



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

ابزارهایی در حوزه یادگیری عمیق

نگارش

امین عبدی پوراصل

۴۰۱۱۳۳۰۱۱

استاد درس

دکتر صفابخش

بهمن ۱۴۰۲

صفحه	فهرست مطالب
۴	مقدمه
۵	۱- درباره Notebook ها
۵	۱-۱- Jupyter Notebook
۶	۱-۲- جایگزین های Jupyter Notebook
۶	۱-۲-۱- DeepNote
۷	۱-۲-۲- Kaggle Notebooks
۸	۱-۲-۳- Microsoft Azure Notebooks
۸	۱-۲-۴- Databricks
۹	۱-۲-۵- CoCalc
۱۰	۱-۳- مقایسه پلتفرم های مختلف
۱۰	۲- Google Colaboratory
۱۲	۳- CUDA & CuDNN
۱۲	۳-۱- CUDA
۱۳	۳-۲- CuDNN
۱۴	۴- ONNX
۱۵	۵- Fast.ai
۱۵	۶- LightGBM
۱۷	منابع و مراجع

صفحه

فهرست اشکال

- شکل ۱: نمایی از اجرای کد در سلول به همراه markdown در Jupyter Notebook [1] ۶
- شکل ۲: نمایی از بلوک DeepNote [4] ۷
- شکل ۳: نمایشی از کد زنی در پلتفرم Microsoft Azure Notebooks [3] ۹
- شکل ۴: مقایسه بین پلتفرم‌های مختلف کد نویسی دارای نوت‌بوک ۱۰
- شکل ۵: نحوه اتصال Google Colab با Google Drive ۱۱

مقدمه

یادگیری عمیق، زیرمجموعه ای از یادگیری ماشین، به عنوان یک روش قدرتمند برای حل مسائل پیچیده در حوزه‌های مختلف، از جمله بینایی کامپیوتر، پردازش زبان طبیعی، و یادگیری تقویتی پدیدار شده است. در قلب پروژه‌های یادگیری عمیق، ابزارها و کتابخانه‌های زیادی وجود دارد که برای ساده‌سازی توسعه، آموزش و استقرار شبکه‌های عصبی عمیق طراحی شده‌اند. این ابزارها طیف وسیعی از قابلیت‌ها، از ساخت و بهینه‌سازی مدل‌ها تا تجسم نتایج و مقیاس‌بندی محاسبات در سیستم‌های توزیع‌شده را در بر می‌گیرند.

چشم انداز ابزارهای یادگیری عمیق گسترده است و به طور مداوم در حال تحول است که هم توسط پیشرفت‌های تحقیقاتی و هم تقاضاهای عملی از برنامه‌های کاربردی صنعت هدایت می‌شود. از جمله برجسته ترین ابزارها می‌توان به TensorFlow، PyTorch و Keras اشاره کرد که با ارائه چارچوب‌های قابل دسترس و انعطاف پذیر برای محققان و پزشکان، یادگیری عمیق را دموکراتیک کرده اند. علاوه بر این، کتابخانه‌های بنیادی مانند NumPy و scikit-learn نقشی مهم در پیش‌پردازش داده‌ها، مهندسی ویژگی‌ها و ارزیابی مدل ایفا می‌کنند و زمینه را برای یادگیری عمیق موثر فراهم می‌کنند.

در این گزارش، برخی از ابزارهای ضروری برای یادگیری عمیق را بررسی می‌کنیم و ویژگی‌های کلیدی، موارد استفاده و کمک به پیشرفت این حوزه را برجسته می‌کنیم. این گزارش با نگاه به فصل ۱۹ کتاب Dive into Deep learning نوشته ژانگ و همکاران [1] تهیه و تدوین شده است.

۱- درباره Notebookها

۱-۱- Jupyter Notebook

Jupyter Notebook یک IDE منبع باز است که افراد می‌توانند به راحتی از آن استفاده کنند. بسیاری از افراد از محیط آن‌اکنونا برای ساخت پلتفرم‌های داده خود استفاده می‌کنند و Jupyter Notebook می‌تواند به راحتی به آن متصل شود.

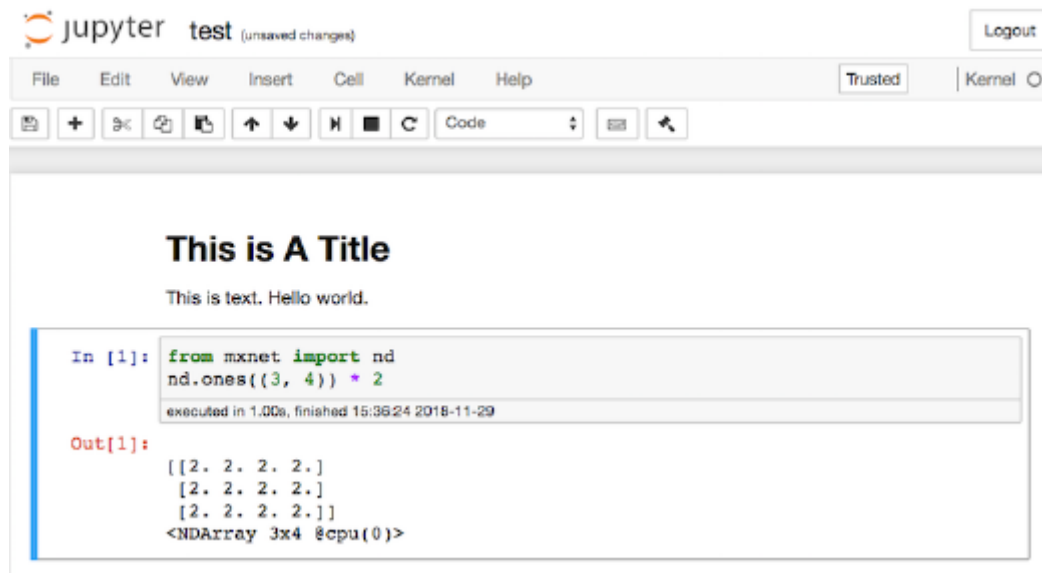
در هسته خود، Jupyter Notebook یک رابط مبتنی بر وب را ارائه می‌دهد که به کاربران اجازه می‌دهد اسنادی حاوی کد زنده، معادلات، تجسم‌ها و متن توضیحی ایجاد و به اشتراک بگذارند. این ترکیب منحصر به فرد یک جریان کاری یکپارچه را ایجاد می‌کند که در آن توسعه‌دهندگان می‌توانند کد را در یک محیط واحد بنویسند، اجرا کنند و تکرار کنند، همگی در حالی که فرآیند فکر خود را مستند می‌کنند و منجر به یک روایت منسجم می‌شود. پشتیبانی از چندین زبان برنامه نویسی، از جمله پایتون، R و جولیا، تطبیق پذیری آن را بیشتر می‌کند و طیف متنوعی از موارد استفاده و ترجیحات را در جامعه یادگیری عمیق ارائه می‌دهد. همچنین شامل موارد زیر است:

- فرمول‌های ریاضی، متن و رسانه تصویری،
- ویژگی‌های جمع آوری، تمیز کردن، تجزیه و تحلیل و تجسم داده‌ها،
- امکان ایجاد و تحلیل مدل‌های یادگیری ماشینی.

با تجزیه کد به سلول‌های قابل مدیریت، کاربران می‌توانند هر سلول را به طور مستقل اجرا کنند، نتایج میانی را بررسی کرده و خطاها را به طور مکرر اشکال زدایی کنند. این فرآیند توسعه مکرر نه تنها نمونه سازی و آزمایش را تسریع می‌کند، بلکه درک عمیق تری از الگوریتم‌ها و داده‌های اساسی را ارتقا می‌دهد. علاوه بر این، توانایی ذخیره و اشتراک گذاری نوت‌بوک‌ها در قالب‌های مختلف، مانند HTML، PDF، یا نمایش اسلاید، همکاری و انتشار دانش را در بین تیم‌ها و جوامع تسهیل می‌کند.

Jupyter Notebook محبوبیت گسترده‌ای در میان جامعه علم داده به دست آورد، تا جایی که به ابزار پیش فرض برای تحقیق تبدیل شد. به لطف ویژگی‌های آن، به انتخاب واقعی دانشمندان داده برای به اشتراک گذاشتن کار، مشاهده و تجزیه و تحلیل داده‌ها در طول فرآیند توسعه، نمونه‌سازی اولیه و تجزیه

و تحلیل اکتشافی تبدیل شده است. شکل ۱ نمایی از اجرای کد در یک سلول در Jupyter Notebook را نمایش می‌دهد [2].



شکل ۱: نمایی از اجرای کد در سلول به همراه markdown در Jupyter Notebook [1]

۲-۱- جایگزین‌های Jupyter Notebook

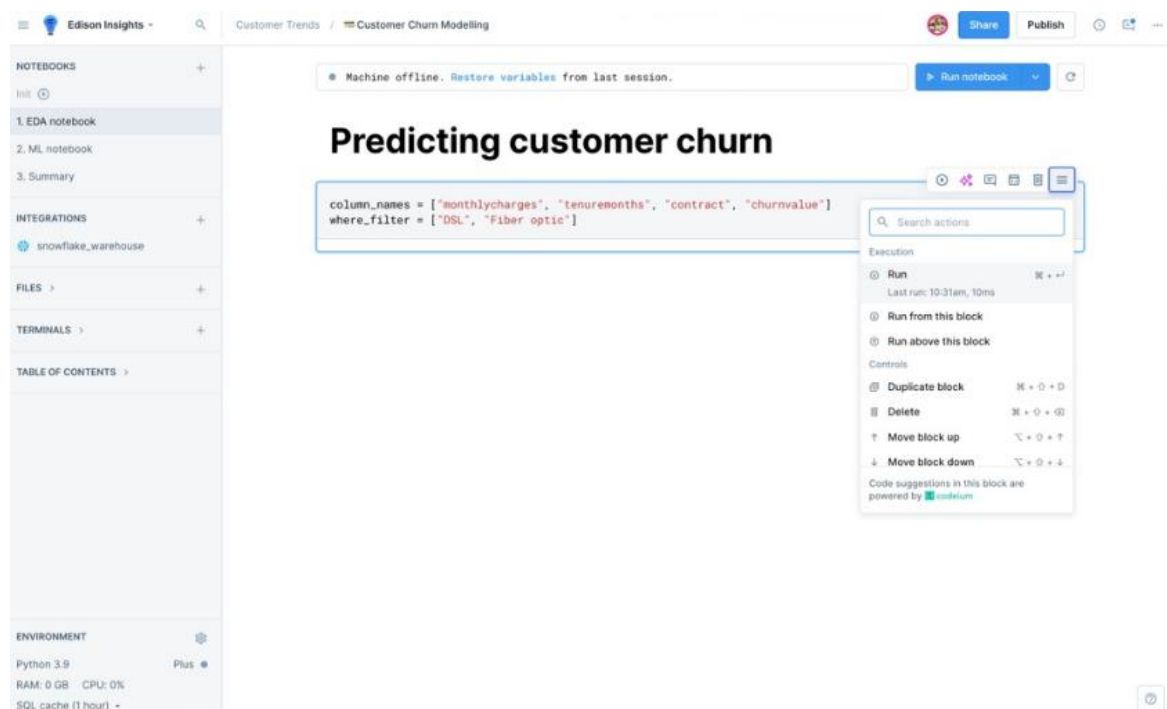
بیاپید نگاهی به برخی دیگر از نوت‌بوک‌های دیگر بیندازیم که ممکن است به آنها علاقه مند باشید. آنها عملکردی مشابه با نوت‌بوک Jupyter دارند، اما امکان همکاری یکپارچه و انعطاف پذیری و سفارشی سازی بیشتر را نیز دارند و در کارهای تیمی ممکن است فواید بیشتری داشته باشند [3].

۱-۲-۱- DeepNote

DeepNote یک پلتفرم آنلاین است که یک محیط مشارکتی و تعاملی برای پروژه‌های هوش مصنوعی و یادگیری ماشین فراهم می‌کند. این پلتفرم ویژگی‌های نوت‌بوک‌های Jupyter را با قابلیت‌های اضافی ترکیب می‌کند تا کار توسعه‌دهندگان را ساده‌تر کند.

این پلتفرم از نوت‌بوک‌های Jupyter پشتیبانی می‌کند و به کاربران اجازه می‌دهد سلول‌های کد پایتون را ایجاد، ویرایش و اجرا کنند، همچنین سلول‌های علامت گذاری برای مستندات و توضیحات ایجاد نمایند. DeepNote امکان همکاری بین اعضای تیم را در همان نوت‌بوک فراهم می‌کند. چندین کاربر

می‌توانند به طور همزمان کار کنند و تغییرات و مشارکت‌های یکدیگر را در زمان واقعی مشاهده کنند، که کار تیمی را افزایش می‌دهد و به اشتراک گذاری دانش را تسهیل می‌کند. DeepNote منابع محاسباتی مبتنی بر فضای ابری را ارائه می‌دهد و نیازی به راه‌اندازی و نگهداری زیرساخت‌های کاربران را از بین می‌برد. این به کاربران اجازه می‌دهد تا از منابع محاسباتی قدرتمند برای اجرای مدل‌های یادگیری ماشین، تجزیه و تحلیل مجموعه داده‌های بزرگ و انجام آزمایش‌ها بدون نگرانی در مورد محدودیت‌های سخت افزاری استفاده کنند. همچنین پشتیبانی GPU را برای تسريع محاسبات یادگیری عمیق فراهم می‌کند. این ویژگی به ویژه برای آموزش شبکه‌های عصبی عمیق در مجموعه داده‌های بزرگ مفید است، زیرا GPUها برای مدیریت محاسبات موازی درگیر در آموزش شبکه‌های عصبی مناسب هستند. در شکل ۲، نمایی از یک بلوک DeepNote را مشاهده می‌نمایید.



شکل ۲: نمایی از بلوک DeepNote [4]

۱-۲-۲- Kaggle Notebooks

Kaggle Notebooks یک پلتفرم نوت‌بوک مبتنی بر ابر برای علاقه‌مندان به علم داده و یادگیری ماشین است. این پلتفرم امکان دسترسی به منابع سخت افزاری GPU و TPU را برای اجرای مدل‌های یادگیری ماشینی و یادگیری عمیق فراهم می‌کند. همچنین شامل تعامل با Kaggle API، پشتیبانی از

کنترل نسخه داده با Git، و امکان اشتراک گذاری و همکاری آسان نوت بوک ها با اعضای تیم است. همچنین نوت بوک های Kaggle یک جایگزین عالی برای پروژه های علم داده است زیرا به کاربران اجازه می دهد به راحتی در این مسابقات Kaggle شرکت کنند و با سایر کاربران و توسعه دهندگان همکاری کنند یا حتی مسابقه با دادگان خاص را تعریف کنند و دیگران در آن ها شرکت نمایند. کاربران می توانند به طور یکپارچه به مجموعه داده های موجود در Kaggle دسترسی داشته باشند و مستقیماً در نوت بوک خود بارگذاری کنند، و کاوش و تجزیه و تحلیل داده ها را بدون نیاز به دانلود یا آپلود دستی فایل ها آسان می کند. شبیه به نوت بوک های Jupyter، نوت بوک های Kaggle از سلول های Markdown پشتیبانی می کنند و به کاربران امکان می دهند متن، معادلات، تصاویر و لینک های فرمت شده را در تجزیه و تحلیل خود بگنجانند. این امر مستندسازی و توضیح کد و نتایج را در خود نوت بوک تسهیل می کند.

۱-۲-۳ Microsoft Azure Notebooks

Microsoft Azure Notebooks نیز یک پلتفرم مبتنی بر ابر برای پروژه های علم داده و یادگیری ماشین است. Microsoft Azure Notebook از طیف گسترده ای از زبان ها و کتابخانه های برنامه نویسی، از جمله Python، R، #F و Julia پشتیبانی می کند. این پلتفرم نیز امکان دسترسی به GPU و TPU را به کاربران می دهد. علاوه بر این، Azure Notebooks به طور یکپارچه با سایر سرویس های Microsoft Azure ادغام می شود [5]. این پلتفرم در VScode در دسترس است و می توان از آن استفاده کرد. شکل ۳، نمایشی از این پلتفرم می باشد.

۱-۲-۴ Databricks

نوت بوک های Databricks به عنوان یک ابزار اساسی در پلتفرم Databricks عمل می کنند و ایجاد جریان های کاری علم داده و یادگیری ماشین را تسهیل می کنند و در عین حال همکاری بین اعضای تیم را تقویت می کنند. آنها به طور یکپارچه با خوشه های Databricks کار می کنند و به کاربران این امکان را می دهند که از منابع محاسباتی به طور موثر استفاده کنند. با ویژگی هایی مانند تجسم داده های تعاملی و پشتیبانی از زبان های برنامه نویسی مختلف، از جمله پایتون و R، نوت بوک های Databricks کاربران را قادر می سازد تا الگوریتم های مختلف را آزمایش کنند.


```

import numpy as np
import pandas as pd
PREVIOUS_MAX_ROWS = pd.options.display.max_rows
pd.options.display.max_rows = 20
np.random.seed(12345)
import matplotlib.pyplot as plt
plt.rc('figure', figsize=(10, 6))
np.set_printoptions(precision=4, suppress=True)

df = pd.DataFrame({'key1': ['a', 'a', 'b', 'b', 'a'],
                  'key2': ['one', 'two', 'one', 'two', 'one'],
                  'data1': np.random.randn(5),
                  'data2': np.random.randn(5)})

df


```

	key1	key2	data1	data2
0	a	one	-0.204708	1.393406
1	a	two	0.478943	0.092908
2	b	one	-0.519439	0.281746
3	b	two	-0.555730	0.769023
4	a	one	1.965781	1.246435

```

grouped = df['data1'].groupby(df['key1'])
grouped


```

```

grouped.mean()


```

شکل ۳: نمایشی از کد زنی در پلتفرم Microsoft Azure Notebooks [3]

۱-۲-۵ CoCalc

CoCalc، مخفف محاسبه مشارکتی^۱، یک پلتفرم آنلاین است که یک محیط مشترک برای ریاضیات محاسباتی، علوم داده و زمینه‌های مرتبط فراهم می‌کند. این پلتفرم نوت‌بوک‌های Jupyter را به عنوان رابط اصلی خود ارائه می‌دهد که به کاربران امکان می‌دهد کد را در زبان‌های برنامه نویسی مختلف از جمله Python، R، Julia و SageMath بنویسند، اجرا کنند و به اشتراک بگذارند. همچنین این پلتفرم چندین کاربر را قادر می‌سازد به طور همزمان روی یک سند کار کنند.

۳. **پشتیبانی از CoCalc: LaTeX** پشتیبانی داخلی برای LaTeX، یک سیستم حروفچینی که معمولاً برای نوشتن اسناد ریاضی و علمی استفاده می‌شود، ارائه می‌دهد. کاربران می‌توانند اسناد

¹ Collaborative Calculation

LaTeX را به صورت یکپارچه در بستر ایجاد کنند و با آنها همکاری کنند. CoCalc دسترسی به منابع محاسباتی از جمله سرورهای CPU و GPU را فراهم می‌کند و به کاربران امکان می‌دهد محاسبات، شبیه سازی‌ها و وظایف پردازش داده‌ها را فشرده انجام دهند. CoCalc ویژگی‌هایی را برای سازماندهی و مدیریت پروژه‌ها، از جمله کنترل نسخه با Git، به اشتراک گذاری پروژه و ردیابی وظایف ارائه می‌دهد.

۳-۱- مقایسه پلتفرم‌های مختلف

در شکل ۴، مقایسه بین پلتفرم‌های مختلف از لحاظ ویژگی‌های مختلف را مشاهده می‌نمایید.

Notebook	Programming languages	Version control	Cloud storage	Security	Collaboration	Other features
Deeptime	Python, R, Julia, SQL	–	Native	Yes	Real-time collaboration	
Kaggle Notebooks	Python, R	–	Native	–	–	
Microsoft Azure Notebooks	Python, R, F#, Julia	Git	Azure storage	via Azure Cloud	No	
Databricks Notebooks	Python, Scala, R	–	S3 storage	via Databricks	Yes	
CoCalc	Python, R, Julia	Git	Native			LaTeX, computer algebra systems

شکل ۴: مقایسه بین پلتفرم‌های مختلف کد نویسی دارای نوت‌بوک

۲- Google Colaboratory

این پلتفرم نیز مبتنی بر نوت‌بوک می‌باشد اما دلیل آن که آن را به صورت جداگانه بررسی می‌کنیم، محبوبیت بالای آن و همچنین کاربرد بسیار بالای آن می‌باشد.

Google Colab، مخفف Google Colaboratory، یک پلتفرم مبتنی بر ابر ارائه شده توسط Google است که یک محیط محاسباتی تعاملی رایگان برای نوشتن، اجرا و به اشتراک گذاری کد پایتون ارائه می‌دهد. این نرم افزار حول محور نوت‌بوک‌های Jupyter ساخته شده است و به طور گسترده توسط محققان، دانشجویان و توسعه دهندگان برای اهداف مختلف از جمله تجزیه و تحلیل داده‌ها، یادگیری ماشینی و محاسبات علمی استفاده می‌شود. Google Colab دسترسی رایگان به منابع رایانش ابری، از جمله نمونه‌های CPU، GPU و TPU را فراهم می‌کند و به کاربران امکان می‌دهد بدون نیاز به سخت افزار محلی قدرتمند، کد را اجرا کرده و محاسبات را انجام دهند. Google Colab چندین کاربر را قادر می‌سازد تا به طور همزمان روی یک نوت‌بوک کار کنند. کاربران می‌توانند نوت‌بوک‌های خود را با دیگران به اشتراک بگذارند و در زمان واقعی، مشابه Google Docs، همکاری کنند. نوت‌بوک‌های Colab در یک حساب Google Drive ذخیره می‌شوند و می‌توانند مانند سایر فایل‌های Google Drive با سایر کاربران به اشتراک گذاشته شوند. نوت‌بوک‌ها همچنین دارای ویژگی ذخیره خودکار هستند، اما از ویرایش همزمان پشتیبانی نمی‌کنند، بنابراین همکاری باید سریالی باشد تا موازی.

Google Colab با کتابخانه‌ها و فریم ورک‌های محبوب پایتون، از جمله TensorFlow، PyTorch، و scikit-learn از پیش نصب شده است که شروع کار بر روی یادگیری ماشین و پروژه‌های یادگیری عمیق را آسان می‌کند.

Google Colab به طور یکپارچه با Google Drive ادغام می‌شود و به کاربران امکان می‌دهد به طور مستقیم به نوت‌بوک‌ها دسترسی داشته باشند و در فضای ذخیره سازی Google Drive خود ذخیره کنند. این ویژگی فرآیند مدیریت و سازماندهی پروژه‌ها و مجموعه داده‌ها را ساده می‌کند (شکل ۵).



```
from google.colab import drive
drive.mount('/my_drive') |
```

شکل ۵: نحوه اتصال Google Colab با Google Drive

Google Colab به کاربران اجازه می‌دهد تا به منابع داده‌های خارجی مانند Google Drive، Google Cloud Storage و مخازن GitHub دسترسی داشته باشند و وارد کردن مجموعه داده‌ها و فایل‌ها را به نوت‌بوک آنها آسان می‌کند.

برای استفاده از Colaboratory، کافیست یک حساب Google داشته باشید. می‌توانید کد پایتون را برای اجرا در سلول کد وارد کنید. کد می‌تواند کاملاً مستقل باشد یا از کتابخانه‌های مختلف پایتون وارد شود. یک نوت‌بوک را می‌توان به‌عنوان یک گزارش متحرک از کار در نظر گرفت، با قطعه‌های کد قبلی که دیگر به نفع نسخه‌های بعدی اجرا نمی‌شوند، یا به‌عنوان مجموعه‌ای در حال تکامل از بلوک‌های کد در نظر گرفته شده برای اجرای مداوم در نظر گرفته می‌شوند. منوی Runtime گزینه‌های اجرا مانند Run all، Run before یا Run the focused cell را برای مطابقت با هر یک از روش‌ها ارائه می‌دهد.

۳- CUDA & CuDNN

۳-۱- CUDA

#CUDA: محاسبه معماری یکپارچه دستگاه

CUDA که مخفف Compute Unified Device Architecture است، یک پلتفرم محاسباتی موازی و مدل برنامه نویسی است که توسط NVIDIA توسعه یافته است. این پلتفرم به توسعه دهندگان اجازه می‌دهد تا از قدرت محاسباتی پردازنده‌های گرافیکی NVIDIA (واحد پردازش گرافیکی) برای سرعت بخشیدن به وظایف محاسباتی همه منظوره، از جمله پردازش داده‌ها، شبیه سازی‌های علمی و محاسبات یادگیری عمیق استفاده کنند. CUDA محاسبات موازی را با بارگذاری وظایف محاسباتی فشرده از CPU به GPU فعال می‌کند. این موازی سازی به طور قابل توجهی سرعت محاسبات را افزایش می‌دهد، زیرا GPUها برای مدیریت هزاران رشته به طور همزمان بهینه شده اند.

CUDA یک مدل برنامه نویسی ارائه می‌دهد که به توسعه دهندگان اجازه می‌دهد تا به شکل موازی کد بنویسند. این مدل شامل ویژگی‌هایی مانند هسته‌ها (توابع موازی اجرا شده در GPU)، همگام سازی رشته‌ها و مدیریت حافظه است.

همچنین NVIDIA جعبه ابزار CUDA را ارائه می‌دهد، یک کیت توسعه نرم افزار جامع (SDK) که شامل کتابخانه‌ها، کامپایلرها، اشکال زداهای دیگر برای توسعه برنامه‌های کاربردی CUDA است. این جعبه ابزار از سیستم عامل‌های مختلف از جمله ویندوز، لینوکس و macOS پشتیبانی می‌کند. بسیاری از فریم ورک‌های یادگیری عمیق محبوب، مانند TensorFlow، PyTorch و MXNet، دارای پشتیبانی CUDA هستند که به توسعه دهندگان این امکان را می‌دهد تا برای عملکرد سریع تر، شبکه‌های عصبی عمیق را بر روی پردازنده‌های گرافیکی NVIDIA آموزش داده و مستقر کنند. پردازنده‌های گرافیکی NVIDIA مجهز به هسته‌های CUDA هستند، واحدهای پردازش تخصصی که برای انجام محاسبات موازی به طور موثر طراحی شده اند. تعداد هسته‌های CUDA در مدل‌های GPU متفاوت است، با پردازنده‌های گرافیکی سطح بالاتر، هسته‌های CUDA بیشتری برای افزایش قدرت محاسباتی دارند.

CUDA با بهره گیری از قابلیت‌های پردازش موازی پردازنده‌های گرافیکی NVIDIA، محاسبات با کارایی بالا را امکان پذیر می‌کند، که منجر به محاسبات بسیار سریع تر در مقایسه با محاسبات مبتنی بر CPU سنتی می‌شود. همچنین CUDA به توسعه دهندگان این امکان را می‌دهد که برنامه‌های خود را در چندین پردازنده گرافیکی مقیاس بندی کنند و توان محاسباتی را بیشتر افزایش دهند و شبیه سازی‌ها و وظایف پردازش داده در مقیاس بزرگتر را فعال کنند [6].

۳-۲- CuDNN

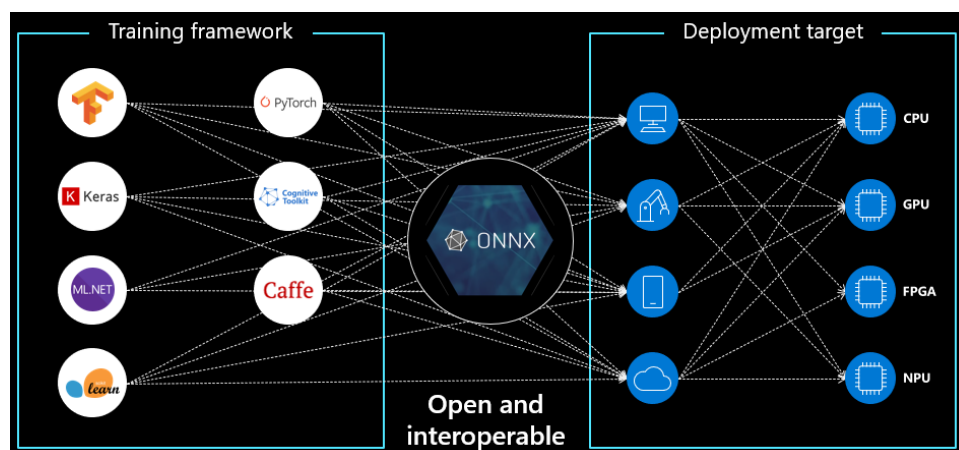
CuDNN مخفف CUDA Deep Neural Network کتابخانه ای است که توسط GPU شتاب می‌گیرد که توسط NVIDIA به طور خاص برای چارچوب‌های یادگیری عمیق توسعه یافته است. این برنامه پیاده‌سازی‌های بسیار بهینه‌شده‌ای از الگوریتم‌ها و ابتدایی‌های یادگیری عمیق را ارائه می‌دهد که برای استفاده از قابلیت‌های پردازش موازی پردازنده‌های گرافیکی NVIDIA طراحی شده‌اند.

CuDNN پیاده سازی‌های بهینه شده عملیات یادگیری عمیق ضروری، مانند توابع کانولوشن، ادغام، عادی سازی و فعال سازی را ارائه می‌دهد. این پیاده سازی‌ها بسیار کارآمد هستند و از قدرت پردازش موازی GPUها برای تسريع محاسبات استفاده می‌کنند. CuDNN نقش مهمی در تسريع محاسبات یادگیری عمیق در پردازنده‌های گرافیکی NVIDIA ایفا می‌کند، و پیاده‌سازی‌های بسیار بهینه‌شده‌ای از

الگوریتم‌ها و اولیه‌های یادگیری عمیق ضروری را ارائه می‌دهد. ادغام یکپارچه آن با چارچوب‌های یادگیری عمیق محبوب و تلاش‌های بهینه‌سازی مستمر، آن را به ابزاری ارزشمند برای محققان، توسعه‌دهندگان و دانشمندان داده تبدیل می‌کند که به دنبال استفاده از شتاب GPU برای پروژه‌های یادگیری عمیق خود هستند.

۴- ONNX

ONNX، مخفف Open Neural Network Exchange، یک قالب منبع باز برای نمایش مدل‌های یادگیری عمیق است. هدف آن ارائه قابلیت همکاری بین چارچوب‌های یادگیری عمیق مختلف است که به کاربران امکان می‌دهد مدل‌ها را در یک چارچوب آموزش دهند و بدون نیاز به تبدیل گسترده مدل، آن‌ها را در چارچوب دیگری استقرار دهند. ONNX تبادل یکپارچه مدل‌های یادگیری عمیق را بین فریم‌ورک‌های مختلف مانند TensorFlow، PyTorch، MXNet، Caffe و ... امکان پذیر می‌کند. این قابلیت همکاری، همکاری و ادغام در ابزارها و پلتفرم‌های مختلف را تسهیل می‌کند. با ONNX، توسعه دهندگان می‌توانند مدل‌های یادگیری عمیق را با استفاده از چارچوب دلخواه خود آموزش دهند و سپس آن‌ها را بدون نیاز به تغییرات قابل توجه در پلتفرم‌ها و محیط‌های سخت افزاری مختلف مستقر کنند. این انعطاف‌پذیری فرآیند استقرار را ساده می‌کند و زمان تولید را تسریع می‌کند. همچنین ONNX یک نمایش متوسط استاندارد شده برای مدل‌های یادگیری عمیق تعریف می‌کند که به اطمینان از سازگاری و سازگاری در چارچوب‌های مختلف کمک می‌کند. این نمایش شامل معماری مدل، پارامترها و محاسبات مورد نیاز برای استنتاج است [7].



۵- Fast.ai

Fast.ai یک کتابخانه یادگیری عمیق و پلتفرم آموزشی سطح بالا است که هدف آن قابل دسترس کردن یادگیری عمیق برای همه است. این پلتفرم توسط جرمی هوارد و ریچل توماس تأسیس شد و بر روی چارچوب PyTorch ساخته شده است. Fast.ai یک API ساده شده و مدل‌های از پیش آموزش دیده را برای کارهای رایج یادگیری عمیق ارائه می‌دهد که شروع یادگیری عمیق و تکرار سریع ایده‌ها را برای مبتدیان آسان می‌کند. این API سطح بالای ارائه شده بسیاری از پیچیدگی‌های یادگیری عمیق را انتزاعی می‌کند و ساخت و آموزش شبکه‌های عصبی را برای کاربران آسان تر می‌کند بدون نیاز به دانش گسترده از ریاضیات یا الگوریتم‌های اساسی.

Fast.ai شامل مدل‌های از پیش آموزش دیده برای طیف وسیعی از وظایف، مانند طبقه بندی تصویر، تشخیص اشیا، و پردازش زبان طبیعی است. این مدل‌های از پیش آموزش دیده را می‌توان به راحتی بر روی مجموعه داده‌های خود تنظیم کرد و به کاربران این امکان را می‌دهد تا با کمترین تلاش به عملکرد خوبی دست یابند. همچنین تکنیک‌های پیشرفته و بهترین شیوه‌ها را در یادگیری عمیق ترکیب می‌کند و اطمینان می‌دهد که کاربران به آخرین پیشرفت‌ها در این زمینه دسترسی دارند. همچنین مستندات و آموزش‌های جامعی را برای کمک به کاربران در درک مفاهیم و تکنیک‌های اساسی ارائه می‌دهد [8].

۶- LightGBM

LightGBM یک چارچوب تقویت کننده گرادیان منبع باز است که توسط مایکروسافت توسعه یافته است که برای ارائه عملکرد، مقیاس پذیری و کارایی بالا برای آموزش و استنتاج مدل‌های تقویت گرادیان

طراحی شده است. LightGBM به طور گسترده برای وظایف مختلف یادگیری ماشین از جمله طبقه بندی، رگرسیون و رتبه بندی استفاده می شود، بدین صورت که تقویت گرادیان را پیاده سازی می کند، یک تکنیک یادگیری گروهی محبوب که با ترکیب چندین یادگیرنده ضعیف (معمولاً درختان تصمیم) به طور متوالی، یک مدل پیش بینی قوی ایجاد می کند.

LightGBM برای عملکرد و استفاده از حافظه بهینه شده است، و آن را به ویژه برای مجموعه داده های مقیاس بزرگ و فضاهای ویژگی با ابعاد بالا کارآمد می کند. از یک الگوریتم مبتنی بر هیستوگرام برای محاسبه نقاط تقسیم استفاده می کند که مصرف حافظه را کاهش می دهد و سرعت تمرین را افزایش می دهد. LightGBM از استراتژی رشد درخت بر اساس برگ به جای عمق استفاده می کند که منجر به زمان آموزش سریعتر می شود.

LightGBM از EFB^1 پشتیبانی می کند، یک تکنیک بسته بندی ویژگی که ویژگی های مرتبط را با هم گروه بندی می کند تا ابعاد فضای ویژگی را کاهش دهد. این تکنیک به بهبود دقت مدل و سرعت آموزش کمک می کند، به خصوص برای مجموعه داده های با ابعاد بالا.

همچنین LightGBM پشتیبانی داخلی برای ویژگی های دسته بندی ارائه می دهد و به کاربران این امکان را می دهد که متغیرهای طبقه بندی را مستقیماً بدون نیاز به رمزگذاری یکباره رمزگذاری کنند. این کار پیش پردازش را ساده می کند و کارایی آموزش را بهبود می بخشد. همچنین از آموزش های موازی و توزیع شده در CPU های چند هسته ای و چارچوب های محاسباتی توزیع شده مانند Apache Hadoop و Apache Spark پشتیبانی می کند. این امکان آموزش مقیاس پذیر مدل ها را بر روی مجموعه داده های بزرگ در چندین گره محاسباتی فراهم می کند.

¹ Exclusive Feature Bundling

منابع و مراجع

- [1] Zhang, Aston, et al. *Dive into deep learning*. Cambridge University Press, 2023.
- [2] jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html
- [3] lakefs.io/blog/jupyter-notebook-10-alternatives-2023
- [4] <https://deepnote.com/docs/notebooks>
- [5] <https://www.cloudbank.org/tool/azure-notebooks>
- [6] <https://developer.nvidia.com/cuda-toolkit>
- [7] <https://microsoft.github.io/ai-at-edge/docs/onnx/>
- [8] <https://www.fast.ai/>