به نام خدا

یادگیری عمیق و شبکه های عصبی
به زبان ساده :
مفاهیم، چالش ها و پیاده سازی

مولف:

میترا نوری

انتشارات ارسطو

(سازمان چاپ و نشر ایران – ۱۴۰۴)

نسخه الکترونیکی این اثر در سایت سازمان چاپ و نشر ایران و اپلیکیشن کتاب رسان موجود می باشد

chaponashr.ir

سرشناسه : نوری، میترا، ۱۳۶۵-
عنوان و نام پدیدآور : یادگیری عمیق و شبکه‌های عصبی به زبان ساده: مفاهیم، چالش‌ها و پیاده‌سازی/ مولف میترا نوری.
مشخصات نشر: انتشارات ارسطو (سازمان چاپ و نشر ایران)، ۱۴۰۴.
مشخصات ظاهری : ۲۲۵ ص.
شابک : ۹۷۸-۶۲۲-۴۵۵-۹۴۷-۰ :
وضعیت فهرست نویسی : فیپا
موضوع : یادگیری عمیق (فراگیری ماشینی) Deep learning (Machine learning)
شبکه‌های عصبی (کامپیوتر) Neural networks (Computer science)
رده بندی کنگره : Q۳۲۵/۷۳
رده بندی دیویی : ۰۰۶/۳۱
شماره کتابشناسی ملی : ۱۰۱۴۳۵۴۱
اطلاعات رکورد کتابشناسی : فیپا

نام کتاب : یادگیری عمیق و شبکه های عصبی به زبان ساده: مفاهیم، چالش ها و پیاده سازی
مولف : میترا نوری
ناشر : انتشارات ارسطو (سازمان چاپ و نشر ایران)
صفحه آرایی، تنظیم و طرح جلد: پروانه مهاجر
تیراژ : ۱۰۰۰ جلد
نوبت چاپ : اول – ۱۴۰۴
چاپ : زبرجد
قیمت : ۲۲۶۰۰۰ تومان
فروش نسخه الکترونیکی - کتاب‌رسان :
https://chaponashr.ir/ketabresan
شابک : ۹۷۸-۶۲۲-۴۵۵-۹۴۷-۰
تلفن مرکز پخش : ۰۹۱۲۰۲۳۹۲۵۵
www. chaponashr. ir

انتشارات ارسطو
چاپ و نشر ایران
Chaponashr.ir

پیشگفتار

در دهه‌های اخیر، هوش مصنوعی و یادگیری ماشین به یکی از ستون‌های تحول در صنایع، خدمات و پژوهش‌های علمی بدل شده‌اند. در میان شاخه‌های گوناگون این حوزه، یادگیری عمیق (Deep Learning) و شبکه‌های عصبی مصنوعی به‌واسطه توانایی بی‌نظیر خود در مدل‌سازی روابط پیچیده و کشف الگوهای پنهان از میان انبوه داده‌ها، نقشی کلیدی ایفا کرده‌اند.

این کتاب با عنوان «یادگیری عمیق و شبکه‌های عصبی به زبان ساده» با هدف آموزش مفاهیم بنیادی تا پیشرفته‌ی یادگیری عمیق به شیوه‌ای روان و گام‌به‌گام تألیف شده است. تلاش کرده‌ایم با بهره‌گیری از زبان ساده، نمودارهای گویا، مثال‌های کاربردی، و نمونه‌کدهای عملی در پلتفرم‌هایی نظیر TensorFlow و PyTorch، پلی میان مفاهیم نظری و پیاده‌سازی عملی ایجاد کنیم.

ویژگی شاخص این کتاب، ترکیب آموزش مفهومی، تحلیل ریاضی، و پیاده‌سازی کد است؛ به‌گونه‌ای که هم دانشجویان تازه‌وارد و هم پژوهشگران میان‌رده و حتی توسعه‌دهندگان صنعتی بتوانند از آن بهره ببرند. ساختار کتاب به گونه‌ای طراحی شده که هر فصل با بیان اصول آغاز می‌شود، با مثال‌های واقعی و کدهای عملی تکمیل می‌گردد و با خلاصه مفهومی و تمرین‌های کاربردی پایان می‌پذیرد.

این کتاب حاصل تجربیات و مطالعات چندساله نویسندگان در حوزه یادگیری ماشین، تدریس دانشگاهی، و پیاده‌سازی پروژه‌های عملی است. امیدواریم که بتواند در مسیر ارتقای دانش، آموزش و تحقیق در حوزه یادگیری عمیق در کشور، سهمی هرچند کوچک ایفا کند.

از اساتید محترم، دانشجویان، و فعالان حوزه فناوری اطلاعات دعوت می‌کنیم با ارائه نظرات، انتقادات و پیشنهادهای خود، ما را در مسیر بهبود نسخه‌های بعدی این کتاب یاری دهند.

فهرست مطالب

فصل اول: مقدمه‌ای بر یادگیری عمیق و شبکه‌های عصبی ۱۵

۱-۱- مقدمه ۱۶

۱-۲- یادگیری عمیق چیست؟ ۱۶

۱-۳- تفاوت یادگیری عمیق با یادگیری ماشین سنتی ۱۷

۱-۳-۱- مهندسی ویژگی‌ها ۱۷

۱-۳-۲- عملکرد در داده‌های بزرگ و غیرخطی ۱۸

۱-۳-۳- نیاز به حافظه و محاسبات ۱۸

۱-۴- تاریخچه و انگیزه‌های ظهور یادگیری عمیق ۱۹

۱-۵- عوامل محبوبیت یادگیری عمیق ۲۲

۱-۵-۱- عملکرد فوق‌العاده در مدیریت داده‌های بزرگ ۲۲

۱-۵-۲- پیشرفت‌های سخت‌افزاری ۲۲

۱-۵-۳- حذف نیاز به استخراج دستی ویژگی‌ها ۲۳

۱-۵-۴- دستاوردهای عملی و موفقیت‌های چشم‌گیر ۲۳

۱-۵-۵- هم‌افزایی عوامل متعدد ۲۳

۱-۶- مزایای یادگیری عمیق ۲۳

۱-۶-۱- مدیریت حجم زیاد داده‌ها ۲۴

۱-۶-۲- پردازش داده‌های بدون ساختار ۲۴

۱-۶-۳- عدم نیاز به استخراج دستی ویژگی‌ها ۲۴

۱-۶-۴- دقت بالا در پیش‌بینی ۲۴

۱-۷- چالش‌ها و محدودیت‌های یادگیری عمیق ۲۴

۱-۷-۱- برازش بیش‌ازحد ۲۴

۱-۷-۲- تفسیرپذیری پایین ۲۵

۱-۷-۳- نیاز به منابع سخت‌افزاری قدرتمند ۲۵

۱-۷-۴- نیاز به داده‌های بسیار زیاد ۲۵

۱-۸- کاربردهای یادگیری عمیق ۲۵

۱-۸-۱- بینایی ماشین ۲۵

۱-۸-۲- پردازش زبان طبیعی ۲۶

۱-۸-۳- پردازش گفتار و صوت ۲۶
۱-۸-۴- یادگیری تقویتی و تصمیم‌گیری هوشمند ۲۶
۱-۸-۵- پزشکی و سلامت دیجیتال ۲۶
۱-۸-۶- سامانه‌های خودران و رباتیک پیشرفته ۲۷
۱-۸-۷- تولید محتوای خلاقانه و هنر دیجیتال ۲۷
۱-۸-۸- اقتصاد، امنیت و صنعت ۲۷
۱-۹- خلاصه فصل اول ۲۷
۱-۱۰- تمرین‌های فصل ۱ ..... ۲۸

فصل دوم: اصول و مفاهیم شبکه‌های عصبی و یادگیری عمیق ۲۹
۲-۱- مقدمه ۳۰
۲-۲- ساختار و عملکرد شبکه عصبی عمیق ۳۰
۲-۲-۱- ساختار نورون مصنوعی ۳۱
۲-۲-۲- مقایسه نورون طبیعی و نورون مصنوعی ۳۱
۲-۲-۳- رابطه ریاضی نورورن مصنوعی ۳۳
۲-۳- وزن‌ها، بایاس و تابع فعال‌سازی ۳۴
۲-۳-۱- وزن‌ها در شبکه‌های عصبی ۳۴
۲-۳-۲- چگونگی تنظیم وزن‌ها ۳۵
۲-۳-۳- بهینه‌سازی و اهمیت نرخ یادگیری ۳۶
۲-۳-۴- تابع فعال سازی ۳۶
۲-۳-۴-۱- افزودن غیرخطی‌بودن به نورون مصنوعی ۳۸
۲-۳-۴-۲- مزیت استفاده از توابع فعال‌سازی ۳۹
۲-۳-۴-۳- انواع تابع فعال‌سازی ۳۹
۲-۴- مدل ریاضی نورون مصنوعی با تابع فعالساز ۴۰
۲-۴-۱- بایاس در نورون مصنوعی ۴۰
۲-۴-۲- نقش حیاتی بایاس در یادگیری عمیق ۴۱
۲-۵- مفهوم تنسور در یادگیری عمیق ۴۲
۲-۵-۱- انواع تنسورها ۴۲
۲-۵-۲- علل اهمیت تنسورها ۴۲
۲-۶- مفهوم گراف محاسباتی در یادگیری عمیق ۴۳
۲-۶-۱- علل اهمیت گراف محاسباتی در یادگیری عمیق ۴۳

۲-۶-۲- انواع گراف‌ها ۴۳
۲-۷- فرآیند یادگیری نورون مصنوعی ۴۴
۲-۷-۱- هدف یادگیری ۴۴
۲-۷-۲- تابع هزینه ۴۵
۲-۸- ساختار و فرآیند یادگیری شبکه عصبی ۴۵
۲-۸-۱- مراحل آموزش شبکه عصبی ۴۶
۲-۸-۱-۱- آماده‌سازی داده‌ها ۴۶
۲-۸-۱-۲- تعریف معماری مدل ۴۶
۲-۸-۱-۳- انتخاب تابع هزینه ۴۶
۲-۸-۱-۴- انتخاب بهینه‌ساز ۴۷
۲-۸-۱-۵- حلقه آموزش ۴۷
۲-۸-۱-۶- ارزیابی عملکرد مدل ۴۷
۲-۸-۱-۷- ذخیره‌سازی و استقرار مدل ۴۸
۲-۸-۲- معرفی لایه‌های مختلف در شبکه عصبی مصنوعی ۴۸
۲-۸-۲-۱- لایه ورودی ۴۸
۲-۸-۲-۲- لایه‌های پنهان ۴۹
۲-۸-۲-۳- لایه اتصال کامل ۴۹
۲-۸-۲-۴-لایه خروجی ۴۹
۲-۸-۲-۵-لایه کانولوشن ۵۰
۲-۸-۲-۶- لایه زیرنمونه‌گیری ۵۰
۲-۸-۲-۷- لایه نرمال‌سازی ۵۰
۲-۸-۲-۸- لایه Dropout ۵۰
۲-۸-۲-۹- لایه‌های بازگشتی ۵۱
۲-۸-۳- یادگیری سلسله‌مراتبی ۵۱
۲-۸-۴- فرآیند یادگیری در شبکه عصبی ۵۱
۲-۸-۴-۱- انتشار رو به جلو ۵۲
۲-۸-۴-۲- انتشار رو به عقب ۵۲
۲-۸-۴-۳-نتیجه فرآیند یادگیری ۵۲
۲-۸-۵- معرفی مدل چندلایه‌ی پرسپترون ۵۶
۲-۸-۶- فریم‌ورک‌های یادگیری عمیق ۵۹
۲-۸-۶-۱- فریم‌ورک PyTorch ۵۹

۲-۸-۶-۲- فریم‌ورک TensorFlow ۶۰
۲-۸-۶-۳- رابط کاربری Keras ۶۰
۲-۸-۷- Google Colab ۶۲
۲-۹- خلاصه فصل دوم ۶۳
۲-۱۰- تمرین‌های فصل ۲ ..... ۶۴

فصل سوم: نقش توابع در یادگیری عمیق ۶۵
۳-۱- مقدمه ۶۶
۳-۱-۱- چرا توابع در یادگیری عمیق حیاتی‌اند؟ ۶۶
۳-۱-۲- تنوع و طبقه‌بندی توابع در شبکه‌های عصبی ۶۶
۳-۲- دسته‌بندی توابع در شبکه‌های عصبی و معیارهای انتخاب آن‌ها ۶۷
۳-۲-۱- دسته‌بندی اصلی توابع در یادگیری عمیق ۶۷
۳-۲-۲- معیارهای انتخاب توابع ۶۸
۳-۲-۳- نمودار ارتباط و جریان توابع در یک شبکه عصبی ۶۹
۳-۳- توابع فعال‌سازی در شبکه‌های عصبی ۶۹
۳-۳-۱- مقدمه‌ای بر نقش توابع فعال‌سازی ۶۹
۳-۳-۲- تابع فعال‌ساز خطی یا همانی ۷۰
۳-۳-۳- توابع فعال‌سازی غیر خطی ۷۱
۳-۳-۳-۱- تابع Sigmoid ۷۱
۳-۳-۳-۲- تابع تانژانت هیپربولیک ۷۲
۳-۳-۳-۳-تابع ReLU ۷۲
۳-۳-۳-۴- تابع Leaky ReLU ۷۳
۳-۳-۳-۵- تابع ELU ۷۳
۳-۳-۳-۶- تابع Swish ۷۴
۳-۳-۳-۷- تابع GELU ۷۴
۳-۳-۴- نمودار مقایسه‌ای توابع فعال‌سازی به‌همراه تحلیل ۷۵
۳-۳-۵- پیاده‌سازی توابع فعال‌سازی در Keras/ TensorFlow ۷۷
۳-۳-۶- انتخاب بهترین تابع فعال‌سازی: نکات و تجربیات ۷۸
۳-۳-۷- بررسی مشتقات توابع فعال‌سازی و نقش آن‌ها در جریان گرادیان ۷۸
۳-۳-۷-۱- چرا مشتق توابع فعال‌سازی مهم است؟ ۷۸
۳-۳-۷-۲- نمودار مشتقات توابع فعال‌سازی در PyTorch ۷۹

۳-۳-۸- انتخاب تابع فعال‌سازی مناسب بر اساس نوع معماری و داده ۸۰
۳-۳-۸-۱- انتخاب مناسب تابع فعال‌سازی بر اساس نوع معماری ۸۰
۳-۳-۸-۲- انتخاب تابع فعال‌سازی بر اساس نوع داده ۸۱
۳-۳-۹- خلاصه توابع فعالسازی ۸۴
۳-۴- توابع هزینه ۸۵
۳-۴-۱- مقدمه ۸۵
۳-۴-۲- طبقه‌بندی توابع هزینه بر اساس نوع مسئله ۸۶
۳-۴-۳- توابع هزینه رایج در مسائل رگرسیون ۸۷
۳-۴-۳-۱- تابع MSE ۸۷
۳-۴-۳-۱-۱- مزایا و معایب: ۸۸
۳-۴-۳-۱-۲- پیاده‌سازی MSE در TensorFlow / Kerasو PyTorch: ۸۸
۳-۴-۳-۱-۳- مشتق تابع MSE ۸۹
۳-۴-۳-۲- تابع هزینه MAE ۹۰
۳-۴-۳-۲-۱- مزایا و معایب ۹۰
۳-۴-۳-۲-۲- پیاده‌سازی MAEدر TensorFlow / Kerasو PyTorch: ۹۱
۳-۴-۳-۲-۳- مشتق تابع MAE ۹۲
۳-۴-۳-۳- تابع هزینه Huber ۹۲
۳-۴-۳-۳-۱- مزایا و معایب ۹۳
۳-۴-۳-۳-۲- پیاده‌سازی Huber در TensorFlow / Kerasو PyTorch ۹۳
۳-۴-۳-۳-۳- مشتق Huber ۹۴
۳-۴-۴- توابع‌هزینه‌رایج در مسائل‌طبقه‌بندی ۹۵
۳-۴-۴-۱- تابع آنتروپی متقاطع دودویی ۹۵
۳-۴-۴-۱-۱- مزایا و معایب BCE ۹۵
۳-۴-۴-۱-۲- پیاده‌سازی BCE در TensorFlow / Kerasو PyTorch ۹۷
۳-۴-۴-۲- تابع آنتروپی متقاطع چنددسته‌ای ۹۷
۳-۴-۴-۲-۱- مزایا و معایب CCE ۹۸
۳-۴-۴-۲-۲- پیاده‌سازی CCE در TensorFlow / Kerasو PyTorch: ۹۸
۳-۴-۴-۳- تابع آنتروپی متقاطع پراکنده برای طبقه‌بندی چندکلاسه ۹۹
۳-۴-۴-۳-۱- مزایا و معایب SCCE ۱۰۰
۳-۴-۴-۳-۲-پیاده‌سازی SCCE در TensorFlow / Kerasو PyTorch: ۱۰۰
۳-۵- توابع بهینه‌سازی ۱۰۲

۳-۵-۱- مفاهیم پایه در بهینه‌سازی ۱۰۲
۳-۵-۲- بهینه‌سازهای مبتنی بر گرادیان کاهشی ۱۰۳
۳-۵-۲-۱- انواع گرادیان کاهشی ۱۰۴
۳-۵-۲-۱-۱- گرادیان کاهشی دسته‌ای ۱۰۴
۳-۵-۲-۱-۲- گرادیان کاهشی تصادفی ۱۰۴
۳-۵-۲-۱-۳-گرادیان کاهشی دسته‌ای کوچک ۱۰۴
۳-۵-۲-۱-۴- گرادیان کاهشی با مومنتوم ۱۰۵
۳-۵-۲-۲- مقایسه کلی بهینه‌سازهای مبتنی بر گرادیان کاهش ۱۰۷
۳-۵-۳- بهینه‌سازهای تطبیقی ۱۰۸
۳-۵-۳-۱- الگوریتم بهینه سازNAG ۱۰۸
۳-۵-۳-۱-۱-پیاده‌سازی الگوریتم NAG ۱۰۹
۳-۵-۳-۲- الگوریتم بهینه‌ساز Adagrad ۱۱۰
۳-۵-۳-۲-۱-پیاده‌سازی الگوریتم Adagrad ۱۱۱
۳-۵-۳-۳- الگوریتم بهینه‌ساز RMSprop ۱۱۱
۳-۵-۳-۳-۱- پیاده‌سازی الگوریتم RMSprop ۱۱۲
۳-۵-۳-۴- الگوریتم بهینه‌ساز Adam ۱۱۴
۳-۵-۳-۴-۱-پیاده‌سازی الگوریتم Adam ۱۱۵
۳-۵-۳-۵- مقایسه بهینه‌سازهای تطبیقی ۱۱۶
۳-۶- خلاصه فصل سوم ۱۱۷
۳-۷- تمرین‌های فصل ۳ ..... ۱۱۸

فصل چهارم: چالش‌های آموزش شبکه‌های عصبی و راهکارهای مقابله با آن ۱۱۹
۴-۱- مقدمه ۱۲۰
۴-۲- چالش‌های مرتبط با داده ۱۲۰
۴-۲-۱- داده ناکافی و راهکارهای داده‌محور ۱۲۰
۴-۲-۱-۱- راهکار اول افزایش مصنوعی داده‌ها ۱۲۱
۴-۲-۱-۲- راهکار دوم انتقال یادگیری ۱۲۲
۴-۲-۱-۳- راهکار سوم استفاده از داده مصنوعی ۱۲۳
۴-۲-۲- داده‌های نامتوازن ۱۲۴
۴-۲-۲-۱- راهکار اول وزن‌دهی نمونه‌ها در تابع هزینه ۱۲۴
۴-۲-۲-۲- راهکاردوم افزایش نمونه کلاس‌های اقلیت ۱۲۵

۴-۲-۲-۳- راهکار سوم کاهش نمونه کلاس اکثریت ۱۲۷
۴-۲-۲-۴- راهکار چهارم استفاده از متریک‌های مناسب ارزیابی ۱۲۷
۴-۲-۳- داده نویزی یا دارای برچسب اشتباه ۱۲۸
۴-۲-۳-۱- راهکار اول هموارسازی برچسب‌ها Label Smoothing ۱۲۹
۴-۲-۳-۲- راهکار دوم حذف یا شناسایی برچسب‌های مشکوک ۱۳۰
۴-۲-۳-۳- راهکار سوم استفاده از مدل‌های مقاوم در برابر نویز ۱۳۰
۴-۲-۴- بیش برازش روی داده آموزش ۱۳۰
۴-۲-۴-۱- راهکار اول کاهش پیچیدگی مدل ۱۳۱
۴-۲-۴-۲- راهکار دوم استفاده از Dropout ۱۳۱
۴-۲-۴-۳- راهکار سوم استفاده از Regularization (L2 Regularization) ۱۳۳
۴-۲-۴-۴- راهکار چهارم استفاده ازتوقف زودهنگام ۱۳۴
۴-۲-۴-۵- راهکار پنجم اعتبار سنجی مقابل ۱۳۶
۴-۲-۵- عدم تنوع داده‌ها ۱۳۸
۴-۲-۵-۱- راهکار اول جمع‌آوری داده‌های متنوع ۱۳۹
۴-۲-۵-۲- راهکاردوم شبیه‌سازی داده در شرایط مختلف ۱۳۹
۴-۲-۶- پیش‌پردازش ضعیف داده‌ها ۱۴۰
۴-۲-۶-۱- راهکار اول نرمال‌سازی ویژگی‌ها ۱۴۰
۴-۲-۶-۲- راهکار دوم کد گذاری صحیح ویژگی‌ها Encoding ۱۴۱
۴-۳- چالش‌های معماری و طراحی مدل ۱۴۳
۴-۳-۱- انتخاب نادرست اندازه یا عمق مدل ۱۴۳
۴-۳-۲- انتخاب نادرست نوع لایه‌ها ۱۴۴
۴-۳-۳- عدم استفاده از تکنیک‌های مدرن معماری ۱۴۴
۴-۳-۴- طراحی ضعیف تابع هزینه ۱۴۵
۴-۴- چالش‌های بهینه‌سازی در آموزش شبکه‌های عصبی ۱۴۶
۴-۴-۱- محوشدگی گرادیان‌ها ۱۴۶
۴-۴-۲- انفجار گرادیان‌ها ۱۴۷
۴-۴-۴- نرخ یادگیری نامناسب ۱۴۹
۴-۴-۵- گیر افتادن در مینیمم محلی ۱۴۹
۴-۴-۶- نوسان در تابع هزینه ۱۵۰
۴-۵- چالش مقداردهی اولیه وزن‌ها در شبکه‌های عصبی ۱۵۰
۴-۵-۱- پیامدهای مقداردهی نادرست وزن‌ها ۱۵۱

۴-۵-۲- راهکارهای اصولی برای مقداردهی اولیه ۱۵۳
۴-۶- خلاصه فصل چهارم ۱۵۵
۴-۷-تمرین‌های فصل ۴ ..... ۱۵۵

فصل پنجم: شبکه‌ی عصبی کانولوشنی ۱۵۷
۵-۱- مقدمه ۱۵۸
۵-۱-۱-تعریف کانولوشن به زبان ساده ۱۵۸
۵-۱-۲- کاربرد کانولوشن در بینایی ماشین و پردازش تصویر ۱۵۹
۵-۱-۳-ارتباط کانولوشن با فیلتر ۱۶۰
۵-۲- مفهوم ریاضی کانولوشن ۱۶۱
۵-۲-۱- تعریف ریاضی عملیات کانولوشن در سیگنال‌های ۱بعدی ۱۶۱
۵-۲-۲-گسترش به تصاویر دوبعدی ۱۶۲
۵-۳- اجزای اصلی کانولوشن ۱۶۸
۵-۳-۱- فیلتر ۱۶۹
۵-۳-۲- گام ۱۶۹
۵-۳-۳- پرکننده Padding ۱۷۰
۵-۳-۴- نقشه ویژگی ۱۷۰
۵-۳-۵- خروجی عملیات کانولوشن ۱۷۱
۵-۴- نحوه اعمال کانولوشن در تصاویر رنگی (RGB): ۱۷۲
۵-۴-۱- فرآیند اعمال کانولوشن در تصاویر رنگی ۱۷۳
۵-۴-۲- مفهوم فیلتر در تصاویر رنگی ۱۷۳
۵-۴-۳- پیاده‌سازی کانولوشن روی تصویر رنگی ۱۷۴
۵-۵- ویژگی‌های قابل استخراج با کانولوشن ۱۷۷
۵-۵-۱- تشخیص لبه‌ها ۱۷۷
۵-۵-۲- تشخیص بافت ۱۷۷
۵-۵-۳- الگوهای تکرارشونده ۱۷۷
۵-۶- مزایای کانولوشن در یادگیری عمیق ۱۷۸
۵-۶-۱- کاهش تعداد پارامترها و منابع محاسباتی ۱۷۸
۵-۶-۲- حفظ ساختار مکانی و سلسله‌مراتبی تصویر ۱۷۸
۵-۶-۳- انتقال‌پذیری ویژگی‌ها ۱۷۹
۵-۷- پیاده‌سازی کانولوشن در NumPy, TensorFlow/Keras , PyTorch ۱۷۹

۵-۷-۱- پیاده‌سازی کانولوشن با NumPy ۱۸۰
۵-۷-۲- پیاده‌سازی کانولوشن در TensorFlow/Keras ۱۸۱
۵-۷-۳- پیاده‌سازی کانولوشن در PyTorch ۱۸۴
۵-۸- خلاصه فصل پنجم ۱۸۶
۵-۹- تمرین‌های فصل ۵ ۱۸۶

فصل ششم: شبکه‌ی عصبی بازگشتی ۱۸۸
۶-۱ مقدمه ۱۸۹
۶-۲- داده‌های زمانی چیست؟ ۱۸۹
۶-۲-۱- نمونه‌هایی از داده‌های زمانی در حوزه‌های مختلف ۱۹۰
۶-۲-۲- ویژگی‌های کلیدی داده‌های زمانی ۱۹۰
۶-۳- شبکه‌های عصبی بازگشتی ۱۹۱
۶-۳-۱- ساختار محاسباتی ساده در شبکه‌های عصبی بازگشتی ۱۹۲
۶-۳-۲- پیاده‌سازی RNN ساده در PyTorch, Keras / TensorFlow ۱۹۳
۶-۳-۳- چالش‌های آموزش شبکه‌های عصبی بازگشتی ۱۹۴
۶-۳-۳-۱- محوشدگی گرادیان ۱۹۴
۶-۳-۳-۲- انفجار گرادیان ۱۹۴
۶-۳-۴- راهکارهای مقابله با چالش‌های شبکه‌های عصبی بازگشتی ۱۹۴
۶-۳-۴-۱- استفاده از تکنیک‌های بهبوددهنده ۱۹۵
۶-۳-۴-۱-۱- نرمال‌سازی گرادیان یا قطع گرادیان و پیاده سازی آن ۱۹۵
۶-۳-۴-۱-۲- اعمال Dropout در شبکه‌های بازگشتی و پیاده سازی آن ۱۹۷
۶-۳-۴-۲- استفاده از معماری پیشرفته RNN دوطرفه و LSTM و GRU ۲۰۰
۶-۳-۴-۲-۱- شبکه عصبی بازگشتی دوطرفه و پیاده سازی آن ۲۰۱
۶-۴- حافظه طولانی‌مدت در شبکه‌های عصبی بازگشتی LSTM ۲۰۳
۶-۴-۱- ویژگی‌های مهم LSTM ۲۰۴
۶-۴-۲- پیاده‌سازی ساده LSTM در PyTorch, TensorFlow/Keras ۲۰۴
۶-۵- واحدهای دروازه‌دار بازگشتی در شبکه‌های عصبی GRU ۲۰۶
۶-۵-۱- ویژگی‌های مهم GRU ۲۰۶
۶-۵-۲- تفاوت‌های کلیدی بین LSTM و GRU ۲۰۷
۶-۵-۳-پیاده‌سازی GRU در PyTorch, TensorFlow/Keras ۲۰۷
۶-۶- خلاصه فصل ششم ۲۰۹

۶-۷-تمرین‌های فصل ۶ ..... ۲۱۰

فصل هفتم: شبکه‌های مولد تخاصمی ۲۱۲

۷-۱- مقدمه ۲۱۳

۷-۲- مفاهیم اولیه و ساختار GAN ۲۱۴

۷-۳- فرایند آموزش شبکه‌های مولد تخاصمی ۲۱۶

۷-۴- چالش‌های GAN ۲۱۶

۷-۴-۱- ناپایداری آموزش ۲۱۷

۷-۴-۲- فرورفتن در مد ۲۱۷

۷-۴-۳- همگرایی ناقص ۲۱۸

۷-۵- کاربردهای GAN ۲۱۹

۷-۶- پیاده سازی GAN برای مجموعه داده MNIST ۲۲۰

۷-۷- خلاصه فصل هفتم ۲۲۳

۷-۸- تمرین‌های فصل ۷ ..... ۲۲۳

فصل اول: مقدمه‌ای بر یادگیری عمیق و شبکه‌های عصبی

۱-۱- مقدمه

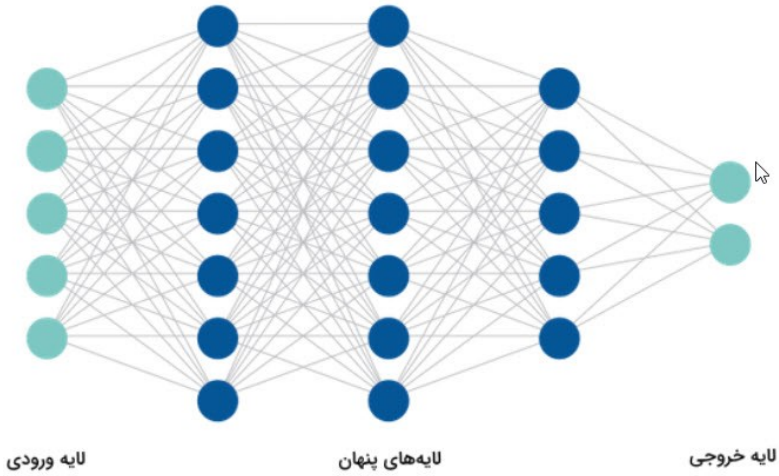
هوش مصنوعی[1] به مجموعه‌ای از تکنیک‌ها و الگوریتم‌ها اطلاق می‌شود که به سیستم‌های کامپیوتری اجازه می‌دهد رفتارهایی هوشمندانه از خود نشان دهند. یادگیری ماشین[2] و یادگیری عمیق[3] دو زیرشاخه اصلی هوش مصنوعی هستند. یادگیری ماشین به سیستم‌ها این توانایی را می‌دهد که بدون برنامه‌نویسی صریح، از داده‌ها بیاموزند و عملکرد خود را بهبود بخشند. یادگیری عمیق به‌عنوان شاخه‌ای پیشرفته‌تر از یادگیری ماشین، با الهام از ساختار مغز انسان، از شبکه‌های عصبی مصنوعی[4] برای یادگیری از داده‌های پیچیده و غیرساختار یافته استفاده می‌کند.

۱-۲- یادگیری عمیق چیست؟

یادگیری عمیق شاخه‌ای پیشرفته از یادگیری ماشین است که بر ساخت مدل‌هایی مبتنی بر معماری‌های چندلایه تمرکز دارد. این مدل‌ها با بهره‌گیری از لایه‌های پنهان متعدد قادرند داده‌های خام را تحلیل کرده و الگوهای بسیار پیچیده‌ای را استخراج کنند.

در واقع، یادگیری عمیق به مدل‌ها این توانایی را می‌دهد که ویژگی‌های مهم داده‌ها را بدون نیاز به طراحی دستی، به‌صورت خودکار کشف کنند. این امر، به‌ویژه در مواجهه با داده‌های عظیم و متنوع، اهمیت فوق‌العاده‌ای پیدا می‌کند. از جمله ویژگی‌های بارز یادگیری عمیق علاوه بر یادگیری خودکار ویژگی‌ها ،مقیاس پذیری بالا و بهره گیری مؤثر از منابع محاسباتی پیشرفته مانند GPU و TPU اشاره کرد.

[1] AI-Artificial intelligence
[2] Machine Learning
[3] Deep Learning
[4] Artificial neural networks



تصویر ۱- ساختار کلی شبکه‌های عمیق

۱-۳- تفاوت یادگیری عمیق با یادگیری ماشین سنتی

درک تفاوت میان یادگیری عمیق و یادگیری ماشین سنتی، نخستین گام برای فهم علت محبوبیت روزافزون یادگیری عمیق در دهه گذشته است. گرچه هر دو زیرمجموعه‌ای از هوش مصنوعی محسوب می‌شوند، تفاوت‌های اساسی در نحوه پردازش داده‌ها، نیاز به مهندسی ویژگی و عملکرد آن‌ها در مسائل پیچیده وجود دارد.

۱-۳-۱- مهندسی ویژگی‌ها[1]

در یادگیری ماشین سنتی، عملکرد مدل‌ها به‌شدت وابسته به مهندسی ویژگی‌ها توسط متخصصان انسانی است. برای مثال، در مسأله تشخیص چهره، باید ویژگی‌هایی مانند فاصله چشم‌ها، شکل بینی یا زاویه فک به‌صورت دستی استخراج شوند.

اما در یادگیری عمیق، به‌ویژه با استفاده از شبکه‌های عصبی کانولوشن[2]، مدل به‌صورت خودکار ویژگی‌های معنادار را از داده‌های خام استخراج می‌کند. این امر باعث کاهش وابستگی به تخصص انسانی و افزایش دقت مدل در مسائل پیچیده می‌شود.

[1] Feature Engineering
[2] CNN- Convolutional neural networks

۱-۳-۲- عملکرد در داده‌های بزرگ و غیرخطی

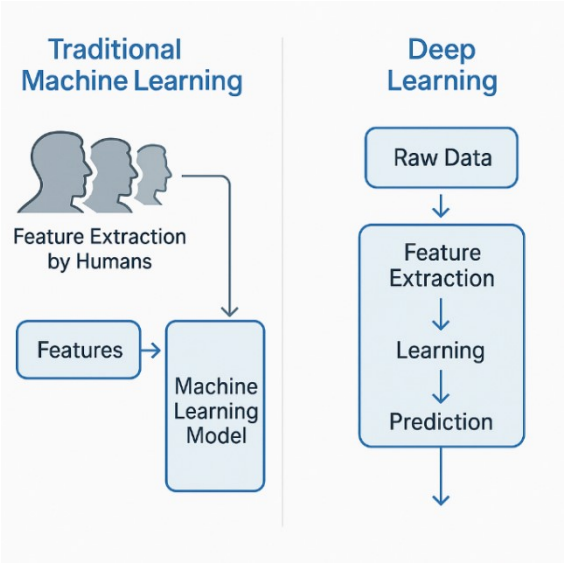
مدل‌های سنتی مانند درخت تصمیم[1] یا ماشین بردار پشتیبان[2] در مسائل خطی و مجموعه‌داده‌های کوچک عملکرد مناسبی دارند. اما با افزایش حجم داده و پیچیدگی الگوها، قدرت تعمیم‌دهی آن‌ها کاهش می‌یابد.

در مقابل، شبکه‌های عصبی عمیق با لایه‌های متعدد و قابلیت یادگیری نمایش‌های سلسله‌مراتبی، قادر به مدل‌سازی روابط بسیار غیرخطی هستند. این ویژگی موجب شده است تا یادگیری عمیق در پردازش تصویر، زبان طبیعی و بینایی ماشین عملکردی خیره‌کننده داشته باشد.

۱-۳-۳- نیاز به حافظه و محاسبات

مدل‌های سنتی نسبتاً سبک و قابل پیاده‌سازی روی رایانه‌های معمولی هستند، ولی شبکه‌های یادگیری عمیق به منابع محاسباتی بالا مانند (GPU) و حافظه بیشتر نیاز دارند.

با این حال، پیشرفت سخت‌افزارها و کتابخانه‌های مدرن مانند (TensorFlow) و (PyTorch) این محدودیت را تا حد زیادی برطرف کرده‌اند.



تصویر ۲- تفاوت یادگیری عمیق با یادگیری ماشین سنتی

[1] Decision Tree
[2] SVM- Support vector machine

در تصویر ۲، تفاوت پردازش داده در یادگیری ماشین سنتی و یادگیری عمیق نمایش داده شده است:

در سمت چپ و در مدل یادگیری ماشین، استخراج ویژگی‌ها توسط انسان صورت می‌پذیرد و ویژگی به مدل یادگیری ماشین داده می‌شود. در سمت راست تصویر، مدل یادگیری عمیق دیتای خام را به‌طور خودکار استخراج کرده و سپس مرحله یادگیری و پیش‌بینی انجام می‌شود.

بنابراین، تفاوت اصلی میان یادگیری عمیق و یادگیری ماشین سنتی در قابلیت خودکارسازی یادگیری ویژگی‌ها، توان مدل‌سازی روابط پیچیده و نیاز به داده و محاسبات بیشتر است. یادگیری عمیق برای مسائل حجیم، غیرخطی و بدون نیاز به طراحی دستی ویژگی‌ها، انتخابی برتر است.

۱-۴- تاریخچه و انگیزه‌های ظهور یادگیری عمیق

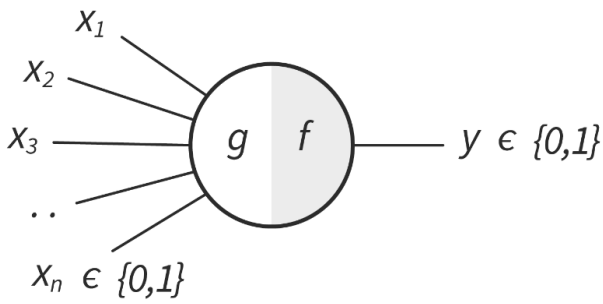
پیدایش حوزه یادگیری عمیق به دهه ۱۹۴۰ بازمی‌گردد. در آن زمان، به‌دلیل نقص‌های مختلفی که در مدل‌های ارائه‌شده وجود داشت، این حیطه از فناوری چندان مورد توجه قرار نگرفت. البته باید خاطرنشان کرد که همان مدل‌های اولیه، راه‌گشای پیشرفت در پژوهش‌های مدل‌های عمیق شدند و نقش مهمی را در شکل‌گیری مدل‌های قدرتمند امروزی ایفا کردند، به‌طوری که همچنان از برخی مدل‌های قدیمی در پیاده‌سازی مسائل امروزی استفاده می‌شود.

در سال‌های ۱۹۴۰ تا ۱۹۶۰، مطالعات اولیه‌ای پیرامون مدل‌های یادگیری عمیق شکل گرفت که ایده اولیه این پژوهش‌ها از نحوه یادگیری بیولوژیکی انسان الهام گرفته شده بود. این مطالعات با نام سایبرنتیک[1] شناخته می‌شدند و هدف آن‌ها ساخت مدل‌های محاسباتی بود. این مدل‌ها مشابه مغز انسان و حیوان، به یادگیری مسئله خاصی می‌پرداختند. تا به امروز نیز این شاخه از مطالعات تحت عنوان علوم اعصاب محاسباتی[2] ادامه دارد.

نخستین مدل محاسباتی توسط وارن موکولوچ[3] و والتر پیتس[4] در سال ۱۹۴۳ ارائه شد. این دو دو فرد به‌عنوان عصب‌شناس و منطق‌دان با استفاده از نحوه عملکرد نورون‌های مغز، به طراحی

[1] Cybernetics
[2] Computational Neuroscience
[3] Warren McCulloch
[4] Walter Pitts

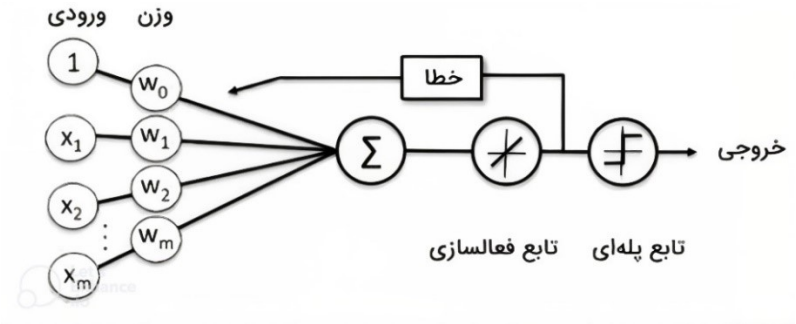
مدلی پرداختند که با دریافت ورودی و اعمال عمل جمع بر روی آن‌ها، مقداری را در خروجی باز می‌گرداند. وزن‌های مورد نیاز این شبکه به‌صورت دستی توسط انسان مشخص می‌شد.



تصویر ۳- نخستین مدل یادگیری عمیق

در اواخر دهه ۱۹۵۰، مدلی با عنوان پرسپترون[1] توسط روانشناس آمریکایی به نام «فرانک روزنبلات»[2] ارائه شد. ساختار این مدل مشابه با ساختار مدل ارائه‌شده توسط موکولوچ و پیتس بود، اما وزن‌های این شبکه عصبی به‌صورت خودکار توسط مدل یاد گرفته می‌شدند. یکی از کاربردهای اصلی مدل پرسپترون در آن زمان، تشخیص تصاویر و دسته‌بندی آن‌ها بود.

همزمان با ارائه مدل پرسپترون، مدل دیگری با عنوان «ادلاین»[3] توسط «برنارد ویدرو»[4] مطرح مطرح شد که یادگیری وزن‌های این مدل نیز همانند مدل پرسپترون به‌صورت خودکار بود. با این حال، بزرگ‌ترین نقصی که این مدل داشت، خطی بودن آن بود. به همین خاطر، این مدل نمی‌توانست برای پیاده‌سازی مسائل غیرخطی نظیر XOR کارایی داشته باشد.



تصویر ۴- مدل Adaline

[1] Perceptron
[2] Frank Rosenblat
[3] Adaline
[4] Bernard Widrow

در دهه ۱۹۸۰، مفهومی با عنوان پیوندگرایی[۱] یا پردازش توزیع‌شده موازی[۲] مطرح شد که برگرفته از علوم شناختی[۳] بود و بر ساخت شبکه‌های عصبی با ساختار لایه‌ای تأکید داشت که به نوعی از نحوه ارتباط نورون‌های مغز انسان الهام گرفته شده بود.

لایه‌های متصل به هم در این شبکه‌ها، امکان پردازش موازی ورودی‌ها را فراهم کردند. اتصال لایه‌ها با مجموعه‌ای از وزن‌ها صورت می‌گرفت که این وزن‌ها مشخص می‌کردند کدام ورودی می‌تواند در طول شبکه جریان داشته باشد. در این دوران، مدل‌ها و الگوریتم‌هایی مانند مدل «حافظه طولانی کوتاه‌مدت»[۴] و روش پس‌انتشار[۵] مطرح شدند که امروزه نیز کاربرد زیادی در پژوهش‌های مربوط به یادگیری عمیق دارند. از سال ۱۹۸۰ تا اوایل دهه ۲۰۰۰، به دلیل کمبود منابع محاسباتی، پژوهش‌های اندکی پیرامون یادگیری عمیق و ارائه مدل‌های جدید انجام شد و سرمایه‌گذاران حاضر نشدند در این دوران هزینه زیادی را در پیشرفت این مسیر صرف کنند. پیشرفت در دیپ لرنینگ در اوایل دهه ۲۰۰۰ اتفاق افتاد، زمانی که محققان الگوریتم‌های جدیدی را برای آموزش شبکه‌های عصبی عمیق توسعه دادند. یکی از رویدادهای کلیدی که به جرقه زدن علاقه فعلی به یادگیری عمیق کمک کرد، مسابقه ImageNet در سال ۲۰۱۲ بود که در آن یک شبکه عصبی عمیق به‌نام AlexNet نسبت به روش‌های قبلی به میزان دقت بسیار بالاتری دست یافت.

از آن زمان، یادگیری عمیق به یک تکنیک پرکاربرد در بسیاری از زمینه‌ها، از جمله بینایی کامپیوتر، پردازش زبان طبیعی و تشخیص گفتار تبدیل شده است. توسعه سخت‌افزارهای قدرتمند مانند پردازنده‌های گرافیکی و تراشه‌های تخصصی نیز در رشد دیپ لرنینگ مؤثر بوده است. در سال ۲۰۰۶، پژوهش‌های این حوزه از فناوری به‌طور جدی‌تر از سر گرفته شد و ارائه شبکه‌های باور عمیق[۶] توسط جفری هینتون[۷] سنگ‌بنایی برای ارائه مدل‌های یادگیری عمیق پیشرفته‌تر شد.

---

[1] Connectionism
[2] Parallel Distributed Processing
[3] Cognitive Science
[4] Short Long Term Memory (LSTM)
[5] Backpropagation
[6] Deep Belief Networks
[7] Geoffrey Hinton

۱-۵- عوامل محبوبیت یادگیری عمیق

تصویر ۵- عوامل محبوبیت یادگیری عمیق

در سال‌های اخیر، یادگیری عمیق به یکی از مهم‌ترین و محبوب‌ترین شاخه‌های هوش مصنوعی تبدیل شده است. این محبوبیت به دلایل متعددی بازمی‌گردد که در ادامه به مهم‌ترین آن‌ها اشاره می‌کنیم:

۱-۵-۱- عملکرد فوق‌العاده در مدیریت داده‌های بزرگ

یادگیری عمیق، به‌ویژه در دنیای داده‌محور امروزی، با قابلیت پردازش حجم عظیمی از داده‌ها، توانسته است عملکردی فراتر از الگوریتم‌های کلاسیک یادگیری ماشین از خود نشان دهد. برخلاف روش‌های سنتی که در برابر داده‌های پیچیده و بزرگ ناتوان‌اند، مدل‌های یادگیری عمیق با افزایش داده‌های آموزشی، عملکرد بهتری پیدا می‌کنند.

۱-۵-۲- پیشرفت‌های سخت‌افزاری

یکی از عوامل کلیدی موفقیت یادگیری عمیق، توسعه چشمگیر سخت‌افزارهای تخصصی مانند واحدهای پردازش گرافیکی (GPU) و واحدهای پردازش تنسوری (TPU) بوده است. این سخت‌افزارها با فراهم کردن توان پردازش بالا و امکان انجام عملیات موازی، باعث تسریع آموزش شبکه‌های عصبی عمیق شده‌اند.

۱-۵-۳- حذف نیاز به استخراج دستی ویژگی‌ها

در روش‌های یادگیری ماشین سنتی، بخش زیادی از زمان و تخصص صرف طراحی و استخراج دستی ویژگی‌ها می‌شد. این کار نیاز به متخصص داشت و معمولاً پیچیده و زمان‌بر بود. یادگیری عمیق با توانایی خود در یادگیری خودکار ویژگی‌ها از داده‌های خام، این فرآیند را به‌کلی متحول کرد و دقت مدل‌ها را نیز افزایش داد.

۱-۵-۴- دستاوردهای عملی و موفقیت‌های چشم‌گیر

از ترجمه ماشینی گرفته تا تشخیص چهره، از رانندگی خودکار تا تشخیص بیماری‌ها، مدل‌های یادگیری عمیق در بسیاری از حوزه‌های واقعی نتایج بسیار چشم‌گیری ارائه داده‌اند. این موفقیت‌ها موجب شده‌اند که اعتماد و تمایل صنعت و دانشگاه به این فناوری به‌طور بی‌سابقه‌ای افزایش یابد.

۱-۵-۵- هم‌افزایی عوامل متعدد

ترکیب عوامل زیرساختی، پژوهشی و اقتصادی موجب رشد انفجاری یادگیری عمیق شده‌اند:

دسترسی به مجموعه‌داده‌های بزرگ و عمومی مانند ImageNet و Common Voice، ابزارها و چارچوب‌های متن‌باز مانند TensorFlow، PyTorch و Keras، همکاری‌های بین‌رشته‌ای بین آمار، علوم رایانه، زبان‌شناسی، پزشکی و غیره، سرمایه‌گذاری گسترده شرکت‌های بزرگ فناوری و استارتاپ‌ها، افزایش آگاهی عمومی نسبت به کاربردهای AI. یادگیری عمیق نه تنها به دلیل نوآوری‌های فنی، بلکه به‌خاطر تلفیق هوشمندانه دانش، فناوری، زیرساخت و تقاضای بازار توانسته جایگاه برتری در چشم‌انداز هوش مصنوعی امروز به دست آورد.

۱-۶- مزایای یادگیری عمیق

یادگیری عمیق با وجود پیچیدگی‌های فنی خود، مزایای قابل توجهی برای پردازش داده‌ها، استخراج ویژگی‌ها و ساخت مدل‌های دقیق فراهم کرده است. در ادامه، مهم‌ترین مزایا و چالش‌های این رویکرد بررسی می‌شود.

۱-۶-۱- مدیریت حجم زیاد داده‌ها

الگوریتم‌های یادگیری عمیق به‌گونه‌ای طراحی شده‌اند که می‌توانند از مجموعه داده‌های بزرگ بیاموزند. این ویژگی آن‌ها را برای کاربردهایی همچون تشخیص تصویر و گفتار که نیازمند داده‌های عظیم هستند، بسیار مناسب می‌سازد.

۱-۶-۲- پردازش داده‌های بدون ساختار

برخلاف الگوریتم‌های سنتی که متکی بر داده‌های ساختاریافته هستند، الگوریتم‌های یادگیری عمیق قادرند از داده‌های بدون ساختار مانند تصاویر، صدا و زبان طبیعی نیز یاد بگیرند. این ویژگی باعث گسترش دامنه کاربرد آن‌ها در پردازش زبان طبیعی، بینایی ماشین و تحلیل گفتار شده است.

۱-۶-۳- عدم نیاز به استخراج دستی ویژگی‌ها

در روش‌های سنتی یادگیری ماشین، استخراج ویژگی‌ها نیازمند دانش تخصصی و زمان‌بر بود. یادگیری عمیق این فرآیند را به‌صورت خودکار انجام می‌دهد و ویژگی‌ها را متناسب با داده و هدف یادگیری به‌صورت سلسله‌مراتبی استخراج می‌کند.

۱-۶-۴- دقت بالا در پیش‌بینی

مدل‌های یادگیری عمیق به‌دلیل توانایی‌شان در شناسایی روابط پیچیده میان داده‌ها، معمولاً خروجی‌هایی با دقت بالا ارائه می‌دهند. در صورت آموزش مناسب با داده‌های کافی، این مدل‌ها می‌توانند در مدت‌زمان کوتاهی نتایج بسیار دقیقی تولید کنند.

۱-۷- چالش‌ها و محدودیت‌های یادگیری عمیق

۱-۷-۱- برازش بیش‌ازحد[1]

در برخی موارد، مدل‌های یادگیری عمیق به‌قدری بر داده‌های آموزشی متمرکز می‌شوند که در تعمیم به داده‌های جدید دچار مشکل می‌شوند. این پدیده باعث کاهش دقت مدل در شرایط واقعی می‌گردد.

[1] Overfitting