خود از Large-Batch Synchronous SGD استفاده می‌کنیم که آزمایش‌ها نشان داده است این رویکرد در محیط‌های مرکز داده پیشرفته‌ترین روش موجود است و عملکرد بهتری نسبت به روش‌های Async دارد.

برای اعمال این روش در هر دور ارتباطی یک نسبت از مشتریان را انتخاب کرده و گرادیان تابع هزینه را روی تمام داده‌های آن مشتریان محاسبه خواهیم کرد.

**این الگوریتم FederatedSGD یا FedSGD نامیده می‌شود.**

**نکته: چرا روی همه داده‌های هر مشتری این کار را انجام می‌دهیم؟**

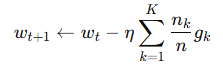
هدف ما این است که با **افزایش مقدار محاسبات روی هر مشتری**، **تعداد دورهای ارتباطی موردنیاز برای آموزش مدل را کاهش دهیم.**

**توضیحات الگوریتم FedSGD به شرح زیر است:**

1. در هر **دور ارتباطی (round)** یک مجموعه تصادفی از **مشتریان** انتخاب می‌شود که در فرآیند آموزش شرکت کنند. این تعداد با کنترل می‌شود، که نشان‌دهنده **نسبت مشتریان فعال در هر دور** است.
2. هر مشتری **گرادیان تابع هزینه** را روی داده‌های محلی خود محاسبه می‌کند که این مقدار **میانگین گرادیان محاسبه‌شده روی داده‌های محلی مشتری** است.



1. **سرور مرکزی** تمام گرادیان‌های ارسال‌شده را جمع‌آوری و میانگین‌گیری می‌کند.





1. مدل به‌روز شده دوباره به مشتریان ارسال می‌شود و فرآیند در دور بعدی تکرار می‌شود.

**مشکل FedSGD چیست؟**

* **در هر دور ارتباطی مشتریان فقط یک گام از SGD را اجرا می‌کنند.**
* **برای رسیدن به یک مدل مناسب تعداد زیادی از دورهای ارتباطی لازم است.**
* **در یادگیری فدرال به دلیل ارتباطات محدود، این مدل ایده‌آل نیست.**

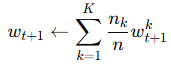
**توضیحات روش FedAvg که بهبود یافته FedSGD است:**

به‌جای اینکه فقط یک گام گرادیان نزولی اجرا کنیم، هر مشتری چندین بار مدل را روی داده‌های خود آپدیت می‌کند. سپس مدل‌های محلی میانگین‌گیری شده و مدل کلی به‌روزرسانی می‌شود.

1. در هر **دور ارتباطی**، یک مجموعه تصادفی از مشتریان انتخاب می‌شود.
2. **به‌جای انجام یک گام گرادیان نزولی**، هر مشتری چندین گام به‌روزرسانی انجام می‌دهد.



1. پس از چندین گام به‌روزرسانی، مدل‌های محلی به سرور ارسال شده و میانگین‌گیری می‌شوند.



**این میانگین‌گیری باعث می‌شود مدل کلی با دقت بیشتری به‌روز شود و نیاز به ارتباطات کمتری داشته باشیم.**

**پارامترهای کلیدی که میزان Computing را کنترل می‌کنند:**

1. **پارامتر که نسبت مشتریانی که در هر دور انتخاب می‌شوند را مشخص می‌کند.**

* اگر باشد تمام مشتریان در هر دور انتخاب می‌شوند.
* اگر کوچکتر باشد، فقط برخی از مشتریان در هر دور شرکت خواهند کرد.

1. **پارامتر تعداد دفعاتی است که هر مشتری مدل را روی داده‌های خود به روز می‌کند.**

* اگر باشد هر مشتری فقط **یک گام** به‌روزرسانی انجام می‌دهد درست مثل FedSGD
* اگر بزرگتر باشد هر مشتری چندین گام به‌روزرسانی قبل از ارسال مدل به سرور انجام می‌دهد.

1. **پارامتر که اندازه دسته‌های داده (Minibatch Size) که در هر به‌روزرسانی استفاده می‌شود را مشخص می‌کند.**

* اگر باشد، کل داده‌های محلی مشتری یکجا استفاده می‌شوند.
* اگر کوچک باشد، مشتری از دسته‌های کوچک‌تر (Minibatches) استفاده می‌کند.

**اگر و باشد در واقع FedAvg دقیقا شبیه به FedSGD خواهد شد!**

مقدار محاسباتی که هر مشتری انجام می‌دهد به تعداد به‌روزرسانی‌های محلی در هر دور بستگی دارد:







1. **اگر مقدار زیاد باشد**

* مشتری **محاسبات بیشتری انجام داده و مدل دقیق‌تر می‌شود**، اما ممکن است پردازش روی دستگاه‌های ضعیف سخت شود.

1. **اگر مقدار کم باشد**

* ارتباطات با سرور **افزایش پیدا می‌کند** که می‌تواند **هزینه ارتباطی را بالا ببرد.**