



چگونه برنامه های هوش مصنوعی «AI» ابر برنامه های مشابه خود را خلق می کند

Anil Ananthaswamy

متخصص هوش مصنوعی و فیزیک (۲۰۲۴)

پژوهش و تحقیق: دکتر بهرام پروانه طهرانی

مدت ها است که محققین جهت بهبود عملکرد هوش مصنوعی در صدد گسترش برنامه هایی هستند که بر اساس کمترین داده ها از طریق برنامه نویسان عمل کنند. اکنون این پروژه با کمک برنامه های بسیار پیشرفته به واقعیت تبدیل شده است

تکمیل و گسترش برنامه های هوش مصنوعی تا درجه بالایی بستگی به حجم اطلاعات اولیه دارد که برنامه نویسان به اصطلاح بر پایه آن برنامه را می نویسند. سعی بر این بوده است تا آنجا که امکان دارد ساختار مغز انسان را به عنوان مدلی جهت عملکرد برنامه و شبکه های آن اساس کار قرار دهند. این کوشش تا حدود ۱۰ سال پیش امکان نداشت و اکنون در زمان حال جدیداً در اختیار بودن اطلاعات کافی و در کنار آن افزایش روز افزون سرعت محاسبه سیستم های سخت افزاری موجب شده است تا بتوان دست به چنین عملی زد.

برنامه های مدرن هوش مصنوعی اکنون در موقعیتی قرار دارند تا بتوان حجم بسیار عظیمی از اطلاعات اولیه را به آنها تزریق نمود و جهت عملکرد سریع و بدون اشتباه باید از طریق برنامه نویسی ارتباطی منطقی میان میلیون ها و چه بسا میلیارد ها اطلاعات و داده های اولیه بر قرار نمود. چنین متغیرهایی (Variables) باعث ایجاد شبکه های مصنوعی عظیمی می گردند تا به اصطلاح نوروں های مصنوعی توان کارکرد قابل قبولی را به نمایش گزارند. هدف دست اند کاران علوم کامپیوتری (IT) در آن نهفته است تا چنین برنامه هایی را تا سرحد ممکن کارایی و سرعت ارتقاء بخشند. به تدریج معلوم شد که این عمل و بهبود بخشیدن به سیستم های موجود کاری بس دشوار بوده و آن طور که در ابتدا فکر می شد به راحتی و در مدت زمانی نسبتاً کوتاه امکان پذیر نیست. چه بسا طبق نظر آقای **Petar Velickovic** از متخصصین هوش مصنوعی در موسسه (Deep Mind – London) دست به چنین کاری زدن و به اصطلاح تمرین دادن سیستم ها ماه ها و شاید سال ها کار بدون وقفه را شامل می شود.

حال به نظر می آید که ورق برگشته و اکنون متخصصانی چون (Boris Knyazev, Graham Taylor) از دانشگاه گولپ در اونتاریو-کانادا با همکاری آقایان (Michal Drozdal, Adriano Romero-Soriano) از دپارتمان هوش مصنوعی موسسه فیس بوک در آمریکا موفق شده اند تا برنامه ای فوق تصور (هایپر پروگرام) را آماده و آزمایش نموده و در اختیار قرار دهند. برنامه یاد شده دارای قابلیت های فرا رایانه ای بوده و قادر است تا در صد بالایی شبکه های مغزی انسان را شبیه سازی نموده و فرایند تمرین دهی سیستم را به مراتب سهل نموده و بهبود بخشد.

برنامه هایپر پروگرام یاد شده قادر است تا در زمانی کمتر از چند دهم ثانیه تشخیص دهد که چه عواملی جهت حل سئوالات مطرح شده که توسط برنامه دیگری در حال جستجو است و الگوریتم آن با چه کمبودهای اطلاعاتی روبرو است را یافته و آن را اصلاح کند. به تفسیری دیگر این هایپر پروگرام قادر است تا به صورت خودکار برنامه های دیگر هوش مصنوعی را با استفاده از هوش مصنوعی خود تکمیل و ارتقاء دهد. لذا می توان گفت که اکنون به مرحله ای رسیده ایم که هوش مصنوعی به گونه ای خودکار قادر به تکمیل هوش خود شده است و انتظار می رود که در آینده حتی مرحله تمرین دهی چنین هایپر پروگرام هایی از ملزومات نباشد. بدیهی است که نتیجه کار خلاقانه کلیه متخصصین برنامه نویسی به آنجا ختم شود تا بتوان تجزیه و تحلیل شبکه های بغرنج فکری بر اساس مغز انسان را در جهت طراحی شبکه های جدید و هوشیارتر مصنوعی به واقعیت تبدیل کرد. ولیکن باید دانست آنچه که در آینده هوش مصنوعی قادر به واقعیت بخشیدن به آن نخواهد بود و هرگز به آن دست نخواهد یافت، درک احساسات انسانی بر اساس مدل عملکرد مغز انسان می باشد. بلی مغز انسان و حتی دیگر حیوانات پیشرفته را نمی توان با یک برنامه کامپیوتری و سخت افزار ساخت دست بشر مقایسه نمود. مغز و ذهنیت انسان جهت تکامل خود به میلیون ها سال زمان نیاز داشته تا به مرحله کنونی برسد و این تکامل همچنان ادامه دارد. پیشنهاد می شود جهت درک عملکرد خارق العاده مغز و ذهنیت انسان به کانال و سایت تلگرامی:

«جهان دانش – t.me/scientificstudies» رجوع شود.

چگونه نرم افزار های پیشرفته هوش مصنوعی قادر به خلق شبکه های جدید فکری (Neural Network) می باشند

در حال حاضر هایپر پروگرام های موجود ظرفیت و سرعت عملکرد خود را به بهترین وجهی به اثبات رسانده اند، ولیکن با در نظر گرفتن حجم عظیم دانسته ها و دانش روز می توان آنها را باز هم توسعه داده و بهبود بخشید. به نظر یکی از متخصصین مشهور هوش مصنوعی آقای (Velickovic)، در صورتی که متخصصین در این راه موفق شوند، علم آموزش و یادگیری رایانه ای، از جمله هوش مصنوعی، تکاملی باور نکردنی را در پی خواهد داشت. همان طور که اشاره شد نرم افزار های هوش مصنوعی جهت عملکرد سریع و درست باید از پیش به اصطلاح تمرین داده شوند. در این فرایند مقادیر عظیمی از اطلاعات در همه زمینه ها از طریق برنامه نویسی به نرم افزار خورنده شده و پس از آن نتایج دریافتی با نتایجی که انتظار می رود مقایسه می شود. هدف اصلی در طی این فاز آن است که تا حد امکان اشتباهات سیستم به عنوان مثال در تشخیص صحیح تصاویر از میان برداشته شود. در حال حاضر بهترین راه کار در نیل به اهداف اشاره شده از تکنیکی به نام «مسیر سیر نزولی – Stochastic gradient descent (SGD)» استفاده می کند. عملکرد الگوریتم (SGD) بر این اصل استوار است که مقادیر فراوانی اطلاعات مستند (به عنوان مثال تصاویر یک گربه و یا این گربه نیست) به نرم افزار تزریق شده و سعی بر آن است تا نتیجه خروجی نرم افزار تا سرحد ممکن با واقعیت عینی تصویر تطابق داشته و در صورت وقوع اشتباه، پارامتر های برنامه به ترتیبی تغییر داده می شوند تا نتیجه دریافتی هر چه بیشتر به واقعیت نزدیک شود. در ادامه تعداد اشتباهات به وقوع پیوسته از طریق تابعی به نام جدول اشتباهات رمز گزاری و شناسایی می شود. سعی بر آن است تا با کمک تغییرات جدید و تصحیح برنامه تعداد این گونه اشتباهات تنزل یابد. خوانندگان توجه داشته باشند که الگوریتم (SGD) روشی تعاملی بوده و بر این اصل استوار است که طی ارتقاء کیفی برنامه و از میان برداشتن خطا های به وقوع پیوسته، مطابق با جدول اشتباهات، مرحله به مرحله با هدف دسترسی به کمترین تعداد نتایج نادرست تعریف می شود. چه بسا گاهی ممکن نیست تا بتوان برنامه و الگوریتم آن را به صورت مطلق و صد در صد بهبود بخشیده و با واقعیت یکی کرد. بلی این مورد دال بر این واقعیت است که هرگز نمی توان و نباید توان مغز و ذهنیت انسان را با برنامه های هوش مصنوعی در هر سطحی مقایسه نمود. چه بسا ممکن است تا به گونه ای خطرناک منحرف و به بیراهه رویم!

بدیهی است زمانی موفق به انجام این کار می شویم که برنامه دیگری جهت بهبود بخشی و تکمیل چنین ابر برنامه ای در اختیار داشته باشیم. این بدان معنی است که مهندسين مجبور خواهند بود تا جهت طراحی و برنامه نویسی نورون های مصنوعی پیشاپیش خطوط ارتباط میان نورون های مصنوعی را شناخته و در این راستا تا حدود زیادی از احساسات شهودی خود و چند قانون سر انگشتی استفاده کنند. به طور معمول شبکه ها و نورون های مصنوعی از چندین لایه تشکیل شده اند، به نحوی که لایه اول مسئول دریافت اطلاعات و لایه آخر مسئولیت خروج نتایج را به عهده دارد. لایه ها و واحد های رایانه ای میانی وظیفه انتقال و پردازش داده ها در درون کل برنامه را به عهده دارند. بدیهی است که تعداد لایه های میانی و نورون های آنها کاملاً بستگی به نوع الگوریتم تعریف شده دارد. حال پرسش این است که چگونه می توان دریافت که چه الگوریتمی برای پردازش میلیون ها اطلاعات دریافتی و خلق پاسخ صحیح مناسب تر از دیگر ساختار یا الگوریتم ها می باشد؟

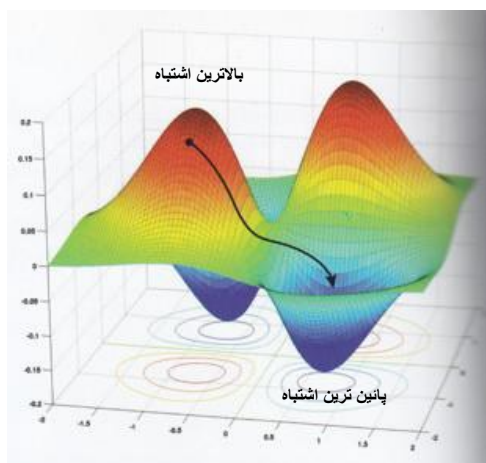
تئوری وار می توان پاسخ این پرسش را چنین داد که چندین ساختار مختلف را انتخاب و هر کدام را به تنهایی گسترش داده و جهت حل موارد گوناگون بهترین آنها را برگزیده و کار تمام است. ولیکن طبق نظر متخصص هوش مصنوعی آقای «Mengye Ren» از دانشگاه نیویورک این روش با در نظر گرفتن میلیون ها ساختار مختلف زمان زیادی را به خود اختصاص داده و عملاً شاید غیر ممکن باشد تا بتوان نتیجه مطلوب را بدست آورد و به اصطلاح کل سیستم را تمرین داد.

لذا با در نظر گرفتن چالش های نامبرده آقای «Ren» با همکاری آقای «Chris Zhang» از دانشگاه تورنتو و خانم «Raquel Urtasun» همکار قدیمی خود کاملاً روش نوظهور جدیدی را برگزیدند. آنها اقدام به طراحی الگوریتم جدیدی با نام «Graphic Hyper Network» یا «ابر شبکه ترسیمی» به اختصار شبکه «GHN» نمودند، که وظیفه آن یافتن بهترین ساختار ممکن برای شبکه های زیر بنایی و حل چالش های تخصصی تعریف شده بود، نمودند. الگوریتم جدید این امکان را فراهم کرد که از میان میلیون ها طراحی مختلف فقط چند روش که جهت حل چالش های مشخص بهترین بودند انتخاب می شدند. ولیکن این پرسش هنوز وجود داشت که میان این چند روش کدام یک از آنها از حیطة آزمایش به گونه ای واقعی سربلند بیرون می آیند.

یافتن بهترین کاندیدا

همان طور که می دانیم این امکان وجود دارد که ساختار شبکه های مصنوعی را از طریق روش های ریاضی به صورت جداول و منحنی های چند بعدی ترسیم نمود. تیم محققین روش جدیدی را به مرحله آزمایش گذاردند، به نحوی که آنها ابتدا با اتکا به ریاضیات چند بعدی مجموعه ای از نقاط و زوایا را تعریف کردند: هر نقطه شامل یک لایه از برنامه را تشکیل و هر زاویه نحوه به هم پیوستن آنها را توجیح می کرد.

ایده بکار بسته بر این اساس استوار بود که میان چند روش موجود (یا به عبارت دیگر کاندیدا ها) یکی از آنها را انتخاب و آن را تبدیل به یک دیاگرام چند بعدی نمود، سپس از یک برنامه پیشرفته هوش مصنوعی خواسته می شود تا در ارتباط با مسئله موجود بهترین جا نمایی نقاط و زوایا را پیشگویی کرده و دیاگرام را به نحوی تغییر و بهبود بخشیده تا نتیجه آن بهترین و واقعی ترین پاسخ ممکن برای سؤال مطرح شده باشد. این عملکرد برای همه کاندیدا ها به اجرا در آمده و بازده همه آنها از طریق برنامه نویسی وارد اسکلت بندی و شبکه کل برنامه هوش مصنوعی شده و نهایتاً مقایسه می گردد که کدام کاندیدا کامل ترین و بهترین پاسخ را ارائه می دهد.



شیب نزولی «Gradient Descent»

هنگامی که میان داده های ورودی یک شبکه هوش مصنوعی و پاسخ های خروجی تفاوت زیادی موجود باشد، می توان آن را از طریق یک دیاگرام سه بعدی و توجیح آن به عنوان بالا ترین و یا کم ترین اشتباه ترسیم نمود. (قرمز: بالا ترین اشتباه و (آبی: کمترین اشتباه) حال در این فاز به یک برنامه جداگانه پیشرفته هوش مصنوعی فرمان داده می شود تا با مقایسه و تحلیل، میان بالاترین و پائین ترین ارقام، گزینه کمترین اشتباه را یافته و در اولویت قرار دهد.

حال در این فاز این سؤال مطرح می شود که ریزه کاری های این فرایند چگونه است؟ همان طور که در بالا کوتاه به آن اشاره شد، به اصطلاح کاندیدا ها (چند واحد از بهترین آلوگوریتم یا روش های انتخاب شده) که به صورت دیاگرام وانمود شده اند، همگی خود مجموعه ای از نقاط هستند و در این میان نقطه های تعریف شده خود دسته ای از اطلاعات هستند که ما آنها را در زبان برنامه هوش مصنوعی نوروں های مصنوعی خطاب می کنیم که دارای قابلیت محاسبه نیز می باشند. از جانب دیگر زوایا به ما نشان می دهند که چگونه نقاط یا به عبارت دیگر نوروں های مصنوعی با یکدیگر، جهت تبادل اطلاعات از شروع تا انتها در تماس هستند.

در فاز بعدی با کمک یک شبکه هوش مصنوعی جداگانه کلیه نقاط و زوایا و اطلاعات ذخیره شده را با پاسخ ایده ال خروجی که انتظار آن را داریم مقایسه نموده و از شبکه کمکی می خواهیم که بر حسب تشابه و یا عدم تشابه اطلاعات موجود در هر نقطه در قیاس با بهترین پاسخ، دیگرانی را به صورت سه بعدی پس از مقایسه ترسیم و نمایش دهد. در این دیاگرام بهترین جواب در قعر یک گودال قرار داشته (کمترین فاصله از نقطه صفر) و بدترین اشتباه در قله یک برآمدگی (بالاترین فاصله از نقطه صفر) قرار دارد. حال در این فاز برنامه نویسان بهترین جواب را به عنوان پاسخ نهایی و بسیار نزدیک به واقعیت اقتباس و تعریف نموده و آن را وارد برنامه در حال تکوین هوش مصنوعی می کنند که همان ابر برنامه جدید است. این فرایند برای کلیه نوروں ها تعریف شده و نتیجه آن دریافت بهترین پاسخ خروجی ممکن می باشد. به اختصار می توان گفت که روند تکمیل برنامه جدید با کمک دیاگرامها و نمادها از قله به قعر با (شیب نزولی) انجام یافته است. بدیهی است که می توان متصور شد که برنامه کمکی و تحلیل کننده نیز باید از پیش تمرین داده شود. چه بسا توروں ها و شبکه های محاسباتی چنین برنامه ای نیز مستلزم شناخت بهترین داده ها و تفکیک آنها از بدترین داده ها بوده و فراتر از آن اجزاء برنامه باید توان تفکیک و بررسی گروه بندی صحیح نقاط و زوایای نامیده را داشته باشند. جهت دستیابی به چنین قابلیت برنامه نویسان احتیاج به اطلاعاتی دارند که بتوانند مطابق آن برنامه کمکی را تمرین دهند (داده های آماده کننده). در ادامه بهترین داده ها و پاسخ های خروجی همان طور که اشاره شد، در بدنه "ابر نرم افزار" مورد نظر از طریق برنامه نویسی وارد و آن را آزمایش نمود. جهت آزمایش به عنوان مثال تصویری را جهت شناخت وارد ابر نرم افزار کرده و بررسی می کنند که نتیجه تا چه درجه ای با واقعیت تطبیق داشته و یا احتمالاً کماکان باید تکمیل برنامه را ادامه داد. حال در صورت احتیاج به ادامه تکمیل نرم افزار در حالتی که اشتباهی در خروجی و تشخیص تصاویر دیده شود (به عنوان مثال تفکیک یک گربه از سگ)، آنگاه برنامه نویسان میانگین اشتباه را محاسبه نموده و بر اساس آن سعی در بهبود نتایج خروجی می نمایند. در چنین فازی دو راه وجود دارد: اولی امکان تغییر در اسکلت بندی نقاط و زوایا بوده و دومی تغییر و بهبود شبکه های لایه های زیرین ابر نرم افزار در شناخت صحیح تصویر می باشد. متخصصین بر خلاف آنچه انتظار می رفت راه دوم را برگزیده و به مراتب نتایج خروجی صحیح تری را بدست آوردند. این روند همچنان برای پرسش های تصویری یا انواع دیگر و کاندیدا های گوناگون ادامه یافته تا پاسخ ها کمترین اشتباه ممکن را نشان دهند، تا جایی که عملاً امکان بهبود بیشتر خروجی ها از هر نوعی وجود نداشته باشد. بدیهی است که این گونه فعالیت های برنامه نویسی و تمرین دهی های پروگرام ها زمان زیادی را به خود اختصاص می دهند. ولیکن باید توجه داشت که چنین عملی طولانی مدت کافی است که یک بار انجام شود و می توان از نتایج آن برای کلیه داده های آتی بهره مند شد.

رقابت دو گروه برنامه نویس با یک هدف

هنگامی که گروه محققین آقای «Knyazev» متوجه پیشرفت های برنامه نویسی هوش مصنوعی توسط گروه آقای «Ren» با برخورداری و استفاده از روش دیاگرام های سه بعدی شدند، به این نتیجه رسیدند که می توان به مراتب چنین روشی را باز هم بهبود بخشید. در نرم افزار جدیدی که آنها توسعه دادند به اثبات رساندند که نه تنها می توان با توسل به انرم افزار های "ابر شبکه ترسیمی یا GHN" که در بالا به آن اشاره شد ساختار بهترین کاندیدا ها را به دست آورد، بلکه می توان مولفه و آلوگوریتم و ساختار بهترین شبکه های مصنوعی برای حل مسائل را نیز پیش گویی و دست چین نمود. بدیهی است که چنین عملی پیشاپیش، محتاج تمرین دهی سیستم می باشد. دلیل اتخاذ چنین تصمیمی در آن نهفته بود که نتایج به دست آمده در خصوص انتخاب بهترین ابر شبکه های ترسیمی که توسط گروه «Ren» به انجام رسیده بود، محدودیت دسترسی به صحیح ترین وردها (متغیرها) و داده های شبکه های مصنوعی را در بر داشت. گرچه نتایج گروه رقیب دارای خصوصیات خوب و قابل قبولی بودند، ولیکن نمی توان آنها را به عنوان بهترین راه کار ممکن تعریف نمود و این امکان وجود داشت تا نتایج خروجی آن را با برخورداری از آلوگوریتم های کاملتر تا درجات بالاتری بهبود داده و به واقعیت عینی نزدیک کرد. بدیهی است که نباید از نظر دور داشت که حتی روش های جدید انتخابی توسط گروه «Knyazev» نیز به نوبه خود با در نظر گرفتن طبیعت این نوع برنامه نویسی های پیچیده قادر نخواهد بود بهترین داده ها و پاسخ ها را ارائه دهد. به گفته ای دیگر می توان چنین تفسیر کرد که پیشرفته ترین برنامه های هوش مصنوعی در قیاس با ذهنیت انسان حد و مرز خود را داشته و به عنوان ابزاری بسیار خوب در خدمت انسان قرار دارد و هرگز در آینده نیز قابلیت رقابت با عملکرد مغز انسان را نخواهد داشت.

کار گروه آقای «Knyazev» به واقع قابل تقدیر است، چه بسا آنها موفق شدند تا سرحد ممکن پاسخ های ابر نرم افزار خود را به واقعیت نزدیکتر کنند و این امکان را به وجود آورند که جهت تمرین دهی و برنامه نویسی به مراتب نیاز به داده های کمتری باشد. این بدین معنی است که برنامه نویسان در فرایند تکامل نرم افزار محتاج داده های اولیه کمتری بوده و در کنار آن سرعت عمل را ارتقاء داده و نیاز به زمان کمتری جهت تکمیل کار دارند، که در امتداد به نوبه خود منجر به کاهش چشمگیر حجم محاسباتی سخت افزارها می گردد.

حال در این بازه می پردازیم به روشی که گروه آقای «Knyazev» جهت عملکرد خود انتخاب کردند: آنها ایده های اعلام شده گروه رقیب را ملاک قرار دادند و به دلیل منتشر نکردن «رمز نرم افزار- Source Code» توسط گروه آقای «Ren»، اقدام به برنامه نویسی مجدد کل ابر نرم افزار توسط گروه خود نمودند. در قدم بعدی تیم گروه آقای «Knyazev» اقدام به تکمیل و بهبود نرم افزار نموده و نام آن را به اختصار به "ابر شبکه ترسیمی شماره ۲ یا GHN-2" تغییر دادند. فراتر از آن گروه «Knyazev» موفق شدند تا از میان هزاران دیاگرام ترسیمی تعداد ۱۵ نقطه اساسی که قابلیت ترکیب شدن با یکدیگر را دارند و می توان از طریق آنها انواع شبکه های زیر بنایی مصنوعی را طراحی نمود، شناسایی کنند. قدم مهم دیگر آنها این بود که اقدام به برنامه ریزی نوعی الگوریتم بی نظیر جهت تمرین دهی ابر نرم افزار نمودند، به نحوی که نرم افزار "GHN-2" قابلیت را کسب کرد تا پارامترهای طراحی انواع گوناگون شبکه های مصنوعی را پیش گویی و تشخیص دهد. طبق گفته آقای «Knyazev» جهت تمرین دهی سیستم، ابتدا به صورت دلخواه اقدام به طراحی انواع گوناگون و متنوع ساختارهای شبکه مصنوعی نمودیم، به نحوی که این راه کار منجر شد تا نرم افزار "GHN-2" حتی قادر به تشخیص شبکه های ناشناخته شده و آنها را پیش بینی کند.

بدیهی است که هدف اصلی پس از انجام کلیه فعالیت های نامبرده در این راستا قرار داشت، تا نرم افزار "GHN-2" را آزمایش کرده و نتایج خروجی را در رابطه با ورودی های واقعی مقایسه نمود. در ادامه پس از تمرین دهی نرم افزار و ارتقاء آن در پیش بینی پارامترهای گوناگون مانند تشخیص صحیح تصاویر، آزمایش قابلیت دیگر نرم افزار مانند شناخت عوامل متغیر کاندیدا های مختلف در دستور کار قرار گرفت. چنین متغیرهایی می توانند از میلیون ها داده های ناشناخته تشکیل شوند که برنامه نویسان آنها را (داده مطرود یا Outliers) می نامند. در چنین فازی باید نرم افزار و قابلیت تشخیص آن را در رابطه با دو نوع داده، مانند داده های شناخته شده و داده های مطرود آزمایش نمود.

طراحی شبکه نورون های مصنوعی

جهت این که ابر نرم افزارها توان پیش بینی پارامترهای شبکه های عصبی مصنوعی را دارا باشند، ابتدا باید آنها را به نوعی دیاگرام سه بعدی ریاضیاتی تبدیل نمود. در این میان هر نقطه معادل یک واحد محاسباتی مانند یک لایه نورون های مصنوعی است. از جانب دیگر زوایا چگونگی تبادل اطلاعات میان واحدهای مختلف را نمایش می دهند. در پائین چند مثال از ساختار اینگونه شبکه های عصبی آورده شده که در آن رنگ های مختلف نمایانگر لایه های گوناگون می باشند.

شبکه های باز

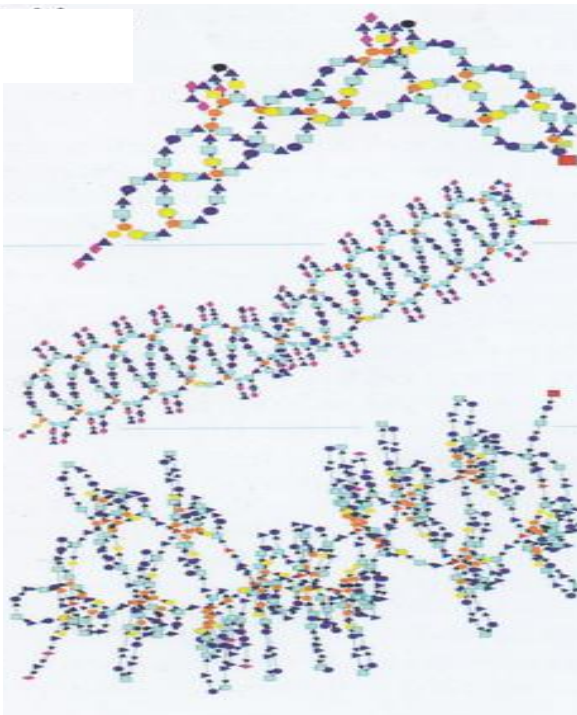
هر لایه متشکل از تعداد زیادی نورون است (نشان داده نشده)

شبکه های عمیق

داده ها میان تعداد زیادی از لایه ها در حرکتند

شبکه های مترابط

در اکثر لایه ها نورون ها با کلیه نورون های دیگر در همسایگی خود، در ارتباط هستند



در این فاز برنامه نویسان اقدام به طراحی داده های متغیر ۵۰۰ شبکه مطرود نموده و آنها را با الگوریتم های شناخته شده طبق فرایند « مسیر سیر نزولی – Stochastic gradient descent (SGD) » مقایسه کردند- به عنوان مثال شناخت تصاویر. در این راستا در ادامه، نتایج خروجی نرم افزار "GHN-2" با خروجی سایر برنامه های رقیب مقایسه شد. نتیجه آن بود که "GHN-2" از خود در بسیاری از موارد کار آیی به مراتب بهتری را به نمایش گزارد. میانگین عدد بازده شناسایی تصاویر در حیطه نرم افزار "GHN-2"، معادل ۶۰ الی ۶۶/۹ درصد بود که این مقدار در مقایسه با نرم افزار هایی که حجم آنها حدود ۲۰ برابر نرم افزار "GHN-2" است بسیار جالب توجه و از اهمیت ویژه ای برخوردار است که در حیطه کنفرانس "NeurIPS 2021" از مهمترین گرد همایی های علوم کامپیوتری به اطلاع عموم رسید.

میزان مصرف کمترین مقدار انرژی

جالب توجه است که این را هم بدانید که نرم افزار "GHN-2" در مقایسه با هایپر نرم افزار تصویری "ImageNet" از چنان بازده چشمگیری که در بالا به آن اشاره شد برخوردار نیست: به نحوی که میانگین بازده آن در شناخت تصاویر چیزی حدود ۲۷/۲ درصد را به خود اختصاص داد، در صورتی که نرم افزار های متداول معمولی توانی حدود ۲۵/۶ درصد را دارا بوده، که چنان از توان شناخت تصاویر توسط ابر نرم افزار "GHN-2" دور نیست. بدیهی است که از منظر برنامه نویسی و تمرین دهی سیستم های هوش مصنوعی این امکان وجود دارد تا مثلاً با استفاده از نرم افزار های ویژه تمرین دهی در قالب سیر نزولی، که قادر به طراحی دیاگرام های هستند که در ۵۰۰۰ تناوب پی در پی طراحی شده اند، توان ابر نرم افزار ها را تا رقم ۹۵ درصد افزایش داد. ولیکن با هزینه و صرف انرژی فراوان! ابر نرم افزار "GHN-2" قادر است تا تصاویر را در عرض یک ثانیه شناسایی کند که همین رقم در مقایسه با "GHN-2" برای نرم افزار های ۵۰۰۰ تناوبی، چیزی حدود ۱۰۰۰۰ بار بیشتر، یعنی حدود چندین ساعت است. فراتر از آن جهت راه اندازی سیستم های سریع تر با راندمان خوب، از منظر سخت افزاری احتیاج به کارت های گرافیک نسبتاً بزرگ و مقادیر فراوانی انرژی الکتریکی می باشد. متضاد آن یک سیستم "GHN-2" قادر است تا با کمک یک پردازش کننده معمولی نتایج چشمگیری را که در بالا به آن اشاره شد ارائه داده و مصرف انرژی را به مراتب تنزل دهیم. به اختصار می توان گفت که اکنون با در اختیار داشتن ابر نرم افزار هایی مانند "GHN-2" قادر هستیم تا با صرف هزینه و انرژی پائین، دانش یک سیستم نرم افزاری را به دیگری منتقل و حتی آن را گسترش نیز داد و آنچه قابل اهمیت است، هزینه نازل آن و استفاده همگانی از ظرفیت های کنونی و آینده هوش مصنوعی می باشد.