



**دانشگاه اصفهان**

**دانشکده مهندسی کامپیوتر**

**گروه هوش مصنوعی و رباتیک**

**پایان نامه کارشناسی ارشد رشته‌‌ی مهندسی کامپیوتر**

**گرایش هوش مصنوعی و رباتیک**

خلاصه‌سازی استخراجی متن با استفاده از رویکرد‌های یادگیری عمیق[[1]](#footnote-1)

**استاد راهنما:**

**دکتر حمیدرضا برادران کاشانی**

**استاد مشاور:**

**دکتر افسانه فاطمی**

**دانشجو:**

**امیررضا صدیقین**

**شهریور 1402**



**تعهدنامه اصالت اثر**

اینجانب **امیررضا صدیقین** دانشجوی مقطع کارشناسی ارشد رشته مهندسی کامپیوتر گرایش هوش مصنوعی و رباتیک متعهد مى‌شوم که مطالب مندرج در این پایان نامه و بروندادهاى حاصل از آن، دستاورد پژوهشی اینجانب با اشراف و راهنمايىِ استاد **دکتر حمیدرضا برادران کاشانی** است و آن دسته از مطالب اين پایان نامه که حاصل تحقیقات دیگران بوده نیز طبق شيوه‌نامه‌هاى مصوّبِ ارجاع، مستند شده و در فهرست منابع و مآخذ اين پژوهش آمده است. ضمنا اظهار مى‌دارم که اين پایان نامه پيش‌تر برای دریافت هیچ مدرک تحصیلی ارائه نشده است. بديهى است دانشگاه اصفهان برای خود اين حق را قائل است که در صورت احراز و اثبات هر گونه تخلف در اين باره، مدرک تحصیلی اینجانب را از درجه اعتبار ساقط نماید و ضمن درج موضوع در جراید کثیر الانتشار، کليه امتیازات و حقوقی را که به موجب آن پس از دوران تحصیل، از آنها بهره‌مند گشته‌ام، از اینجانب سلب و به طرف ذى‌نفع مسترد کند.

|  |
| --- |
| ـ برای رساله‌هايى که با حمایت جزيىِ مراکز برون‌دانشگاهى تدوين شده است، عبارت زير تکمیل شود**:**  **اين پایان نامه در دانشگاه اصفهان و با حمایت انجام شده است.** |

|  |
| --- |
| ـ برای رساله‌هايى که طی یک قرارداد مشخص، تحت حمایت سازمان یا نهادی تدوين شده است، عبارت زير تکمیل شود**:**  **اين پایان نامه در دانشگاه اصفهان و با حمایت طی قرارداد شماره انجام شده است.** |

|  |
| --- |
| ـ برای همه رساله‌های تدوین شده در دانشگاه اصفهان عبارت زير درج شود:  **کليه حقوق مادی و معنوی مترتب بر دستاوردهای مطالعات و نوآورى‌های ناشی از پژوهش در اين پایان نامه در چارچوب آيين‌نامه مالکیت فکری و تجاری‌سازىِ دانشگاه تعيين مى‌شود.** |

**نام و نام خانوادگى دانشجو: امضاء**

**نام و نام خانوادگى استاد (/ استادان) راهنما: امضاء**

حوزه معاونت پژوهش و فناوری

**دانشگاه اصفهان**

**دانشکده مهندسی کامپیوتر**

**گروه هوش مصنوعی و رباتیک**

پايان‌نامه آقای

**امیررضا صدیقین**

دانشجوی رشته‌ی مهندسی کامپیوتر گرایش هوش مصنوعی و رباتیک

**عنوان پایان نامه**

به عنوان بخشي از ملزومات درجه کارشناسي ارشد

در تاريخ 01/01/1402 توسط هيات داوران زير بررسي و با درجه ............ به تصويب نهايي رسيد.

1. استاد راهنمای پایان‌نامه دکتر نام استاد راهنما با مرتبه‌ی علمی مرتبه استاد راهنما از دانشگاه/موسسه نام-دانشگاه-یا-موسسه امضا
2. استاد مشاور پایان‌نامه دکتر نام استاد مشاور با مرتبه‌ی علمی مرتبه استاد مشاور از دانشگاه/موسسه نام-دانشگاه-یا-موسسه امضا
3. استاد داور داخل گروه دکتر نام داور داخلی با مرتبه‌ی علمی مرتبه علمی داور داخلی از دانشگاه/موسسه نام-دانشگاه-یا-موسسه امضا
4. استاد داور خارج از گروه دکتر نام داور خارجی با مرتبه‌ی علمی مرتبه علمی داور خارجی از دانشگاه/موسسه نام-دانشگاه-یا-موسسه امضا

مهر و امضای مدیر گروه

**سپاس ‌گزاری**

سپاس از اساتید بزرگوارم که بدون کمک آنها امکان حرکت در این مسیر وجود نداشت.

سپاس از همه افرادی که در این مسیر من را یاری نمودند.

**تقدیم به**

تقدیم به پدر و مادر عزیزم که من را در این مسیر همراهی کرده اند.

**چکیده**

متن چکیده فارسی

**کلیدواژه‌ها:** کلیدواژه‌های فارسی

# 

# فصل اول بیان مسئله‌ی پژوهشی

## مقدمه

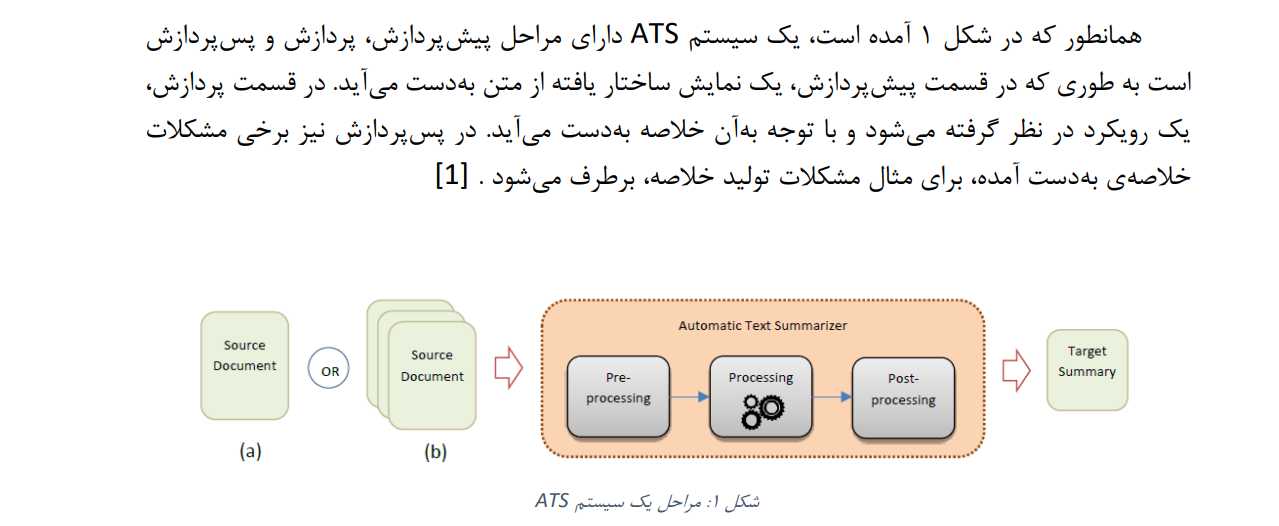
در دهه‌‌های گذشته، نحوه‌‌ی ذخیره و توزیع اطلاعات به‌ صورت چشمگیری تغییر پیدا کرده است. در حالی‌ که کتابخانه‌ها مسئولیت نگهداشت تعداد زیادی کتاب چاپی بوده است، امروزه بخش‌ زیادی از محتوای متنی به‌ صورت الکترونیکی در دسترس‌ هستند [1]. در جهان امروز، روزانه منابع متنی گسترده‌ای با سرعت بالا در حال تولید هستند، این منابع در تنوع و حجم بالا، مخزن غنی از اطلاعات می‌باشند [2]؛ با این حال، این توسعه یک چالش را به همراه دارد؛ مقادیر متنی موجود برای هر موضوع معین، معمولاً آنقدر زیاد است که یک فرد بتواند در یک زمان معقول آن را پردازش کند. به عنوان مثال، نسخه‌ی انگلیسی ویکی‌پدیا در سپتامبر 2018، حاوی 6 میلیون مقاله و سند بود. پروژه‌ی Google Books بیش از 25 میلیون نسخه کتاب را تا سال 2015 دیجیتالی کرده بود. همچنین تعداد کل صفحات شاخص گذاری شده، در سپتامبر 2018 برابر 4/4 میلیارد برآورد شده است [3]. فرآیند خلاصه‌سازی دستی متن بدون شک راهی موثر برای حفظ معنا و مفهوم متن است. با این حال، خلاصه‌سازی دستی متن، یک فعالیت وقت‌گیر و پرهزینه می‌باشد [4]. به همین منظور، مسئله‌ای در حوزه‌ی پردازش زبان‌های طبیعی[[2]](#footnote-2)، به عنوان خلاصه‌سازی خودکار متن[[3]](#footnote-3) (ATS) به وجود آمده است.

خلاصه سازی متن یک فرآیند مهم در پردازش زبان طبیعی است که در آن سعی می‌شود اطلاعات کلیدی و مهم متن اصلی به طور خلاصه و مختصری باقی بماند. این فرآیند به افزایش دسترسی به اطلاعات برای مخاطبین، صرفه‌جویی در زمان و انرژی، و همچنین ایجاد یک نمای کلی از متن‌های بزرگتر کمک می‌کند. یکی از رویکرد های خلاصه سازی خودکار متن، خلاصه سازی استخراجی متن می باشد که در آن جملات مهم متن استخراج می‌شوند. اگر متن دارای ساختار منظم و جملات معینی باشد، این رویکرد می‌تواند بهترین انتخاب باشد.

شبکه‌های عصبی به علت قدرت پردازشی و توانایی فهم و تحلیل داده‌های پیچیده از اهمیت بالایی در خلاصه سازی متن برخوردار هستند. با استفاده از شبکه‌های عصبی می‌توان اطلاعاتی را از متن‌ها استخراج کرد و یا جملات جدیدی را برای تشکیل خلاصه ایجاد کرد. این شبکه‌ها می‌توانند از مدل‌های زبانی پیشرفته مانند ترنسفرمر[[4]](#footnote-4) استفاده کنند که قدرت بسیار بالایی در فهم زبان و تولید محتوا دارند [5].

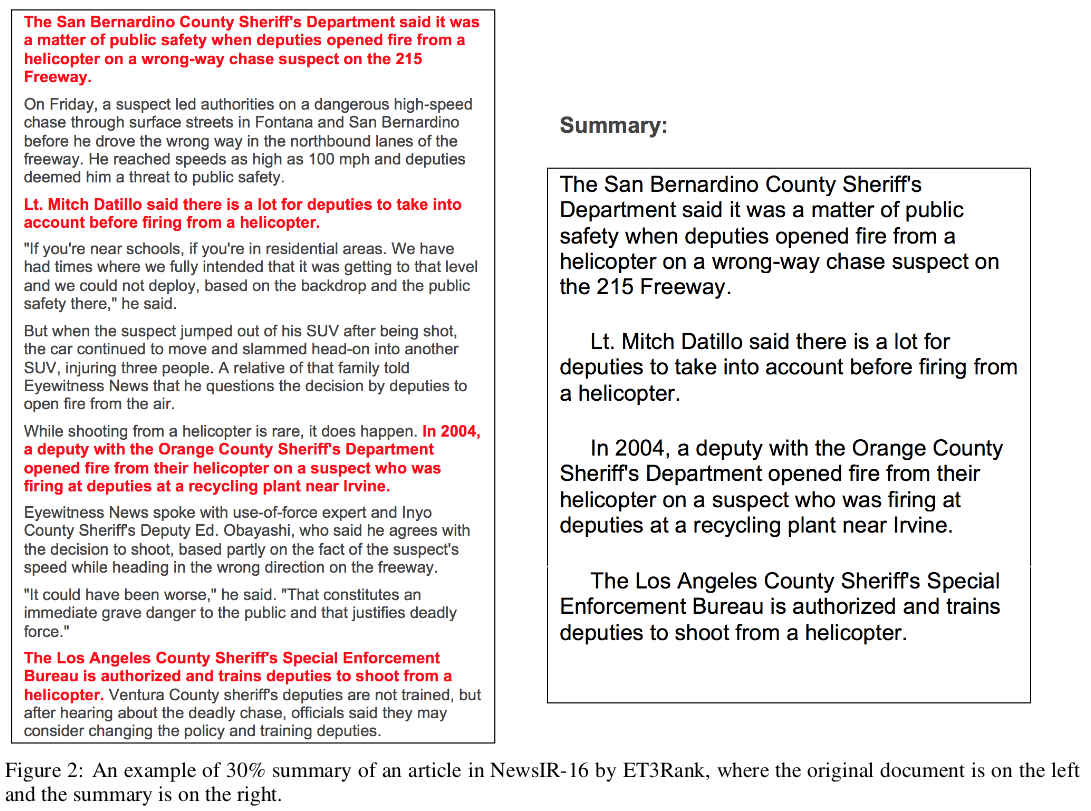
## 2-1- کلیات پژوهش

هدف سیستم‌های ATS، ساخت خلاصه‌هایی حاوی اطلاعات مهم و مرتبط با سند مربوطه، در زمان کوتاه و هزینه‌ی کم می‌باشند [3]. عمده‌ی کاربردهای سیستم‌های ATS در سیستم‌های بازیابی اطلاعات، استخراج اطلاعات و جواب‌دهی به سوالات است. برای مثال در سیستم‌های بازیابی اطلاعات، از ATS برای تقویت موتورهای جستجو استفاده می‌شود. منابع متنی می‌تواند در حوزه‌های مختلفی باشد، برای همین می‌توان کاربردهای متنوعی برای سیستم‌های ATS در نظر گرفت که خلاصه‌سازی اخبار، نظرات، کتاب، داستان، ایمیل، مقالات علمی، توییت‌ها و اطلاعات شبکه‌های اجتماعی، از موارد آن‌ها می‌باشند. همانطور که در شکل 1 آمده است، یک سیستم ATS دارای مراحل پیش‌پردازش[[5]](#footnote-5)، پردازش و پس‌پردازش[[6]](#footnote-6) متن است. در قسمت پیش‌پردازش متن، یک نمایش ساختار یافته از متن به دست می‌آید [2]. از عملیات موجود در این بخش می‌توان به برچسب‌گذاری POS[[7]](#footnote-7)، جداسازی کلمات توقف[[8]](#footnote-8)، هرس کردن[[9]](#footnote-9) و محاسبه‌ی ویژگی‌ها با رویکرد‌های مختلف، اشاره کرد. در قسمت پردازش، یک رویکرد خلاصه‌سازی در نظر گرفته می‌شود و با توجه به آن خلاصه به‌دست می‌آید [4]. در پس‌پردازش نیز برخی مشکلات خلاصه‌ی به‌دست آمده، برای مثال مشکل عدم تشخیص مرجع ضمایر یا مرجع زمان‌های نسبی، بر طرف می‌شود [2]. انتخاب ویژگی‌های مناسب یکی از مهم‌ترین گام‌های موجود در تولید یک مدل ATS می‌باشد. شباهت کلمات یک جمله با عنوان[[10]](#footnote-10)، موقعیت جملات در متن، ویژگی‌های TF-IDF[[11]](#footnote-11)، طول جملات، کلمات موضوعی[[12]](#footnote-12)(کلمات مرتبط با دامنه و دارای میزان تکرار زیاد)، اسامی خاص، شباهت بین جمله‌ای و اعداد، ویژگی‌های مهمی هستند که در یک سیستم ATS مورد توجه قرار می‌گیرند [4].

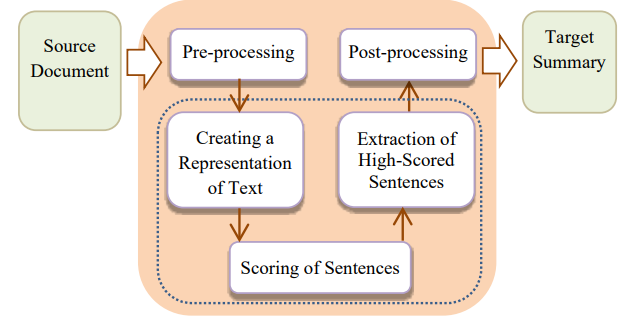


شکل 1: نمای کلی از مراحل یک سیستم ATS [4].

سیستم‌های ATS را از زاویه‌های متنوع دسته‌بندی کرد. از لحاظ تعداد سند، این سیستم‌ها به دو دسته‌ی تک‌سندی و چند‌سندی تقسیم می‌شوند. همچنین برپایه‌ی رویکردهای موجود در این نوع سیستم‌ها، سه رویکرد استخراجی[[13]](#footnote-13)، انتزاعی یا اسنتتاجی[[14]](#footnote-14) و ترکیبی[[15]](#footnote-15) وجود دارد . در رویکرد استخراجی، جملات خلاصه عیناً از جملات متن مرجع استخراج می‌شوند. در رویکرد انتزاعی، سعی در درک متن مرجع و ساخت خلاصه‌ی منسجم نزدیک به خلاصه‌های انسانی است. در رویکرد ترکیبی نیز، ابتدا با رویکرد استخراجی، جملات از متن ورودی استخراج، سپس با استفاده از این جملات، یک خلاصه‌ی منسجم تولید می‌شود. در ادامه به بررسی دقیق‌تر رویکرد‌های حوزه‌ی ATS، پرداخته خواهد شد [2]. شکل ؟؟ نمونه‌ای از خلاصه‌سازی متن با رویکرد استخراجی است. همچنین شکل ؟؟ نمای کلی از مراحل یک سیستم ATS با رویکرد استخراجی را نشان می‌دهد. در این مقاله به رویکرد استخراجی در خلاصه‌سازی متن‌های تک‌سندی پرداخته شده است.



شکل 2: نمونه‌ای از خلاصه‌سازی خودکار متن با رویکرد استخراجی [6].



شکل 3: نمای کلی از معماری سیستم‌های ATS با رویکرد استخراجی [2].

برای ساخت یک سیستم ATS چالش‌های زیادی وجود دارد، شناسایی بخش‌های مهم متن، خلاصه‌سازی سند‌های بزرگ مثل کتاب، خلاصه‌سازی هم‌زمان چند سند در قالب یک سیستم، ارزیابی سیستم و ساخت خلاصه شبیه به خلاصه‌های انسان از جمله این چالش‌ها هستند [2]. در ادامه به برخی از چالش‌های اصلی در رویکرد استخراجی پرداخته خواهد شد.

* **مشکل افزونگی**[[16]](#footnote-16)[3]

افزونگی باعث کاهش کیفیت یک خلاصه می‌شود. وظیفة خلاصه‌ی یک متن علاوه بر دور انداختن مطالب زائد، انتقال محتوای اصلی متن نیز می‌باشد. اگر مطالب تکراری را حذف کنیم، می‌توانیم اطلاعات مهم بیشتری در خلاصه جای دهیم. یکی از روش‌های جلوگیری از مشکل افزونگی، استفاده از اندازه‌گیری‌های شباهت بین جملات است .

* **مشکل بی‌ربط بودن**[[17]](#footnote-17)[3]

هدف اصلی یک سیستم‌ ATS، استخراج مطالب مرتبط با سند در یک نمای سریع و کلی، است. برخی از ویژگی‌های مورداستفاده در خلاصه‌سازی، ممکن است تمایل به ایجاد محتوای فرعی یا نامربوط در خلاصه داشته باشند. ازاین‌رو، بسیار مهم است که بدانیم کدام ویژگی‌ها و با چه میزان نسبت توجه به آن‌ها، برای ایجاد خلاصه باکیفیت مناسب هستند.

* **مشکل پوشش ندادن تمام موضوعات اصلی**[[18]](#footnote-18)[3]

پوشش تمام موضوعات اصلی سند در خلاصه، یک موضوع اساسی در سیستم‌های ATS است. یک خلاصه عمومی خوب باید اطلاعات مربوط به هر جنبه از سند را ذکر کند. بیشتر تکنیک‌های خلاصه‌سازی کنونی، تضمین پوشش تمام موضوعات اصلی را به کاربر نمی‌دهند. این مشکل بیشتر در خلاصه‌سازی چند سندی به دلیل تنوع و پراکندگی بیشتر موضوعات، قابل‌مشاهده هستند.

* **مشکل ناخوانایی و انسجام کم محتوا**[[19]](#footnote-19)[3]

یک خلاصه خوب باید خوانا و منسجم باشد. منظور از خواندنی و منسجم این است که مطالب خلاصه باید از نظر مفهومی با یکدیگر مرتبط و پیوستگی داشته باشند. بیشتر روش‌های استخراجی، فاقد این ویژگی هستند.

* **چالش درنظرگرفتن روابط بین جمله‌ای با فاصله زیاد از هم**[[20]](#footnote-20)[7]

جملات موجود در یک سند، هرچند با فاصله از هم ممکن است روابط و وابستگی‌های زیادی با هم داشته باشند. بیشتر مدل‌های پیشنهادی ارائه شده، توجه زیادی به روابط بین جمله‌ای ندارند. مدل‌های مبتنی بر شبکه‌های عصبی بازگشتی، تا حدی این مورد را در نظر می‌گیرند. با این‌حال مدل‌های مبتنی بر شبکه‌های عصبی بازگشتی، نمی‌توانند به‌خوبی روابط بین جملات با فاصله‌ی زیاد از هم را در نظر بگیرند.

تاثیر شبکه‌های عصبی بر روی خلاصه سازی خودکار متن بسیار قابل توجه است. این شبکه‌ها با بهره‌گیری از معماری‌های عمیق و مدل‌های توجه، قدرتمندترین ابزارها برای انجام خلاصه سازی متن هستند. با پیشرفت تکنولوژی شبکه‌های عصبی و استفاده از مجموعه داده‌های بزرگ، می‌توان به دقت و کیفیت بالاتری در خلاصه‌های تولید شده دست یافت. همچنین، توانایی‌های ترنسفورمر و شبکه‌های بازگشتی[[21]](#footnote-21)(RNN) در فهم ارتباطات طولانی‌تر متن و اجزای آن، موجب بهبود چشم‌گیری در تولید خلاصه‌های معنادار و مرتبط با محتوای متن اصلی شده است. از طرفی، شبکه‌های عصبی می‌توانند با تعمیم‌دهی و یادگیری از داده‌های متنوع و چندزبانه، بهبود مسائل ترجمه و خلاصه‌سازی متن را در زبان‌های مختلف ممکن سازند. به‌طور خلاصه، تحولات در حوزه شبکه‌های عصبی، از اهمیت بسیاری برای ارتقاء و بهبود روش‌های خلاصه سازی خودکار متن برخوردار است [8]. همچنین تاثیر گراف و شبکه‌های عصبی گراف بر روی خلاصه سازی خودکار متن از جمله موضوعات پرطرفدار در حوزه پردازش زبان طبیعی است. در این رویکرد، اطلاعات متنی به صورت گراف ترسیم می‌شوند که اجزا و ارتباطات میان واژه‌ها را نشان می‌دهد. این گراف‌ها می‌توانند با استفاده از روش‌های مبتنی بر شبکه‌های عصبی گراف، مورد استفاده قرار گیرند تا خلاصه‌های متنی با کیفیت و منطبق با محتوای اصلی ایجاد شوند. شبکه‌های عصبی گراف با توجه به ارتباطات بین واژه‌ها و ساختار معنایی متن، می‌توانند بهبود قابل توجهی در فهم و تولید خلاصه‌های معنادار داشته باشند [9], [10].

## 3-1- سوالات پژوهش

در طی پژوهش انجام شده به سوالاتی در زمینه‌ی خلاصه‌سازی خودکار متن با رویکرد استخراجی پاسخ داده شده است. در بخش نتایج و پیشینه پژوهش به میزان تاثیر شبکه‌های عصبی، به خصوص شبکه‌های عصبی گراف و میزان تاثیر ساخت گراف در مسئله‌ی ATS پرداخته شده است. همچنین این بخش‌ها، شامل مباحث دیگر از قبیل ویژگی‌ها و پیش‌پردازش‌های مناسب برای مسئله‌ی ATS می‌شوند.

## 4-1- اهداف پژوهش

با وجود روابط و متغیر‌های گوناگون در یک متن، هدف این پژوهش ارائه‌ی مدلی با رویکرد شبکه‌های عصبی جهت مدل‌سازی مناسب روابط و اطلاعات مسئله‌ی خلاصه‌سازی خودکار متن با رویکرد استخراجی می‌باشد. در این پژوهش به بررسی معماری‌های مختلف شبکه‌ی عصبی، تاثیر ساختار‌های گرافی بر روی نتایج و ویژگی‌های مناسب برای این مسئله پرداخته شده است.

## 5-1- روش ارائه شده

## 6-1- سازماندهی مطالب

مطالبی که در فصول آینده آمده است به قرار زیر است:

* فصل دوم به ادبیات موضوع و توضیح مفاهیم پایه برای استفاده در این پژوهش پرداخته است. در این بخش مواردی همچون مفاهیم شبکه‌ی عصبی، سنجه‌ها و معیار‌های مناسب در مسئله‌ی ATS و همچنین مباحث ریاضی و گرافی در این زمینه وجود دارند.
* در فصل سوم کار‌های پیشین در این حوزه و چالش‌های آن‌ها بررسی شده است. برخی تحقیقات و مدل‌های مبتنی بر شبکه‌عصبی که در حوزه‌ی ATS موجود است، در این بخش ذکر شده است.
* فصل چهارم شامل توضیح روش ارائه شده و بررسی جوانب مختلف آن می‌باشد. در این بخش برخی روش‌های آزمایش شده به همراه بررسی آن‌ها وجود دارند.
* در فصل پنجم پنجم نتایج آزمایشهای روش ‌های ارائه شده بر روی مجموعه داده مختلف بررسی شده و نتایج مورد تحلیل و بررسی قرار گرفته است.
* در نهایت در فصل ششم نتیجه‌گیری نهایی و جمع‌بندی پژوهش ارائه و پیشنهاداتی برای ادامه‌ی پژوهش داده شده است.

# فصل دوم ادبیات موضوع

# 2-1- مقدمه

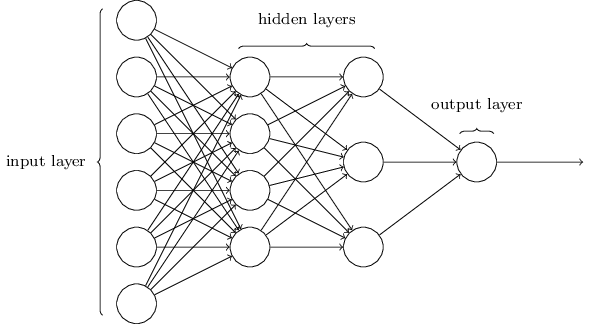
ساختار‌ها و روش‌های مختلفی برای حل مسائل خلاصه‌سازی متن ارائه شده است. در اکثر روش‌های نوین خلاصه‌سازی متن، یادگیری عمیق و ساختار‌های آن مهم‌ترین چهارچوب مورد استفاده است. از جمله شبکه‌های عصبی پیچشی (CNN)[[22]](#footnote-22) و شبکه‌های عصبی بازگشتی (RNN)[[23]](#footnote-23) از جمله شبکه‌های عصبی پرکاربرد در مسائل پردازش زبان طبیعی[[24]](#footnote-24) و پردازش متن می‌باشند. همچنین در سال‌های اخیر شبکه‌های عصبی گراف[[25]](#footnote-25) تحولات زیادی در یادگیری عمیق[[26]](#footnote-26) به وجود آورده است.

در این فصل در مورد انواع شبکه‌های عصبی و روش‌ محاسبات آن‌ها و خاصیت هر کدام توضیحاتی داده شده است . همچنین معیار‌های ارزیابی مدل‌ها در خلاصه‌سازی متن معرفی می‌گردد و در ادامه به چگونگی استفاده از شبکه‌های عصبی در پردازش زبان طبیعی به خصوص خلاصه‌سازی خودکار متن پرداخته می‌شود.

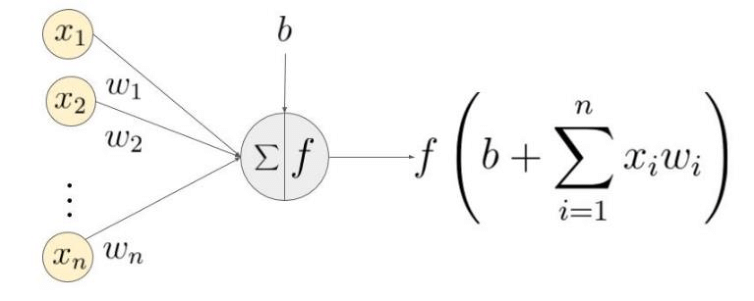
# 2-2- یادگیری عمیق

یادگیری عمیق یک حوزه پویا و حیاتی در علم داده[[27]](#footnote-27) و یادگیری ماشین[[28]](#footnote-28) است که با استفاده از شبکه‌های عصبی عمیق سعی دارد نمایش‌های پیچیده داده‌ها به منظور استخراج ویژگی‌های نهان بدست آورد. شبکه‌های عصبی عمیق به عنوان مدل‌های محاسباتی با تعداد لایه‌های بسیار زیاد شناخته می‌شوند. این روش‌ها با الهام از ساختار مغز انسان و اقتباس از عملکرد نورون‌ها، مفاهیمی چون یادگیری نمایش‌ها، تشخیص الگوها و پیش‌بینی‌های دقیق را فراهم می‌کنند. شبکه‌های عصبی عمیق به عنوان ابزاری کارآمد در بسیاری از حوزه‌ها از جمله پردازش تصویر، پردازش زبان‌های طبیعی، پردازش صورت و حتی بازیابی اطلاعات، به کار گرفته می‌شوند [11].

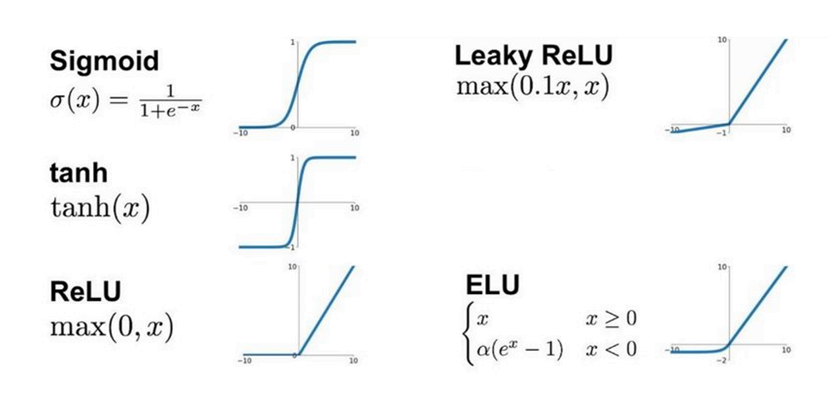
شبکه عصبی چندلایه [[29]](#footnote-29) یکی از مدل‌های اولیه و مهم در زمینه یادگیری عمیق است. این شبکه به عنوان یک نمونه از شبکه‌های عصبی عمیق، از چند لایه از نورون‌ها (واحدهای پردازشگر) تشکیل شده است. همانطور که در شکل ؟؟ آمده است، در این نوع شبکه‌ها، از سه بخش اصلی به نام لایه‌ی ورودی[[30]](#footnote-30)، لایه‌های پنهان[[31]](#footnote-31)، و لایه خروجی[[32]](#footnote-32) تشکیل شده است. در لایه ورودی، ویژگی‌های ورودی به شبکه وارد می‌شوند. لایه‌های پنهان با انجام عملیات‌های خطی و توابع غیرخطی[[33]](#footnote-33) یا فعالسازی[[34]](#footnote-34) بر روی ورودی‌ها، اطلاعات را انتزاع می‌کنند و ویژگی‌های پنهان را استخراج می‌کنند. در نهایت، لایه خروجی نتایج پایانی را تولید می‌کند. این شبکه‌ها از یک معماری کاملاً متصل [[35]](#footnote-35)هستند، به این معنی که هر نورون در یک لایه با تمام نورون‌های لایه قبل و بعد از خود ارتباط دارد. این ویژگی باعث تغییر ابعاد داده و استخراج ویژگی‌های پیچیده‌تر می‌شود. برخی مواقع داده‌ها به صورت خطی[[36]](#footnote-36) تفکیک پذیر نیستند، به همین منظور در برخی از لایه‌ها از توابع غیرخطی یا فعالسازی استفاده می‌کنند. همانطور که در شکل ؟؟ نحوه‌ی محاسبات درون هر نرون به نمایش گذاشته شده، ورودی هر نرون خروجی لایه‌های قبل یا لایه‌ی ورودی شبکهی عصبی می‌باشد. همچنین متناظر با هر مؤلفه ورودی، وزنی در آن ضرب می‌شود که وزن‌ها مقادیر قابل آموزش می‌باشند. در هر نرون یک ورودی با مقدار 1 به همراه وزن آن (بایاس) نیز اضافه می‌شود. در آخر تابع فعالسازی برروی مجموع ضرب ورودی با وزن‌های متناظر اعمال می‌شود. در شکل ؟؟، برخی توابع فعالسازی متداول، معرفی شده‌اند.



شکل 4: ساختار یک شبکه عصبی که دارای سه بخش اصلی لایه‌ی ورودی، لایه مخفی و لایه‌ی خروجی میباشد [12].



شکل 5: نحوه‌ی محاسبه‌ی خروجی هر نرون در شبکه‌ها عصبی. هر نرون از چندین ورودی و وزن به همراه بایاس(مقدار وزن با ورودی 1) تشکیل شده و با استفاده از تابع غیر خطی f ویژگی غیر خطی به خروجح میدهد [13].



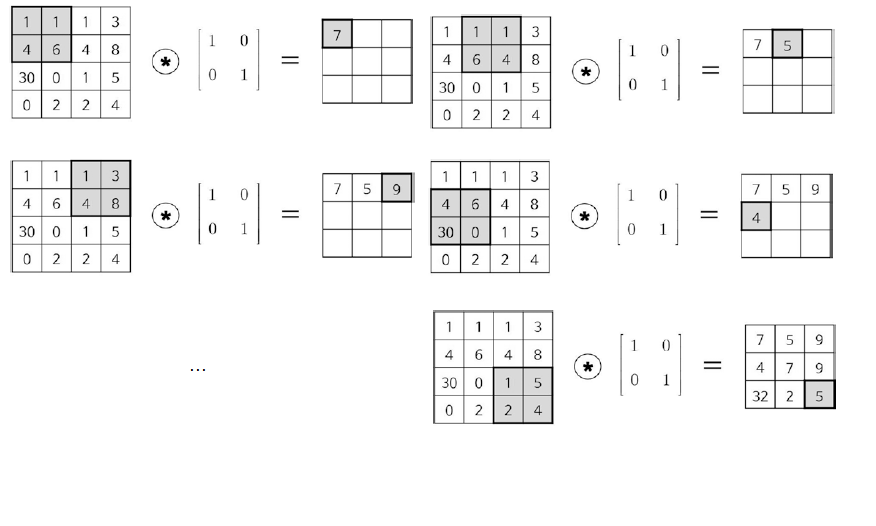
شکل 6: برخی توابع فعالسازی معروف[14]

متناسب با نوع مسئله و الگوریتم‌های بهینه‌سازی، فرایند آموزش شبکه‌ی عصبی شامل دو عنصر مهم تابع خطا[[37]](#footnote-37) و الگوریتم بهینه‌سازی[[38]](#footnote-38) می‌باشد. در هر دور از آموزش، داده‌ها به شبکه‌ی عصبی ارسال شده و متناسب با نوع مسئله، تابع خطا مقادیر خطا را محاسبه می‌کند و با استفاده از الگوریتم‌های بهینه‌سازی که معمولا با محاسبه‌ی مشتق خطا و قابلیت زنجیره‌ای آن همراه است، مقادیر وزن‌ها را بروزرسانی می‌کند و سعی می‌کند مقادیر خطا را به حداقل حالت خود برسند [11].

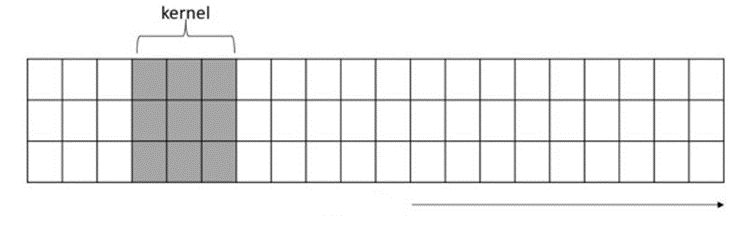
در ادامه به یک سری از شبکه‌های عصبی متداول و روش‌های معروف در این حوزه، پرداخته می‌شود.

## 1-2-2- شبکه‌های عصبی پیچشی[[39]](#footnote-39)

شبکه‌های عصبی پیچشی یا CNNها، نوع دیگری از شبکه‌های عصبی هستند که معمولا برای استخراج ویژگی‌ها و استخراج الگو‌ها استفاده می‌شود. این شبکه‌ها معمولا برای پردازش تصاویر و پردازش داده‌های توالی استفاده می‌شوند. شبکه‌های عصبی پیچشی از لایه‌های پیچشی[[40]](#footnote-40) و لایه‌های ادغام[[41]](#footnote-41) تشکیل شده‌اند. لایه‌های پیچشی به عنوان فیلترها یا کرنل‌ها عمل کرده و با حرکت این فیلترها روی داده‌ها، ویژگی‌های مختلف با محاسبه‌ی عملیات پیچشی[[42]](#footnote-42) از آن استخراج می‌کنند. عملیات پیچشی، حاصل جمع ضرب هر مؤلفه کرنل با مؤلفه نظیرش در داده می‌باشد. لایه‌های ادغام به وسیله‌ی روش‌های تجمیع[[43]](#footnote-43)، با کاهش ابعاد داده و حذف اطلاعات غیرضروری، کمک به کاهش تعداد پارامترها و افزایش سرعت آموزش می‌کنند[14] [15]. لایه‌های پیچشی به دو دسته‌ی یک بعدی و دو بعدی تقسیم‌میشوند. کرنل‌ها‌ی لایه‌ی پیچشی دو بعدی در دو جهت حرکت می‌کنند و بیشتر برای پردازش تصویر کاربرد دارند. شکل ؟؟، عملیات پیچشی دو بعدی را نشان می‌دهد. در مقابل، کرنل‌های لایه‌ی پیچشی یک بعدی تنها در یک بعد حرکت می‌کنند و بیشتر برای پردازش‌ داده‌های توالی، مثل پردازش سیگنال یا پردازش متن، استفاده می‌شوند. شکل ؟؟ نحوه‌ی حرکت کرنل پیچشی یک بعدی را نشان می‌دهد. در فرایند آموزش این شبکه‌ها، مؤلفه‌های شبکه بروزرسانی شده و الگو‌های مناسب از داده‌ها استخراج می‌شود.



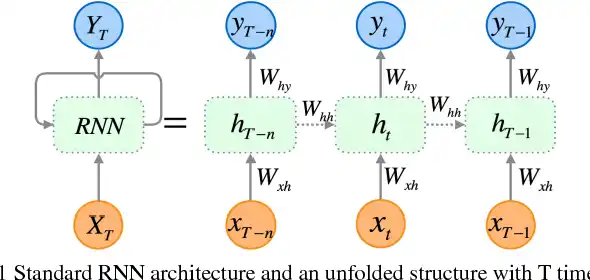
شکل 7 : مراحل بدست آوردن خروجی از اعمال عملیات پیچش یک فیلتر بر روی یک ماتریس ورودی [17] .



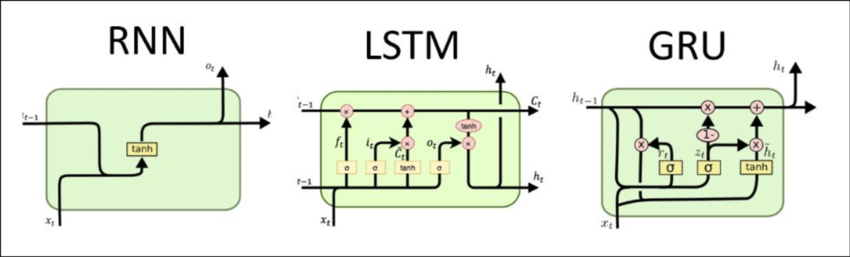
شکل 8: نحوه‌ی حرکت کرنل پیچشی یک بعدی[18]

## 2-2-2- شبکه‌های عصبی بازگشتی [[44]](#footnote-44)

شبکه‌های عصبی بازگشتی یا RNN‌ها یک نوع از معماری‌های شبکه‌های عصبی هستند که برای پردازش داده‌های توالی، مانند داده‌های زمانی یا پردازش کلمات در جمله، استفاده می‌شوند. همانطور که در شکل ؟؟ نمایش داده شده است، این نوع از شبکه‌های عصبی از ویژگی حافظه‌ی بازگشتی[[45]](#footnote-45) برخوردار هستند که این امکان را فراهم می‌سازد تا اطلاعات از گذشته در فرایند آموزش و پیش‌بینی مورد استفاده قرار گیرد. این شبکه‌ها، در مسائل مختلفی مانند ترجمه ماشینی[[46]](#footnote-46)، تشخیص گفتار[[47]](#footnote-47)، پیش‌بینی متن[[48]](#footnote-48)، و حتی تولید متون[[49]](#footnote-49) جدید مؤثر هستند. محو شدن اطلاعات داده‌های اولیه و مشکل حذف گرادیان از مشکلات جدی این نوع معماری می‌باشد، با این حال با تغییرات و اصلاح طراحی‌های موجود در این شبکه‌ها، بهبود‌های چشمگیری در مدیریت این چالش‌ها ایجاد شده است. LSTM‌[[50]](#footnote-50) و GRU‌[[51]](#footnote-51) که در به نمایش درآمده‌اند، از متداول‌ترین شبکه‌های عصبی بازگشتی می‌باشند که در زمینه‌ی این چالش‌ها بهینه شده‌اند [19]. شکل ؟؟، معماری‌های GRU و LSTM و تفاوت آن‌ها با RNNهای اولیه را نمایش می‌دهد.



شکل 9: نحوه‌ی عملکرد شبکه‌های عصبی بازگشتی[20] .

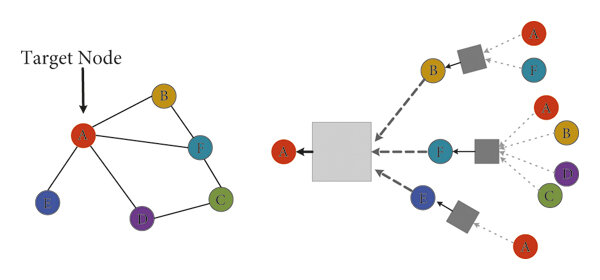


شکل 10: شبکه‌های GRU و LSTM و تفاوت آن‌ها با شبکه‌های RNN اولیه [21].

## 3-2-2- شبکه‌های عصبی گراف[[52]](#footnote-52)

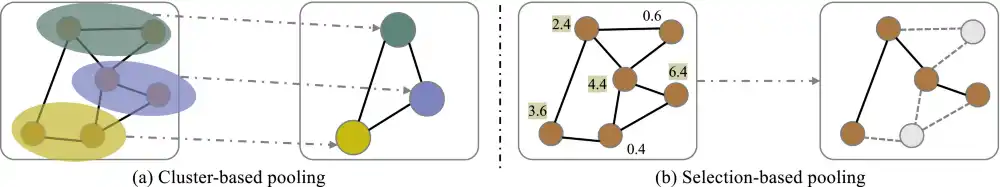
شبکه‌های عصبی گراف یک دسته از مدل‌های یادگیری عمیق هستند که برای مدل‌سازی داده‌های گرافی مانند شبکه‌های اجتماعی، شبکه‌های مولکولی و ساختارهای گرافی مشابه به کار می‌روند. این شبکه‌ها توانمندی بالایی در تفسیر و تحلیل روابط پیچیده و الگوهای شبکه‌های گرافی دارند. به عبارت دیگر، آنها قابلیت مدل‌سازی تعاملات و وابستگی‌های بین گره‌ها و لینک‌های یک گراف را دارا هستند [22].

شبکه‌های عصبی گراف از مکانیزم انتقال پیام[[53]](#footnote-53)، به منظور تبادل اطلاعات بین گره‌ها در یک گراف بهره می‌برند. در این مکانیزم، هر گره اطلاعات مرتبط با خود را با همسایگان خود به اشتراک می‌گذارد و از اطلاعات جمع‌آوری شده برای به‌روزرسانی ویژگی‌های خود استفاده می‌کند. این فرآیند چند مرحله‌ای ادامه می‌یابد تا اطلاعات به صورت تدریجی از یک گره به دیگری منتقل شوند. همچنین به غیر از اطلاعات درون یک گره، میتوان برای یال‌ها نیز اطلاعات و ارتباطاتی[[54]](#footnote-54) نیز در نظر گرفت. مدیریت پیچیدگی مدل‌ها، تعامل با ساختار گراف، انعطاف پذیری در مقابل اندازه‌ی گراف‌ها، آموزش ناهمگن[[55]](#footnote-55) گراف‌ها و مقاومت در برابر نویز از قابلیت‌های شبکه‌های عصبی می‌باشد[22]. یکی از اولین و متداول‌ترین شبکه‌های عصبی گراف، GCN‌ها[[56]](#footnote-56) می‌باشند که در شکل ؟؟، نحوه‌ی عملکرد آن نمایش داده شده است. یکی دیگر از شبکه‌های عصبی گراف معروف، شبکه‌ی عصبی GAT‌[[57]](#footnote-57) می‌باشد که با استفاده از مکانیزم توجه، می‌تواند میزان اهمیت بین گره‌ها در فرایند انتقال پیام را آموزش ببیند.



شکل 11: نحوه‌ی انتقال پیام برای یک گره مشخص در شبکه‌ی GCN [23]

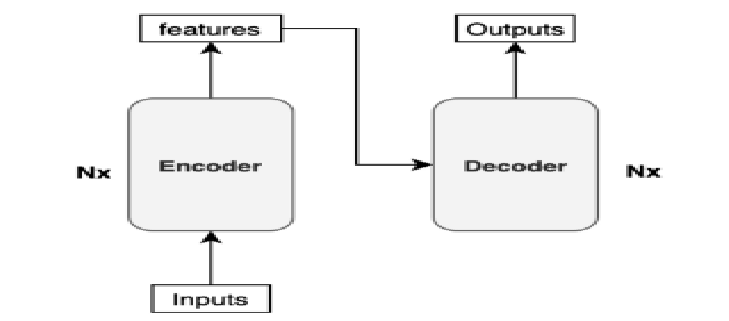
برخی مواقع در شبکه‌های عصبی گراف از لایه‌های ادغامی گراف[[58]](#footnote-58) به منظور به‌دست آوردن گراف کوچک‌تر استفاده می‌شود. همچنین با این روش می‌توان نمایشی برای گراف مورد نظر نیز به‌دست آورد. در شکل ؟؟ یک نمونه از ادغام گره‌های گراف نمایش داده شده است.



شکل 12: ادغام گره‌های گراف و تولید گراف کوچکتر با استفاده از لایه‌ی ادغام گراف [24]

# 3-2- ساختار کدگذار-کدگشا [[59]](#footnote-59)

ساختار‌های مختلفی برای حل مسائل متفاوت بر بستر یادگیری عمیق ارائه شده‌اند. یکی از این ساختار‌ها، ساختار کدگذار-کدگشا است. ساختار کدگذار-کدگشا یک معماری متداول در حوزه شبکه‌های عصبی است که برای مسائل مختلفی از جمله ترجمه ماشینی، تولید متن، و تولید تصویر[[60]](#footnote-60) مورد استفاده قرار می‌گیرد. کدگذار مسئول تبدیل ورودی به یک فضانی نهان[[61]](#footnote-61) یا بردار ویژگی است. این فضای نهان معمولاً اطلاعات مهم و تجمعی از ویژگی‌های ورودی را نمایندگی می‌کند. سپس کدگشا با استفاده از این نمایش نهان، وظیفه بازسازی و یا تولید خروجی مورد نظر را برعهده دارد [25]. شکل ؟؟، ساختار کلی کدگذار-کدگشا را نشان می‌دهد.



شکل 13: ساختار کدگذار-کدگشا [26]

## 4-2- روش‌های مبتنی بر توجه[[62]](#footnote-62)

1. Extractive text summarization using deep learning approaches [↑](#footnote-ref-1)
2. Natural language processing [↑](#footnote-ref-2)
3. Automatic text summarization [↑](#footnote-ref-3)
4. Transformer [↑](#footnote-ref-4)
5. Preprocessing [↑](#footnote-ref-5)
6. Postprocessing [↑](#footnote-ref-6)
7. Part of speech tagging [↑](#footnote-ref-7)
8. Stop words Filtering [↑](#footnote-ref-8)
9. Stemming [↑](#footnote-ref-9)
10. Title similarity [↑](#footnote-ref-10)
11. Term frequency – inverse term frequency [↑](#footnote-ref-11)
12. Thematic words [↑](#footnote-ref-12)
13. Extractive [↑](#footnote-ref-13)
14. Abstractive [↑](#footnote-ref-14)
15. Hybrid [↑](#footnote-ref-15)
16. Problem of redundancy [↑](#footnote-ref-16)
17. Problem of irrelevancy [↑](#footnote-ref-17)
18. Problem of loss of coverage [↑](#footnote-ref-18)
19. Problem of non-readability and less cohesive content [↑](#footnote-ref-19)
20. Long-distance relationships [↑](#footnote-ref-20)
21. Recurrent Neural Network [↑](#footnote-ref-21)
22. Convolutional neural networks [↑](#footnote-ref-22)
23. Recurrent neural networks [↑](#footnote-ref-23)
24. Natural language processing (NLP) [↑](#footnote-ref-24)
25. Graph neural networks [↑](#footnote-ref-25)
26. Deep learning [↑](#footnote-ref-26)
27. Data science [↑](#footnote-ref-27)
28. Machine learning [↑](#footnote-ref-28)
29. Multi-layer perceptron (MLP) [↑](#footnote-ref-29)
30. Input layer [↑](#footnote-ref-30)
31. Hidden layers [↑](#footnote-ref-31)
32. Output layer [↑](#footnote-ref-32)
33. Non-linear [↑](#footnote-ref-33)
34. Activation functions [↑](#footnote-ref-34)
35. Fully connected [↑](#footnote-ref-35)
36. Linear [↑](#footnote-ref-36)
37. Loss function [↑](#footnote-ref-37)
38. Optimizer algorithms [↑](#footnote-ref-38)
39. Convolutional neural network [↑](#footnote-ref-39)
40. Conolutional layers [↑](#footnote-ref-40)
41. Pooling layers [↑](#footnote-ref-41)
42. Convolution [↑](#footnote-ref-42)
43. Aggregation methods [↑](#footnote-ref-43)
44. Recurrent neural networks [↑](#footnote-ref-44)
45. Recurrent memory [↑](#footnote-ref-45)
46. Machine translation [↑](#footnote-ref-46)
47. Speech recognition [↑](#footnote-ref-47)
48. Text prediction [↑](#footnote-ref-48)
49. Text generation [↑](#footnote-ref-49)
50. Long short-term memory [↑](#footnote-ref-50)
51. Gated recurrent unit [↑](#footnote-ref-51)
52. Graph neural networks [↑](#footnote-ref-52)
53. message passing [↑](#footnote-ref-53)
54. relations [↑](#footnote-ref-54)
55. Heterogeneous Learning [↑](#footnote-ref-55)
56. Convolutional network [↑](#footnote-ref-56)
57. Graph attention networks [↑](#footnote-ref-57)
58. Graph pooling layers [↑](#footnote-ref-58)
59. Encoder-decoder structure [↑](#footnote-ref-59)
60. Image generation [↑](#footnote-ref-60)
61. Latent space [↑](#footnote-ref-61)
62. Attention-based methods [↑](#footnote-ref-62)