



# گزارش پژوهش درس شبکه‌های عصبی

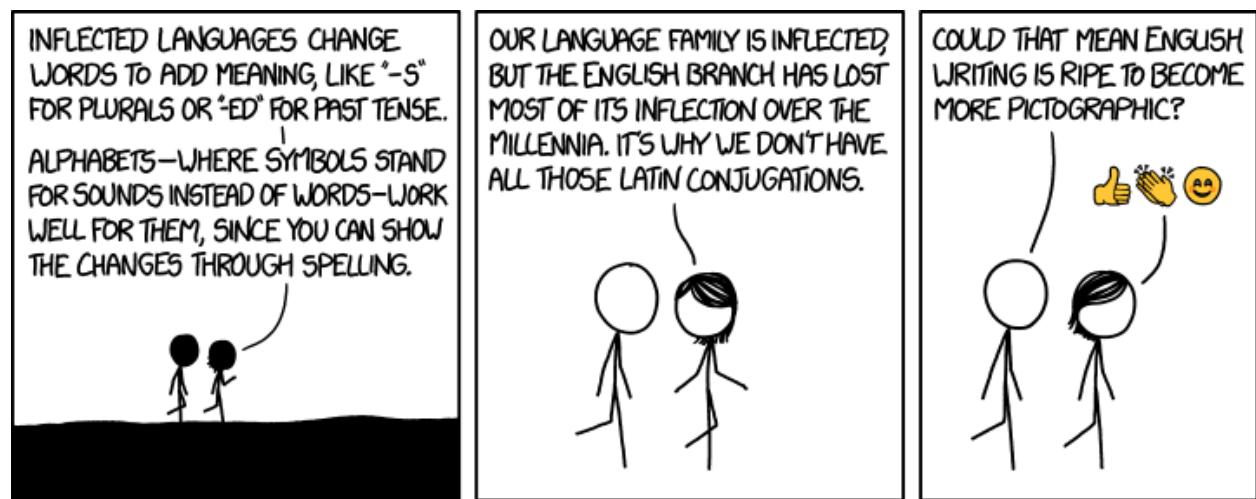
نام استاد : هادی ویسی

نام دانشجو : هادی رهجو

پردازش زبان طبیعی (nlp) شاخه‌ای از علوم رایانه، مهندسی اطلاعات و هوش مصنوعی

است که به تعامل بین کامپیوترها و زبان‌های انسانی (طبیعی) می‌پردازد. چالش‌ها در پردازش زبان طبیعی شامل تشخیص گفتار، درک زبان طبیعی و تولید زبان طبیعی است.

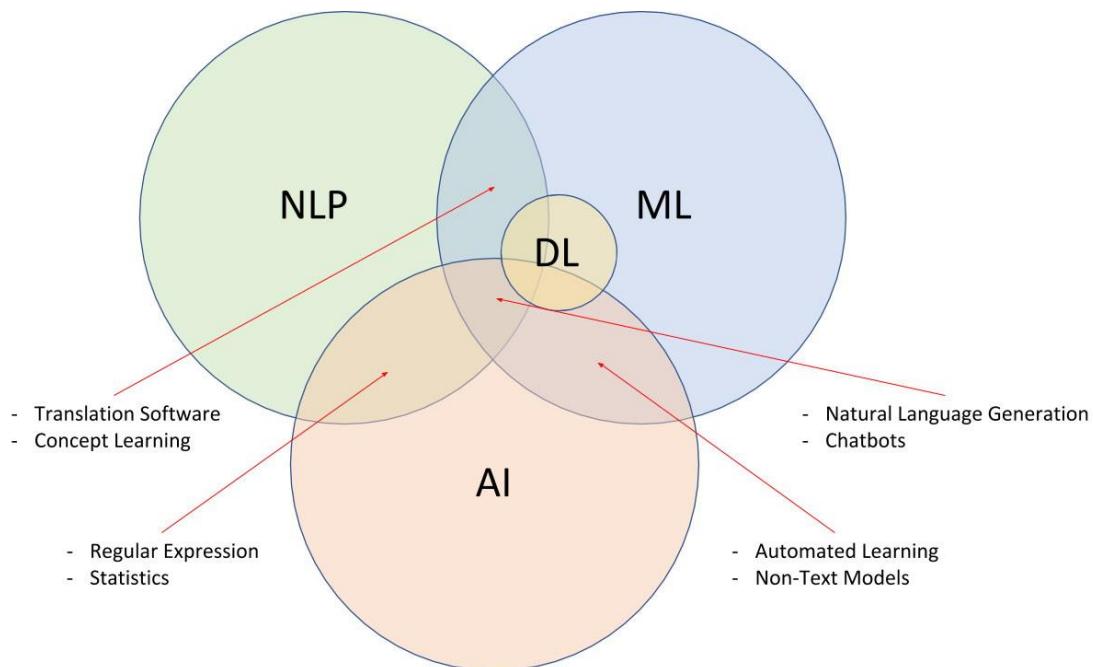
داده‌های ساختنیافته و به طور خاص متن، تصاویر و ویدیوها حاوی حجم بالای اطلاعات هستند. با این حال به دلیل پیچیدگی ذاتی پردازش و تجزیه و تحلیل این داده‌ها، افراد غالباً از صرف زمان و تلاش زیاد روی مجموعه داده‌های ساختنیافته که در حکم کاوش معدن طلا هستند اجتناب می‌کنند.



پردازش زبان طبیعی فناوری ای است که از کامپیوتر استفاده می کند تا بتواند زبان طبیعی انسان را بفهمد. فهم این که انسان ها چگونه با یکدیگر ارتباط برقرار می کنند، به سادگی ممکن نیست.

هدف نهایی NLP ، خواندن، رمزگشایی کردن، فهمیدن و یافتن ارزش های زبان انسان هاست. بسیاری از تکنیک های NLP ، برای یافتن معانی زبان های انسانی، وابسته به یادگیری ماشین (Machine Learning) هستند. تصویر زیر به خوبی می تواند نقاط اشتراک NLP، یادگیری ماشین و هوش مصنوعی

را نشان دهد:



تاریخچه پردازش زبان طبیعی به طور کلی در دهه ۱۹۵۰ آغاز شد، اگرچه کار را می‌توان از دوره‌های قبلی پیدا کرد. در سال ۱۹۵۰، آلن تورینگ مقاله‌ای با عنوان "ماشین‌آلات و اطلاعات" منتشر کرد که آنچه در حال حاضر آزمون تورینگ نامیده‌می‌شود را معیار هوش نامید.

در سال ۱۹۵۴ تجربه جورج تاون به طور کامل شامل ترجمه کامل بیش از شصت زبان‌روزی به زبان انگلیسی شد. نویسنده‌گان ادعا کردند که در طول سه یا پنج سال ترجمه‌ماشینی یک مشکل حل شده تلقی می‌شود. با این حال، پیشرفت واقعی بسیار کندتر بود، و پس از گزارش ALPAC در سال ۱۹۶۶، که نشان داد تحقیقات ده‌ساله نتوانسته‌اند انتظارات را برآورده کنند، بودجه ترجمه ماشینی به شدت کاهش یافته.

تحقیقات بیشتری در زمینه ترجمه ماشینی تا اواخر دهه ۱۹۸۰ انجام شد، زمانی که اولین ترجمه ماشینی آماری توسعه یافت. برخی از مهم‌ترین سیستم‌های پردازش زبان طبیعی که در دهه ۱۹۸۰ توسعه یافت عبارتند از: یک سیستم زبان طبیعی که در "بلوک‌های" محدود با واژگان محدود و الی زا، یک شبیه‌سازی روان روان درمانگر، نوشته شده توسط جوزف مک‌لین بین دو ام. الیزا با استفاده از تقریباً هیچ اطلاعاتی در مورد احساسات و عواطف انسانی، گاهی اوقات یک تعامل انسان شبیه انسان را فراهم می‌کرد. هنگامی که "بیمار" از پایه دانش بسیار کوچک فراتر رفته است، الیزا ممکن است یک پاسخ عمومی ارائه دهد، برای مثال، پاسخ دادن به "آسیب سر من" با "چرا گفتن سرت درد می‌کند؟

."

در طول دهه ۱۹۸۰، بسیاری از برنامه نویسان شروع به نوشتن "آنالوژی مفهومی" کردند که اطلاعات دنیای واقعی را به داده‌های قابل فهم کامپیوتری ساخت. تا دهه ۱۹۸۰، اغلب سیستم‌های پردازش زبان طبیعی براساس مجموعه‌های پیچیده قواعد نوشته شده بودند. با این حال، در اواخر دهه ۱۹۸۰، یک انقلاب در پردازش زبان طبیعی با معرفی الگوریتم‌های یادگیری ماشین برای پردازش زبان وجود داشت. این به دلیل افزایش مداوم توان محاسباتی (به عنوان مثال قانون Moore's) و کاهش تدریجی تسلط نظریه‌های زبان‌شناسی (به عنوان مثال دستور انتقالی) بود که پایه‌های نظری آن نوع زبان‌شناسی پیکره زبانی را که زمینه‌ساز رویکرد یادگیری ماشین به پردازش زبانی است، دلسوزد کرد.

برخی از الگوریتم‌های یادگیری ماشین مورد استفاده، مانند درخت‌های تصمیم‌گیری، سیستم‌های سخت را ایجاد می‌کنند اگر - آنگاه قوانین مشابه قواعد مكتوب موجود باشند. با این حال، برچسب گذاری بخشی استفاده از مدل‌های مارکوف پنهان را به پردازش زبان طبیعی ارایه کرد، و به طور فزاینده‌ای، تحقیقات بر مدل‌های آماری تمرکز کرده‌است، که تصمیم‌های ساده، احتمالی براساس اتصال وزن‌ها با ارزش واقعی به ویژگی‌هایی که داده‌های ورودی را می‌سازند. مدل‌های زبانی حافظه نهان که بسیاری از سیستم‌های بازشناسی گفتار در حال حاضر بر آن تکیه دارند نمونه‌هایی از چنین مدل‌هایی هستند.

چنین مدل‌هایی معمولاً وقتی داده‌های ورودی ناآشنا را ارایه می‌دهند، استحکام بیشتری دارند، به خصوص ورودی که حاوی خطأ هستند (همان طور که برای داده‌های دنیای واقعی رایج است)، و هنگامی که در یک سیستم بزرگ‌تر متشكل از وظایف فرعی چندگانه ادغام می‌شوند، نتایج قابل اطمینانی ایجاد می‌کنند.

بسیاری از موفقیت‌های اولیه قابل توجه در زمینه ترجمه ماشینی، به ویژه در تحقیقات ibm رخ داد، که در آن مدل‌های آماری پیچیده بیشتری توسعه یافته‌اند. این سیستم‌ها می‌توانستند از پیکره‌های متنه موجود استفاده کنند که توسط پارلمان کانادا و اتحادیه اروپا تولید شده بود در نتیجه قوانینی که خواستار ترجمه همه اقدامات دولتی به همه زبان‌های رسمی نظام مربوطه بودند. با این حال، اغلب سیستم‌های دیگر وابسته به پیکره بندی به طور خاص برای کارهای انجام‌شده توسط این سیستم‌ها هستند که (و اغلب به عنوان محدودیت اصلی در موفقیت این سیستم‌ها) به کار می‌روند. در نتیجه، تحقیقات زیادی به روش‌های یادگیری موثرتر از مقادیر محدود داده تبدیل شده‌است.

تحقیقات اخیر به طور فزاینده‌ای بر الگوریتم‌های یادگیری بدون ناظارت و ناظارت شده متمرکز شده‌است. چنین الگوریتم‌هایی قادر به یادگیری از داده‌هایی هستند که با پاسخ‌های مورد نظر، یا با استفاده از ترکیبی از داده‌های مشروح و غیر مشروح تعیین نشده‌اند. به طور کلی، این کار بسیار دشوارتر از یادگیری ناظارت شده است، و به طور معمول نتایج دقیق‌تری برای مقدار داده‌شده از داده‌های ورودی تولید می‌کند. با این حال، مقدار بسیار زیادی از داده‌های non موجود (از جمله، در میان چیزهای دیگر، کل محتوای وب گسترده جهانی) وجود دارد، که اگر الگوریتم مورد استفاده قرار گیرد پیچیدگی زمانی پایینی دارد که عملی باشد.

در دهه ۲۰۱۰، یادگیری نمایش و روش‌های یادگیری ماشین به شبکه عصبی در پردازش زبان طبیعی رواج پیدا کرد، به دلیل بخشی از نتایجی که نشان می‌دهد این تکنیک‌ها در بسیاری از

وظایف طبیعی زبان ، برای مثال در مدل‌سازی زبان ، و بسیاری دیگر انجام می‌شوند . تکنیک‌های مردمی شامل استفاده از embeddings کلمه برای ثبت ویژگی‌های معنایی کلمات و افزایش در یادگیری نهایی یک کار سطح بالاتر ( مثلا ، پرسش پاسخ ) به جای تکیه بر خط لوله وظایف میانی جداگانه ( به عنوان مثال ، برچسب گذاری روی گفتار ) . در برخی مناطق ، این تغییر مستلزم تغییرات اساسی در چگونگی طراحی سیستم‌های NLP است ، به گونه‌ای که رویکردهای مبتنی بر شبکه عصبی ممکن است به عنوان نمونه جدیدی متمایز از پردازش زبان طبیعی تلقی شوند . برای مثال ، عبارت ترجمه ماشین عصبی ( NMT ) بر این حقیقت تاکید دارد که رویکردهای عمیق مبتنی بر یادگیری برای ترجمه ماشینی به طور مستقیم به تبدیل توالی به توالی ، نیاز به گام‌های میانی مانند همترازی کلمات و مدل‌سازی زبان که در ترجمه ماشینی آماری استفاده می‌شوند ( SMT ) تاکید می‌کند .

## ابزارهای NLP

در زیر فهرستی از برخی از رایج‌ترین وظایف در پردازش زبان طبیعی آورده شده است . توجه داشته باشید که برخی از این وظایف کاربردهای مستقیم دنیای واقعی دارند ، در حالی که برخی دیگر به عنوان وظایف فرعی بکار می‌روند که برای حل مسایل بزرگ‌تر به کار می‌روند . اگرچه وظایف پردازش زبان طبیعی به هم گره خورده‌اند ، اغلب به دسته‌بندی برای راحتی تقسیم‌بندی می‌شوند . در زیر آمده است :

## Grammar induction

یک دستور زبان رسمی را تولید کنید که نحو زبان را توصیف می‌کند.

## Lemmatization

بردن کلمات صرفی تنها و رسیدن به شکل فرهنگ لغت  
وظیفه از بین پایه کلمه‌ای است که به قیاس یا Lemma معروف است.

## Morphological segmentation

کلمات جداگانه به تکواز تکی و سطح تکواز را شناسایی کنید . دشواری این کار به شدت بستگی به پیچیدگی مورفولوژی ( یعنی ساختار کلمات ) زبان در نظر گرفته می‌شود . انگلیسی مورفولوژی نسبتاً ساده‌ای دارد ، به خصوص مورفولوژی صرفی ، و بنابراین اغلب ممکن است که این وظیفه را کاملاً نادیده بگیریم و به سادگی تمام اشکال ممکن یک کلمه را مدل کنیم ( به عنوان مثال " open , opens , opened , opening " ) به عنوان کلمات جداگانه . با این حال ، چنین رویکردی امکان پذیر نیست ، چون هر ورودی فرهنگ لغت هزاران فرم ممکن دارد .

## Part-of-speech tagging

با در نظر گرفتن یک جمله ، بخش گفتار ( POS ) را برای هر کلمه مشخص کنید . بسیاری از کلمات ، به ویژه آنها که رایج هستند، می‌توانند به عنوان بخش‌های مختلف سخنرانی عمل کنند . به عنوان مثال ، " کتاب " می‌تواند یک اسم ( " کتاب روی میز " ) یا فعل باشد ؛ مجموعه " می‌تواند یک اسم ، فعل یا صفت باشد ؛ و " بیرون " می‌تواند هر کدام از پنج بخش مختلف سخنرانی باشد . برخی از زبان‌ها ابهام بیشتری نسبت به دیگران دارند . زبان‌ها با مورفولوژی صرفی و مورفولوژی صرفی نظیر انگلیسی ، به طور خاص مستعد چنین ابهام هستند . چین در معرض چنین ابهام قرار دارد ، زیرا این زبان در دوران verbalization زبانی مربوط به تن رنگ است . چنین دستوری به آسانی از طریق هویت‌های درگیر در املا به منظور انتقال مفهوم مورد نظر ، منتقل نمی‌شود .

## Parsing

درخت تجزیه ( تحلیل گرامری ) یک جمله مشخص را تعیین کنید . دستور زبان برای زبان‌های طبیعی مبهم است و جملات معمول چندین تحلیل ممکن دارند . در واقع شاید تعجب آور باشد که برای یک جمله معمولی ممکن است هزاران پتانسیل بالقوه وجود داشته باشد که بیشتر آنها به طور کامل به یک انسان شبیه هستند ) . دو نوع اصلی تجزیه ، تجزیه و تجزیه وابستگی وجود دارد . تجزیه وابستگی بر روابط بین کلمات در یک جمله ( مشخص کردن چیزهایی مانند اشیا اولیه و پیش‌بینی‌کننده ) تمرکز دارد ، در حالی که تجزیه حوزه با استفاده از گرامر مستقل از بافت به ساخت درخت تجزیه می‌پردازد .

## Sentence breaking (sentence boundary disambiguation)

با دادن یک تکه متن ، مرز جمله را پیدا کنید . مرزهای جمله اغلب با دوره یا سایر علائم نقطه‌گذاری علامت‌گذاری می‌شوند ، اما این ویژگی‌ها می‌توانند به اهداف دیگری نیز باشند .

## Stemming

فرآیند ساده سازی مشتق کلمات ( یا گاهی اوقات مشتق شده ) را به فرم ریشه شان تبدیل می‌کند . ( به عنوان مثال " close " ، ریشه برای " closer " ، " close " ، " closing " ، " closed " ) خواهد بود .

## Word segmentation

یک تکه از متن پیوسته را به کلمات جداگانه تقسیم کنید . برای زبانی مانند انگلیسی ، این نسبتاً<sup>۱</sup> بی‌اهمیت است ، زیرا کلمات معمولاً<sup>۲</sup> به وسیله فضاهای از هم جدا می‌شوند . با این حال ، برخی زبان‌های نوشتاری مانند چینی ، ژاپنی و تایلندی مرزهای کلمه را به چنین روشی علامت‌گذاری نمی‌کنند ، و تقسیم‌بندی متن زبان یک وظیفه مهم است که نیاز به دانش لغوی و تکواز شناسی لغات در زبان دارد . گاهی اوقات این فرآیند در مواردی مانند ( Bag of Words ) بکار رفته است .

۱. می‌گیرد .

## **Terminology extraction**

هدف استخراج اصطلاحات ، استخراج خودکار اصطلاحات مرتبط از یک پیکره زبانی است .

## **Lexical semantics**

معنای محاسباتی کلمات فردی در بافت چیست ؟

## **Distributional semantics**

چگونه می توانیم نمایش معنایی را از داده ها یاد بگیریم ؟

## **Machine translation**

ترجمه خودکار متن از یک زبان انسانی به زبان دیگر . این یکی از دشوارترین مشکلات است و یکی از اعضای طبقه ای از مشکلات است که " هوش مصنوعی " نامیده می شود ، یعنی نیاز به انواع مختلف دانشی که انسانها دارند ( دستور زبان ، معانی ، واقعیات در مورد دنیای واقعی و غیره ) به منظور حل صحیح .

## **Named entity recognition (NER)**

با توجه به جریان متن ، مشخص کنید که کدام آیتم‌های موجود در نقشه متن برای نام مناسب ، مانند افراد یا مکان‌ها ، و چه نوع از هر یک از این نام ( به عنوان مثال ، مکان ، سازمان ) . توجه داشته باشید که اگر چه سرمایه‌گذاری بزرگ می‌تواند به تشخیص موجودیت‌های اسمی به زبان‌های انگلیسی کمک کند ، این اطلاعات نمی‌توانند در تعیین نوع نام گذاری شده کمک کنند ، و در هر حال اغلب نادرست یا ناکافی است . برای مثال ، اولین حرف یک جمله به حروف بزرگ اضافه می‌شود ، و entities نام گذاری اغلب چندین کلمه را بیان می‌کنند، که فقط برخی از آن‌ها با حروف بزرگ صحبت می‌شوند . علاوه بر این، بسیاری از زبان‌های دیگر در متون غیر غربی ( به عنوان مثال چینی یا عربی ) اصلاً سرمایه‌گذاری در سرمایه‌گذاری ندارند و حتی زبان‌ها با بزرگ‌نویسی حروف به طور مداوم از آن برای تمایز نام‌ها استفاده نمی‌کنند . به عنوان مثال ، تمام اسم‌ها را بدون توجه به اینکه آن‌ها نام دارند ، و فرانسه و اسپانیایی نام‌هایی که به عنوان صفت بکار برده می‌شوند را ذکر نمی‌کنند.

### Natural language generation

تبديل اطلاعات از پايگاهداده‌های کامپيوتری يا مقاصد معنائي به زبان انسان قابل خواندن.

### Natural language understanding

تکه‌های متن را به نمایش رسمي تری مانند ساختارهای منطقی درجه اول تبدیل کنید که برای برنامه‌های کامپيوتری راحت‌تر دستکاری می‌شوند . درک زبان طبیعی شامل شناسایی معنائی از معانی

چندگانه ممکن است که می‌تواند از یک عبارت زبان طبیعی مشتق شود که معمولاً فرم of سازمان یافته از مفاهیم زبان طبیعی را به خود می‌گیرد . مقدمه و ایجاد زبان هستی‌شناسی و هستی‌شناسی زبان ، با این حال راه حل‌های تجربی کارآمد هستند . رسمی سازی صریح معناشناصی زبان طبیعی بدون سر و کار با مفروضات ضمنی مانند فرض بسته جهان ( CWA ) در مقابل . یک فرض در جهان ، یا " بله " در مقابل . صحیح / اشتباه انتظار می‌رود که ساخت‌وساز مبنای رسمی سازی معنا شناسی باشد .

### **Optical character recognition (OCR)**

با توجه به تصویری که نشان‌دهنده متن چاپی است ، متن مربوطه را تعیین کنید .

### **Question answering**

با در نظر گرفتن یک سوال انسانی ، پاسخ آن را تعیین کنید . سوالات معمول دارای پاسخ صحیح خاص ( مانند " پایخت کانادا " است ؟ " ) ، اما گاهی اوقات سوال‌های بی‌پاسخ نیز در نظر گرفته می‌شوند ( مانند " معنی زندگی چیست ؟ " ) . کارهای اخیر به سوالات پیچیده تری توجه کرده‌اند .

### **Recognizing Textual entailment**

با داشتن دو قطعه متن ، مشخص کنید که آیا یکی از آن‌ها حقیقی است یا نه ، و یا به دیگری اجازه می‌دهد که یا درست یا غلط باشد .

## **Relationship extraction**

یک تکه از متن را با استفاده از یک تکه از متن شناسایی کنید ( به عنوان مثال کسی که با چه کسی ازدواج کرده است ) .

## **Sentiment analysis (multimodal sentiment analysis)**

استخراج اطلاعات ذهنی از مجموعه‌ای از اسناد ، اغلب با استفاده از بازبینی‌های آنلاین برای تعیین "پلاریته" در مورد اشیا خاص . این به ویژه برای شناسایی روند افکار عمومی در رسانه‌های اجتماعی ، به منظور بازاریابی مفید است.

## **Topic segmentation and recognition**

با استفاده از یک تکه از متن ، آن را به بخش‌هایی تقسیم کنید که هر کدام از آنها به یک موضوع اختصاص دارد و موضوع بخش را شناسایی می‌کند .

## **Word sense disambiguation**

کلمات بسیاری بیش از یک معنا دارند ; ما باید معنایی را انتخاب کنیم که بیشترین احساس را در متن داشته باشد. برای این مشکل ، به طور معمول فهرستی از کلمات و معانی لغات مرتبط ، از یک فرهنگ لغت یا یک منبع آنلاین مانند وردنت ارائه شده است .

## Automatic summarization

یک خلاصه خوانا از یک تکه از متن را تولید کنید . اغلب برای ارایه خلاصه‌ای از متن یک نوع شناخته شده ، مانند مقالات پژوهشی ، مقالات در بخش مالی یک روزنامه ، به کار می‌رود .

## Coreference resolution

با توجه به یک جمله یا قسمت بزرگتر متن ، مشخص کنید که کدام کلمه ( " اشاره " ) به همان اشیا اشاره دارد ( " entities " ) . تفکیک پذیری یک مثال خاص از این وظیفه است و به طور خاص مربوط به تطابق up با اسم‌ها یا نام‌هایی است که آن‌ها به آن اشاره می‌کنند . وظیفه کلی تفکیک پذیری نیز شامل مشخص کردن اصطلاح " روابط اتصالی " است که شامل عبارات مربوط به آن می‌شود . برای مثال ، در یک جمله از جمله " او وارد خانه John's از طریق در جلویی شد " ، " در جلویی " عبارت است از بیان و رابطه پل زنی که باید شناسایی شود این واقعیت است که در ورودی خانه front ( به جای برخی از ساختار دیگری که ممکن است به آن ارجاع داده شود ) است .

## Discourse analysis

این عنوان شامل تعدادی از وظایف مرتبط است . یک وظیفه شناسایی ساختار گفتمان متن متصل ، یعنی ماهیت روابط گفتمان بین جملات ( به عنوان مثال ، توضیح ، کنtraست ) است . یک

وظیفه احتمالی دیگر تشخیص و طبقه‌بندی فعالیت‌های گفتاری در یک تکه از متن (به عنوان مثال به نه سوال، پرسش محتوا، بیانیه، بیانیه وغیره) است.

### **Speech recognition**

با توجه به کلیپ صوتی فرد یا افراد، نمایش متن سخنرانی را مشخص کنید. این نقطه مقابل متن است و یکی از مشکلات بسیار دشوار است که در اصطلاح "هوش مصنوعی" نامیده می‌شود. در گفتار طبیعی، به ندرت بین کلمات متوالی مکث وجود دارد، و در نتیجه بخش‌بندی گفتار، نشانه ضروری تشخیص گفتار است. توجه داشته باشید که در بیشتر زبان‌های گفتاری، صدای‌هایی که نشان‌دهنده حروف متوالی با یکدیگر در یک فرآیند موسوم به نویز هستند، بنابراین تبدیل سیگنال آنالوگ به کاراکترهای گستته می‌تواند یک فرآیند بسیار دشوار باشد. همچنین با توجه به اینکه کلمات در زبان مشابه توسط افراد با لهجه‌های مختلف بیان می‌شوند، نرم‌افزار تشخیص گفتار باید قادر به تشخیص انواع مختلف ورودی به صورت یکسان با یکدیگر از لحاظ متنی باشد.

### **Speech segmentation**

با استفاده از یک کلیپ صوتی از یک فرد و یا افراد صحبت کردن، آن را به کلمات تقسیم کنید. A به رسمیت شناختن گفتار و به طور معمول با آن گروه‌بندی می‌شوند.

### **Text-to-speech**

با استفاده از یک متن ، آن واحدها را تغییر داده و یک نمایش گفتاری را تولید کنید . می‌تواند برای کمک به اختلالات بینایی استفاده شود.

## Dialogue

اولین کار منتشر شده توسط یک هوش مصنوعی در سال ۲۰۱۸ ، با نام "[خیابان یک](#)" ، که به عنوان رمانی به بازار عرضه شد ، شامل شصت میلیون کلمه است .

## Chatbot

یک برنامه کامپیوتری است که مکالمه را از طریق روش‌های شنیداری یا متنی انجام می‌دهد . این برنامه‌ها اغلب برای شبیه‌سازی چگونگی رفتار یک انسان به عنوان یک شریک صحبت طراحی شده‌اند ، اگرچه تا سال ۲۰۱۹ قادر به گذراندن آزمون تورینگ نیستند . معمولاً در سیستم‌های تبادل برای اهداف کاربردی مختلف از جمله خدمات مشتری یا کسب اطلاعات استفاده می‌شوند . برخی از chatbot ها از سیستم‌های پردازش زبان طبیعی پیچیده استفاده می‌کنند ، اما بسیاری از آن‌ها برای کلمات کلیدی درون ورودی جستجو می‌کنند ، سپس با بیشترین کلمات کلیدی یا بهترین الگو ، از یک پایگاهداده پاسخ می‌دهند .

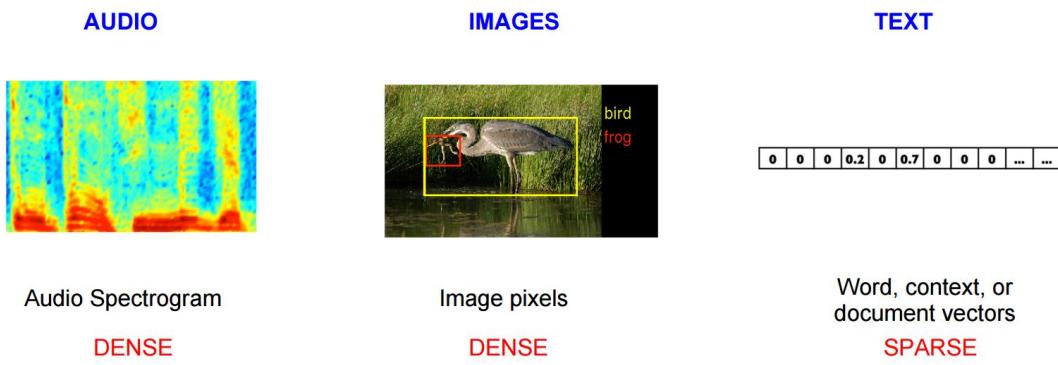
واژه " chatterbot " در اصل توسط مایکل Mauldin ( سازنده اولین Verbot ، جولیا ) در سال ۱۹۹۴ برای توصیف این برنامه‌ها ابداع شد [1]. امروزه ، اغلب chatbots از طریق دستیاران مجازی مانند Google Assistant و الکس تیو ، از طریق برنامه‌های پیامرسانی مانند فرستاده فیس بوک یا WeChat ، یا از طریق برنامه‌ها و وب سایت‌ها از طریق برنامه‌های پیامرسانی به دست می‌آیند . Chatbot ها را می‌توان به دسته‌های کاربرد مانند تجارت مکالمه ( تجارت الکترونیکی از طریق گپ ) ، تجزیه و تحلیل ، ارتباطات ، پشتیبانی مشتری ، طراحی ، توسعه ، بهداشت ، منابع انسانی ، بازاریابی ، اخبار ، بازاریابی ، اخبار ، امور شخصی ، بازاریابی ، ورزشی ، سفر و خدمات طبقه‌بندی کرد [2] . علاوه بر این ، هوش مصنوعی محاوره‌ای به استفاده از برنامه‌های پیامرسانی ، دستیاران مبتنی بر گفتار و پردازشگر برای خودکارسازی ارتباطات و ایجاد تجربیات شخصی شخصی در مقیاس اشاره دارد.

## نمایش برداری کلمات (Vector Representations of Words)

انگیزه : چرا نمایش برداری کلمات را یاد بگیرید ؟

سیستم‌های پردازش تصویر و تصویری با مجموعه داده‌های غنی و با ابعاد بالا به صورت بردارهای شدت پیکسل خام برای داده‌های تصویری و یا ضریب چگالی طیفی قدرت طیفی برای داده‌های صوتی کار می‌کنند . برای وظایفی مانند تشخیص شی یا گفتار می‌دانیم که تمام اطلاعات مورد نیاز برای انجام

موفقیت‌آمیز کار در داده‌ها کدگذاری می‌شوند ( زیرا انسان‌ها می‌توانند این وظایف را از داده‌های خام انجام دهند ) . با این حال ، سیستم‌های پردازش زبان طبیعی به طور سنتی با کلمات به عنوان نمادهای آنکه گسته رفتار می‌کنند و بنابراین " گربه " ممکن است به صورت " Id537 " و " سگ " به صورت " Id143 " نمایش داده شود . این بدان معنی است که مدل می‌تواند از چیزی که در مورد " گربه‌ها " یاد گرفته استفاده کند زمانی که اطلاعات مربوط به " سگ " را پردازش می‌کند ( به طوری که هر دو حیوان ، چهار پا و حیوان خانگی هستند ) . تشخیص کلمات به صورت منحصر به فرد و مجزا منجر به پراکندگی داده‌ها می‌شود و معمولاً به داده‌های بیشتری نیاز دارد تا مدل‌های آماری را با موفقیت آموزش دهند . استفاده از نمایش‌های برداری می‌تواند بر برخی از این موانع غلبه کند .



مدل‌های فضای برداری ( VSMs [3] ) عبارت هستند از کلمات در یک فضای برداری مستمر که در آن کلمات مشابه به لحاظ معنایی به نقاط نزدیک نگاشته می‌شوند ( در نزدیکی یکدیگر قرار گرفته‌اند در آن کلمات مشابه به ساخته طولانی و غنی در NLP هستند ، اما همه روش‌ها به طریقی وابسته به فرضیه VSMs . ) . VSMs هستند ، که بیان می‌کند که کلماتی که در زمینه‌های مشابه ظاهر می‌شوند معنای distributional

معنایی دارند . رویکردهای متفاوتی که از این اصل استفاده می‌کنند را می‌توان به دو دسته تقسیم کرد : روش‌های شمارش دستی ( به عنوان مثال ، آنالیز معنایی ) ، و روش‌های پیش‌گویانه ( به عنوان مثال مدل‌های زبان احتمالاتی [4] ) .

این قایز به طور خلاصه توسط Baroni [5] و همکاران به تفصیل شرح داده می‌شود ، اما به طور خلاصه : روش‌های مبتنی بر کنت ، آمار چگونگی رخ دادن چند کلمه را با کلمات همسایه خود در یک پیکره زبانی بزرگ محاسبه می‌کنند و سپس این آمار را برای یک بردار کوچک و متراکم برای هر کلمه انتخاب می‌کنند . مدل‌های پیش‌گویانه به طور مستقیم سعی می‌کنند یک کلمه از همسایگانشان را از نظر بردارهای کوچک و متراکم ( که پارامترهای مدل را در نظر می‌گیرند ) پیش‌بینی کنند .

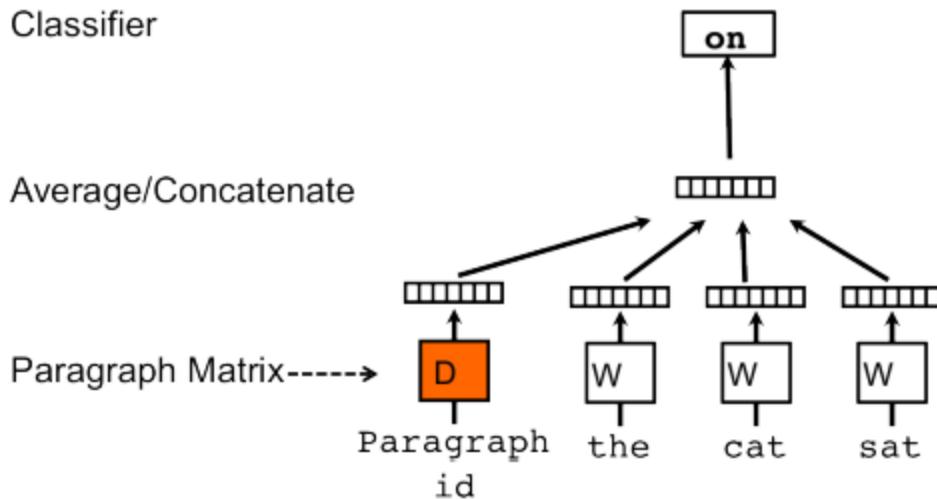
Word2Vec یک مدل پیش‌بینی‌کننده کارآمد از نظر محاسباتی برای یادگیری کلمه تعبیه از متن خام است . در سال ۲۰۱۳ ارائه شد قصد دارد به شما این را به شما بدهد : یک نمایش عددی برای هر کلمه ، که می‌تواند چنین روابطی را در بالا ثبت کند . این بخشی از مفهوم وسیع‌تر یادگیری ماشین - بردارهای ویژگی . Word2Vec در دو زمرة است ، مدل پیوسته از کلمات ( CBOW ) و مدل skip-gram ( بخش ۳,۱ و ۳,۲ در Mikolov و همکاران [6] ) . به صورت الگوریتمی ، این مدل‌ها مشابه هستند ، به جز اینکه CBOW کلمات هدف را از کلمات متن منبع پیش‌بینی می‌کند ( به عنوان مثال ، cat روی صفحه می‌نشیند ) ، در حالی که skip-gram عکس معکوس است و متن مبدا - کلمات را از کلمات هدف پیش‌بینی می‌کند . این وارونگی ممکن است به نظر یک انتخاب دلخواه به نظر برسد ،

اما از نظر اماری تاثیر آن بر مقدار زیادی از اطلاعات توزیعی به دست می‌آید ( با اضافه کردن کل متن به عنوان یک مشاهده ) . در اکثر موارد ، این تبدیل به یک چیز مفید برای مجموعه داده‌های کوچک‌تر می‌شود . با این حال ، skip-gram با هر جفت "موضوع - هدف" به عنوان یک مشاهده جدید برخورد می‌کند ، و این کار زمانی که مجموعه داده‌های بزرگ‌تر داریم ، بهتر عمل می‌کند .

## Doc2vec

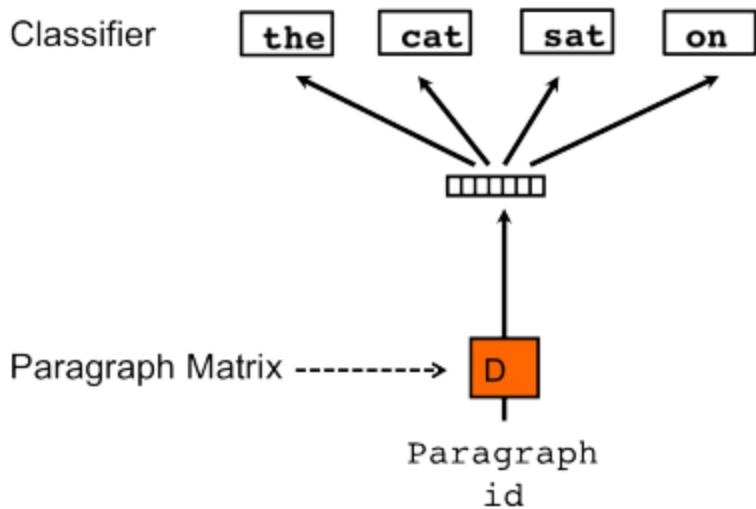
پس از درک این مطلب که Word2Vec چیست ، درک چگونگی کار Doc2Vec آسان‌تر خواهد بود . همانطور که گفته شد ، هدف Doc2Vec ایجاد یک نمایش عددی از یک سند بدون در نظر گرفتن طول آن است . اما بر خلاف کلمات ، اسناد در ساختارهای منطقی مثل کلمات نیستند ، بنابراین روش دیگری باید پیدا شود .

مفهومی که Mikilov و لو از آن استفاده کردند ساده است ، با این حال هوشمندانه : آنها از مدل word2vec استفاده کردند ، و یک بردار دیگر ( شناسه بند را در زیر ) اضافه کردند :



اگر با طرح بالا آشنا هستید ، به این دلیل است که یک بسط کمی برای مدل CBOW وجود دارد . اما به جای استفاده از کلمات فقط برای پیش‌بینی کلمه بعدی ، یک بردار ویژگی دیگر نیز اضافه کردیم ، که مستند است . بنابراین هنگامی که کلمه بردارهای  $W$  را آموزش می‌دهیم ، بردار مدرک  $D$  نیز آموزش داده می‌شود ، و در پایان آموزش ، یک نمایش عددی از سند را در دست دارد .

مدل فوق ، نسخه حافظه توزیع شده بردار بند ( PV - DM ) نامیده می‌شود . این به عنوان یک حافظه عمل می‌کند که آنچه را از بافت فعلی یا به عنوان موضوع بند از قلم افتاده را به یاد می‌آورد . در حالی که کلمه بردار مفهوم یک کلمه را نشان می‌دهد ، بردار مدرک قصد دارد مفهوم یک سند را نشان دهد . یک الگوریتم دیگر ، که شبیه skip-gram است ، ممکن است از بسته توزیع شده کلمات بردار بند ( PV - DBOW ) استفاده شود .



در اینجا ، این الگوریتم در واقع سریع‌تر بوده و حافظه کمتری مصرف می‌کند ، زیرا نیازی به ذخیره بردارهای کلمه وجود ندارد . مولفان پیشنهاد می‌کنند که از ترکیبی از هر دو الگوریتم استفاده کنند ، هر چند که مدل  $pv - dm$  بهتر است و معمولاً<sup>۱</sup> به حالت ایده آل نیز منتج می‌شود .

مدل‌های رگرسیون ممکن است به روش زیر مورد استفاده قرار گیرند : برای آموزش ، مجموعه‌ای از اسناد مورد نیاز است . یک بردار کلمه برای هر کلمه تولید می‌شود و یک بردار سند برای هر سند تولید می‌شود . همچنین مدل وزن‌های یک لایه پنهان را آموزش می‌دهد . در مرحله استنباطی ، یک سند جدید ارائه می‌شود و تمام وزن‌ها برای محاسبه بردار سند ثابت هستند .

## NLP پروژه

این پروژه شامل دو فاز است که هر یک به خودی خود ساختار کامل یک پروژه پردازش زبان طبیعی را دارد.

### فاز اول : تولید زبان طبیعی

این فاز سعی در خلق یک متن را با استفاده از یک مدل مبتنی بر کاراکتر مدنظر دارد که با مجموعه داده از نوشهای شکسپیر و با بهره‌گیری از شبکه‌های عصبی بازگشتی پیاده سازی شده است. با توجه به دنباله‌ای از کاراکترهای این داده‌ها، یک مدل برای پیش‌بینی کاراکتر بعدی در توالی (e) وجود دارد. توالی‌های طولانی‌تر می‌توانند با فراخوانی دوباره مدل ایجاد شود.

در این فاز به طور پیشفرض از CPU به عنوان پردازنده بهره گرفته شده، ولیکن برای استفاده GPU نیز پیاده سازی صورت گرفته که در صورت وجود بسیار در سرعت آموزش تاثیر خواهد داشت.

در این فاز برای پیاده سازی شبکه های عصبی بازگشتی از کتابخانه های tensorflow و keras بهره گرفته ایم .

Word Embedding یا گنجاندن کلمات ، نام جمعی مجموعه ای از تکنیک های مدل سازی زبانی و تکنیک های یادگیری در پردازش زبان طبیعی (NLP) است که در آن کلمات یا عباراتی از واژگان برای بردارهای اعداد حقیقی نگاشته می شوند . از نظر مفهومی ، شامل گنجاندن ریاضی از فضا با ابعاد بسیار در هر کلمه برای یک فضای برداری پیوسته با بعد بسیار پایین تر است . برای هر کاراکتر مدل timestamp آن را نگاه می کند ، شبکه عصبی حافظه بلند مدت را با embedding به عنوان ورودی اجرا می کند و سعی می کند حدس بزند احتمال کاراکتر بعدی رو بر اساس log-likelihood :

حروف :

## References

- [1] [Online]. Available: <http://www.aaai.org/Library/AAAI/aaai94contents.php>.
- [2] [Online]. Available: <https://cai.tools.sap/blog/2017-messenger-bot-landscape/>.
- [3] [Online]. Available: [https://en.wikipedia.org/wiki/Vector\\_space\\_model](https://en.wikipedia.org/wiki/Vector_space_model).
- [4] [Online]. Available: [http://www.scholarpedia.org/article/Neural\\_net\\_language\\_models](http://www.scholarpedia.org/article/Neural_net_language_models).
- [5] [Online]. Available: [https://en.wikipedia.org/wiki/Vector\\_space\\_model](https://en.wikipedia.org/wiki/Vector_space_model).

در حقیقت، اندرکنش بین انسان‌ها و ماشین‌ها در هنگام استفاده از پردازش زبان طبیعی، می‌تواند

به صورت زیر تعریف شود:

1. یک انسان با ماشین صحبت می‌کند.
2. ماشین صدای فرد را ثبت می‌کند.
3. صدا به نوشته و متن تبدیل می‌شود.

۴. پردازش بر روی داده های متنی انجام می شود.

۵. داده های متنی به صوت تبدیل می شود.

۶. ماشین به انسان با اجرا کردن فایل صوتی پاسخ می دهد.

کاربرد NLP چیست؟

زبان پردازش طبیعی می تواند در موارد زیر پرکاربرد باشد:

- در اپلیکیشن های مترجم مانند Google Translate

- پردازشگرهای کلمه مانند برنامه Microsoft Word و یا سرویس آنلاین Grammarly که از

پردازش زبان طبیعی جهت چک کردن دقیق و صحت گرامر در متون مختلف بهره می برند.

- اپلیکیشن های پاسخ گفتاری متقابل (Interactive Voice Response (IVR)) که در مراکز تماس

جهت پاسخ به درخواست های کاربران مشخص مورد استفاده قرار می گیرد.

- اپلیکیشن های دستیار شخصی (Personal Assistant) مانند Siri ، OK Google و Cortana

Alexa

به چه دلیل NLP، به عنوان یک فناوری سخت شناخته می شود؟

در علوم کامپیوتر، پردازش زبان طبیعی، به عنوان یک برنامه سخت شناخته می شود. این طبیعت زبان

انسان است که منجر به سخت شدن NLP شده است.

قوانينی که بر انتقال اطلاعات توسط زبان های طبیعی حاکم است، به سادگی نمی توانند توسط کامپیوترها درک شوند. برخی از این قوانین، می توانند بسیار پیچیده و سطح بالا باشند. به عنوان مثال، زمانی که یک فرد از بیان طعنه آمیز جهت انتقال اطلاعات استفاده می کند. از جهت دیگر، بعضی از این قوانین می توانند سطح پایین باشند. به عنوان مثال، استفاده از کاراکتر «s» برای نشان دادن جمع بودن یک کلمه.

برای آن که بتوان به یک فهم جامع از زبان انسان دست پیدا کرد، نیاز است که هم کلمات به خوبی شناخته شوند و هم مفهومی که برای رساندن یک پیام مشخص به یکدیگر متصل شده اند، به خوبی درک شود.

در حالی که انسان ها می توانند به سادگی یک زبان را یاد بگیرند، حرف های غیر دقیق و مبهم زبان های طبیعی باعث سخت شدن اجرای NLP توسط ماشین ها می شوند.



پردازش زبان طبیعی چگونه کار می کند؟

الگوریتم هایی را اعمال می کند که باعث تعریف شدن و نشان دادن قوانین زبان طبیعی می شوند. این قوانین به گونه ای تعریف می شوند که داده های بی ساختار زبان، به شکلی تبدیل می شود که قابل فهم برای کامپیوتر باشد.

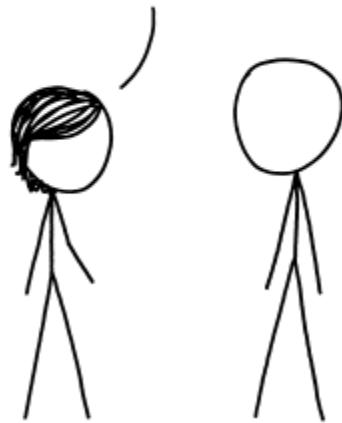
زمانی که متنی به کامپیوتر داده می شود، کامپیوتر الگوریتم هایی را برای خارج کردن معانی هر جمله به کار برد و داده های حیاتی آن ها را جمع آوری می کند. گاهی اوقات، کامپیوتر نمی تواند معنی یک جمله را به خوبی درک کند که این خود منجر به ارائه نتایج مبهم می شود. به عنوان مثال، می توان به اتفاق خنده داری که در سال ۱۹۵۰ در هنگام ترجمه بعضی از کلمات انگلیسی به زبان روسی رخ داد، اشاره کرد.

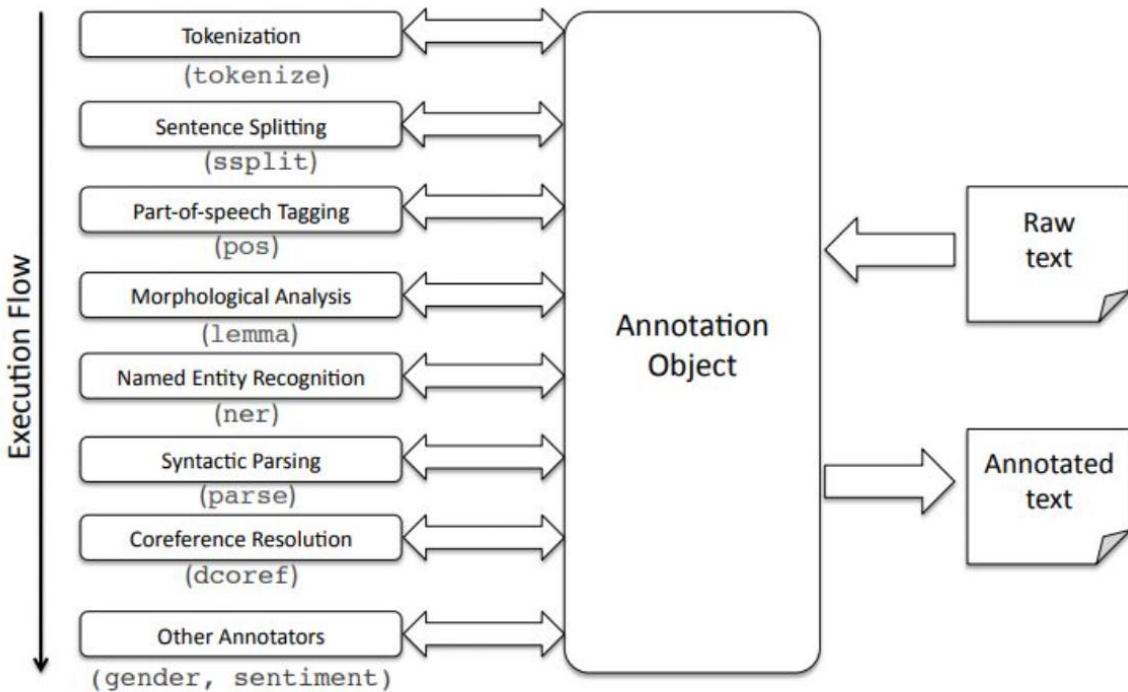
چه تکنیک هایی در NLP مورد استفاده قرار می گیرد؟

آنالیز نحوی (Syntactic) و معنایی (Semantic)، اصلی ترین تکنیک های کامل کننده وظایف پردازش زبان طبیعی هستند. پردازش زبان طبیعی (NLP) به بهره گیری از ابزارها، تکنیکها و الگوریتمها برای پردازش و درک داده های طبیعی مبتنی بر زبان مربوط است که معمولاً در قالب های ساخت نیافته ای مانند متن، سخنرانی و غیره وجود دارند. در نهایت، حوزه های تخصصی در علوم رایانه و هوش مصنوعی محسوب می شود که ریشه در زبانشناسی محاسباتی دارد. دغدغه اصلی این حوزه از علوم، طراحی و ساخت برنامه و سیستم هایی است که امکان تعامل بین ماشین ها و زبان های طبیعی را فراهم

سازند و در طی زمان برای استفاده انسان تکامل پیدا کنند. از این رو در اغلب موارد این حوزه علمی به عنوان یک زمینه کم عمق و سطحی برای تحقیق نگریسته می شود و افراد تمایل دارند که بیشتر روی یادگیری ماشین و یادگیری آماری تمرکز کنند.

I DON'T MEAN TO GO ALL LANGUAGE NERD ON YOU, BUT I JUST LEGIT ADVERBED "LEGIT," VERBED "ADVERB," AND ADJECTIVED "LANGUAGE NERD."





در تصویر فوق با جریان اجرای مراحل NLP بر روی متن آشنا می شویم.

NLG

یک فرآیند نرم افزاری است که داده های ساختار یافته را به محتوای ساده انگلیسی تبدیل می کند . آن می تواند برای تولید محتوای فرم طولانی برای سازمان ها جهت خودکار کردن گزارش های سفارشی ، و همچنین تولید محتوای سفارشی برای یک وب یا کاربرد تلفن همراه استفاده شود . همچنین می توان از آن برای تولید blurb کوتاه متن در مکالمات تعاملی استفاده کرد ( a ) که ممکن است حتی با صدای بلند توسط یک سیستم متن به سخنرانی خوانده شود .

NLG خودکار می‌توانند در مقایسه با فرآیند استفاده از انسان‌ها هنگام تبدیل ایده‌ها به نوشت و یا سخن گفتن، مقایسه شوند . Psycholinguists ، تولید زبان را برای این فرآیند ترجیح می‌دهند، که هم چنین می‌تواند در اصطلاحات ریاضی توصیف شود ، یا در یک کامپیوتر برای تحقیقات روانشناسی decompilers مدلسازی شود . سیستم‌های NLG نیز می‌توانند با مترجمان زبان رایانه‌ای مصنوعی مثل transpilers یا می‌کنند . زبان‌های انسانی به طور قابل توجهی پیچیده‌تر هستند و ابهام بیشتر و تنوع بیان را نسبت به زبان‌های برنامه‌نویسی فراهم می‌کنند که باعث می‌شود چالش برانگیزتر باشد .

NLG ممکن است به عنوان متضاد درک زبان طبیعی در نظر گرفته شود : در حالی که در درک زبان طبیعی ، سیستم باید جمله ورودی را برای تولید زبان نمایش ماشین خنثی کند ، در NLG سیستم باید NLU در مورد چگونگی قرار دادن یک مفهوم در کلمات تصمیم‌گیری کند . ملاحظات عملی در ساخت NLU در مقابل . سیستم‌های NLG متقارن نیستند . NLU باید با ورودی کاربر مبهم یا اشتباه سروکار داشته باشد ، در حالی که ایده‌های سیستم می‌خواهد از طریق NLG به طور کلی به طور دقیق مشخص شوند . NLG باید نمایش متنی خاص و سازگار از نمایش‌های بالقوه را انتخاب کند ، در حالی که NLG به طور کلی تلاش می‌کند تا یک نمایش منفرد و عادی از این ایده را تولید کند . برای مدتی NLU طولانی وجود داشته است. اما اخیراً تکنولوژی NLG تجاری اخیراً در دسترس عموم قرار گرفت . این تکنیک‌ها از سیستم‌های ساده الگوی مبتنی بر الگو مانند یک ترکیب نامه استفاده می‌کنند که حروف

شکل را تولید می‌کند، به سیستم‌هایی که درک پیچیده‌ای از دستور زبان انسانی دارند. NLG همچنین می‌تواند با آموزش یک مدل آماری با استفاده از یادگیری ماشین، به طور معمول در مجموعه بزرگی از متون نوشتاری انسان انجام شود.

فرآیند تولید متن می‌تواند به سادگی نگه داشتن یک لیست از متن کنسرو شده باشد که کپی و چسبانده شده و احتمالاً با برخی متن چسب مرتبط است. نتایج ممکن است در دامنه‌های ساده از قبیل ماشین‌های horoscope و یا ژنراتور نامه‌های شخصی رضایت‌بخش باشد. با این حال، یک سیستم NLG پیچیده باید شامل مراحل برنامه‌ریزی و ادغام اطلاعات باشد تا تولید متنی که طبیعی به نظر می‌رسد و تکراری نشود. مراحل معمول تولید زبان طبیعی، همانطور که توسط Reiter و پیشنهاد شد، عبارتند از: تعیین محتوا: تصمیم‌گیری در مورد چه اطلاعاتی در متن. به عنوان مثال، در مثال گردد در بالا، تصمیم‌گیری درباره اینکه آیا باید صراحتاً اشاره کنیم که سطح گردد در جنوب شرقی است یا نه. ساختاربندی اسناد: یک سازمان کلی از اطلاعات برای انتقال. به عنوان مثال، تصمیم‌گیری برای توصیف مناطق با سطح گردد بالا اول، به جای مناطقی با سطح pollen پایین. تجمع: ادغام جملات مشابه برای بهبود خوانایی و طبیعی بودن. به عنوان مثال، ادغام دو جمله زیر: سطوح گردد علف برای جمعه از سطح متوسط تا بالا در اکثر بخش‌های کشور تا ۷ به ۷ افزایش پیدا خواهد کرد: سطوح گردد علف برای جمعه از متوسط تا سطوح بالای روز گذشته با مقادیر حدود ۶ تا ۷ در اغلب بخش‌های کشور افزایش پیدا کرده است.

گزینه Lexical : قرار دادن کلمات به مفاهیم . برای مثال ، تصمیم‌گیری در مورد اینکه آیا متوسط یا متوسط باید در هنگام توصیف سطح گردد از ۴ مورد استفاده قرار گیرد .

ایجاد حالات اشاره . ایجاد حالات اشاره که اشیا و مناطق را شناسایی می‌کنند . برای مثال ، تصمیم‌گیری در مورد استفاده در جزایر شمالی و شمال شرقی اسکاتلند برای اشاره به منطقه‌ای خاص در اسکاتلند . این وظیفه همچنین شامل تصمیم‌گیری در مورد pronouns و انواع دیگر of است .

ادراک : ایجاد متن واقعی ، که باید مطابق با قواعد نحو ، مورفولوژی ، و املاء صحیح باشد . به عنوان مثال ، استفاده از آن برای آینده پرتنش آینده خواهد بود .

یک رویکرد دیگر برای NLG استفاده از یادگیری ماشین به انتها " برای ساخت یک سیستم بدون داشتن مراحل مجزا در بالا است . به عبارت دیگر ، ما یک سیستم NLG را با آموزش الگوریتم یادگیری ماشین ( اغلب یک LSTM ) بر روی یک مجموعه داده بزرگ از داده‌های ورودی و متون خروجی ( human ) می‌سازیم . شاید رویکرد نهایی در تصویر captioning موفق بوده ، که به طور خودکار عنوان متنی را برای یک تصویر ایجاد می‌کند .

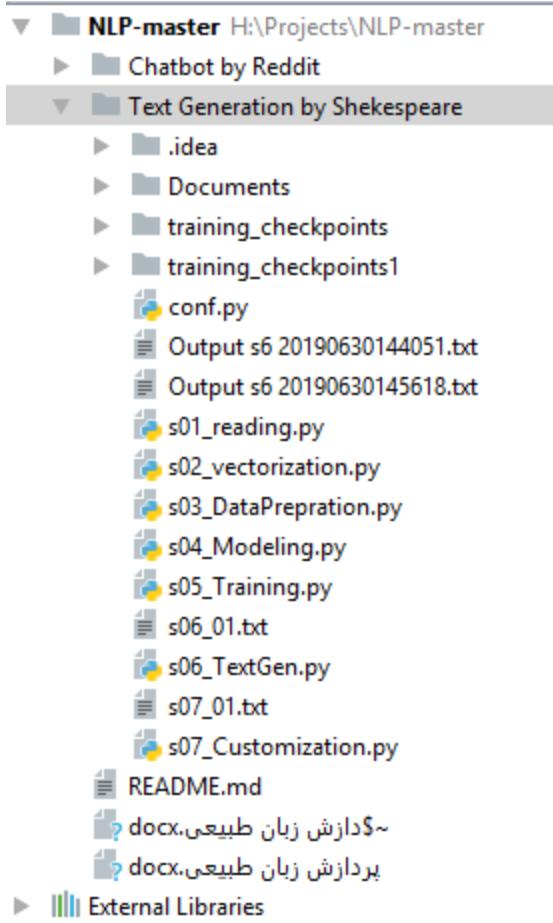
## ابعاد پژوهش

پژوهش حاضر شامل دو شاخه یا پروژه کاملاً جداگانه می باشد یکی پروژه Text generation و دیگری پروژه Chatbot. در پروژه اول برای پیاده‌سازی اجزای پروژه و شبکه‌های عصبی بازگشتی از کتابخانه‌های TensorFlow و Numpy بهره گرفته ایم و روی پایتون 3.6 پیاده سازی شده است (گرچه احتمالاً با سایر ورژن‌ها نیز همخوانی دارد) و در پروژه دوم برای پیاده‌سازی شبکه‌های عصبی بازگشتی از کتابخانه TensorFlow شرکت گوگل، برای پیش‌پردازش داده‌ها از کتابخانه‌های Pandas و Numpy و برای کار با فایل از کتابخانه‌های os و pickle استفاده شده است.

## پروژه تولید متن شکسپیر

این پروژه به تولید متن با استفاده از RNN مبتنی بر ویژگی Word2Vec پرداخته است. به عنوان مجموعه داده‌ها از نوشته‌های شکسپیر استفاده شده است. شبکه‌های عصبی LSTM و GRU پیاده سازی شده است.. سعی شده با در نظر گرفتن کل متن به عنوان داده آموزش، ادامه گفتمان‌ها استخراج گردد. توالی بیشتر متن می‌تواند با فراخوانی مجدد مدل تولید شود.

توجه : این پروژه به گونه‌ای پیاده‌سازی شده است که هم با CPU و هم با GPU می‌تواند به عنوان پردازنده اجرا گردد و به علت نوع پروژه و استفاده از فضاهای برداری استفاده از GPU می‌تواند سرعت اجرا را بسیار بالا ببرد.



این پروژه از چندین فایل تشکیل شده است .

فایل conf.py تنظیمات کلی برنامه را شامل می‌شود

فایل s01\_reading.py به خوانش فایل و اصلاحات

unicode و ساختن دایره لغات برنامه می‌پردازد. به

این صورت که تمامی لغات موجود در متن را در

استخراج و به عنوان دایره المعارف داخل یک لیست

نگه می‌دارد.

گام بعدی s02\_vectorization است که به تخصیص

یک عدد به هر کارکتر و ساختن بردار از روی کلمات

اشاره دارد. قبل از آموزش ، ما نیاز به نقشه‌ای برای یک نمایش عددی داریم . ایجاد دو جدول جستجو :

یکی از نویسه‌های نقشه‌برداری به اعداد ، و دیگری برای اعداد به حروف.

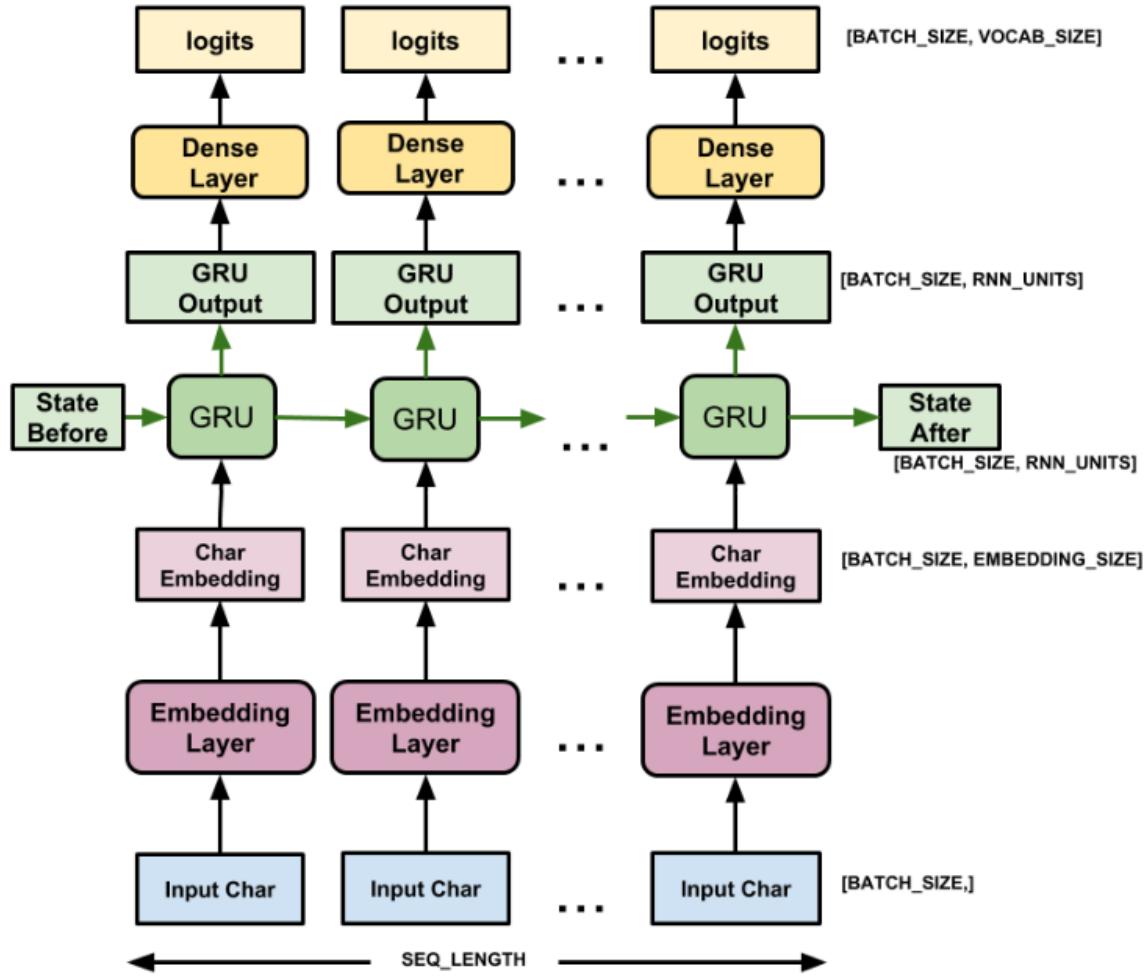
گام بعدی s03\_DataPreparation است که سپس متن را به عنوان توالی نمونه تقسیم کنید . هر توالی

وروودی شامل نویسه‌های طول توالی از متن خواهد بود . برای هر توالی ورودی ، اهداف مربوطه شامل

طول پنجره یکسان هستند ، به جز یک کاراکتر به سمت راست بنا براین ، متن را به صورت تکه های به طول ۱ + ۱ بشکنید . به عنوان مثال ، می گویند که طول پنجره ۴ است و متن ما " سلام " است . توالی from\_tensor\_slices و توالی هدف " ello " خواهد بود . برای انجام این کار ابتدا از تابع برای تبدیل بردار متن به جریان شاخص های کاراکتر استفاده می کنیم . از روی بردارهای ساخته شده در مرحله قبل به ساختن دیتا است می پردازد و داده ها را به صورت target data و input data برای مدل سازی آماده می کند .

گام چهارم و مهم s04\_Modeling نام دارد که با توجه به تنظیمات داخل فایل conf به ایجاد مدل شبکه عصبی بر روی داده ها می کند . با داشتن یک شخصیت ، یا یک رشته از شخصیت ها ، محتمل ترین شخصیت بعدی چیست ؟ این کاری است که ما برای انجام آن به مدل آموزش می دهیم . ورودی مدل یک دنباله از کاراکتر خواهد بود ، و ما مدل را آموزش می دهیم تا خروجی را پیش بینی کنیم - کاراکتر زیر در هر مرحله زمانی از آنجا که rnns یک حالت داخلی را حفظ می کنند که به عناصر مشاهده شده قبلی بستگی دارد ، با توجه به تمام شخصیت های محاسبه شده تا این لحظه ، شخصیت بعدی چیست ؟

برای این مثال ساده از سهلایه برای تعریف مدل استفاده می شود :



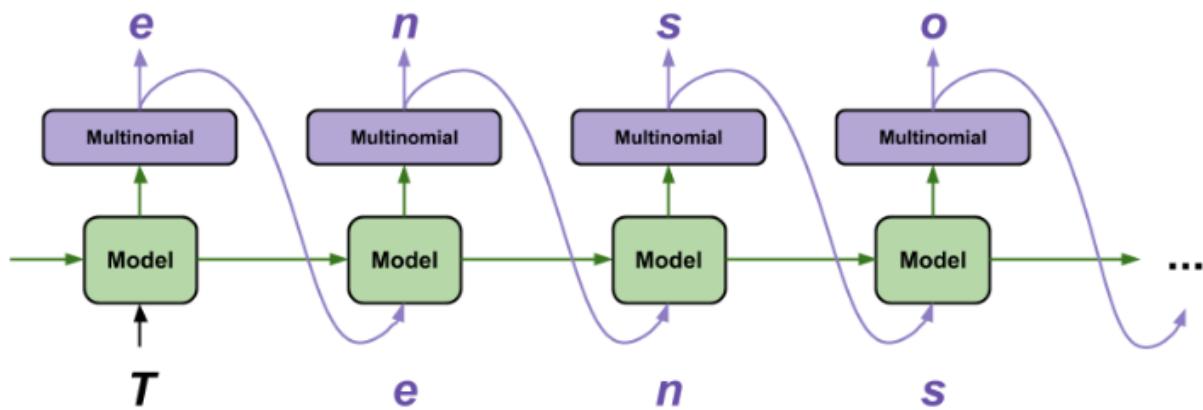
گام آموزش s05\_Training، لایه ورودی، لایه بازگشتی با تعداد نود مشخص شده در فایل conf و لایه خروجی. برای هر کاراکتر مدل embedding را نگاه می‌کند، واحد بازگشتی متشکل از معماری GRU را با embedding به عنوان ورودی اجرا می‌کند برای پیش‌بینی لگاریتم درستنمایی(log-likelihood) کاراکتر بعدی بکار می‌برد. این گام با توجه به تعداد Epoch تنظیم شده در فایل conf زمانبُر خواهد بود و خروجی آن یک شبکه آموزش دیده با وزن‌های مناسب است که در فایل training\_checkpoints آمده است.

ذخیره می‌گردد. دی‌اینجا برای صرفه‌جویی در زمان برای تحویل پروژه یک مدل پیش‌آموزش دیده با epoch 10 موجود و آماده بهره‌برداری می‌باشد.

در نهایت TextGen\_06 به تکرار پیش‌بینی و تولید متن می‌پردازد. بلوک کد زیر متن را تولید می‌کند:

```
سروع با انتخاب یک رشته آغاز، راهاندازی حالت اولیه و تنظیم تعداد کاراکترها برای تولید می‌کند.
توزیع پیش‌بینی کاراکتر بعدی با استفاده از سیم شروع و حالت پایا. از یک توزیع چندجمله‌ای برای محاسبه شاخص شخصیت پیش‌بینی شده استفاده کنید. از این کاراکتر پیش‌بینی شده به عنوان ورودی بعدی مدل استفاده کنید.
```

این مدل به مدل بازگشته است به طوری که اکنون بافت بیشتری دارد، به جای یک کلمه. بعد از پیش‌بینی کلمه بعدی، حالت‌های تغییر یافته دوباره به مدل داده می‌شوند، که این است که چگونه یاد می‌گیرد که بافت بیشتری از کلمات پیش‌بینی شده قبلی می‌گیرد.



آسان‌ترین راه برای داشتن پیش‌بینی‌های بهتر بزرگ نمودن تعداد Epoch در اجرا است. ولیکن تعداد Epoch با زمان اجرای آموزش رابطه مستقیم دارد. همچنین می‌توانید با یک رشته شروع متفاوت آزمایش کنید، و یا اضافه کردن یک لایه RNN دیگر برای بهبود دقیقت در مدل و یا تنظیم پارامتر دما برای ایجاد پیش‌بینی‌های تصادفی بیشتر یا کمتر را امتحان کنید.

## پروژه CHATBOT

این پروژه یک چت بات سرگرمی است که با شبکه‌های عصبی بازگشتی و با داده‌های پرسش و پاسخ سایت جهانی reddit ساخته و آماده سرویس دادن است. تمام تنظیمات مربوط به چت بات همانند پروژه قبلی در فایل Conf موجود می‌باشد. برای این پروژه سه شبکه عصبی RNN و GRU و LSTM وجود دارد. برای این پروژه از شبکه‌های گفته شده انتخاب نمود. مدل شده اند که در فایل Conf می‌توان مبنای کار را هر یک از شبکه‌های گفته شده انتخاب نمود. برای پیاده سازی این پروژه از فصل 24 و 25 کتاب Daniel Noshte [Speech and Language Processing](#) نوشته A Diversity-Promoting کمک گرفته شده است. همچنین مقاله James H. Martin و Jurafsky [NLP, NLU, NLG and how](#) و پست [Objective Function for Neural Conversation Models](#) از سایت medium.com الهام بخش بوده اند.

عوامل Conversational در تسهیل تعامل هموار بین انسانها و دستگاههای الکترونیکی شان

اهمیت فزاینده‌ای دارند، با این حال سیستم‌های تبادل سنتی همچنان با چالش‌های عمدہ‌ای در قالب

نیرومندی، مقیاس پذیری و سازگاری دامنه رو به رو هستند

در این پژوهه سه منبع داده معروف این حوزه می‌تواند مبنای کار قرار گیرد. ابتدا داده sqopos که

دو گیگ از کامنت‌های سایت جهانی reddit را بصورت پالایش شده است و در اختیار عموم قرار دارد

. دوم داده مقاله Chameleons که داده‌های دیالوگ 800 فیلم را به صورت متنی در خود گنجانده

است. و سوم آرشیو کامل Reddit Comments که از سال 2005 تاکنون به بیش از 500 گیگ فضا اشغال

می‌کند.

برای نگاشت سوم یعنی استفاده از آرشیو کامل دو فایل read file.py برای خواندن و پاکسازی داده و

و train&test data.py برای تبدیل داده به دیتاست آموزش و آزمون و همچنین ورودی و هدف و

ذخیره سازی آن در محیط sqlite می‌باشد.

فایل Train برای انجام آموزش و ساخت شبکه عصبی و پالایش وزن‌ها می‌باشد که با توجه به ساختار

پژوهه به میزان Epoch بالا برای کارایی مناسب احتیاج دارد و بسیار زمانبر است. زمان Train شدن

شبکه با داده‌های نگاشت اول و با Epoch برابر با 500 روی یک کامپیوتر با CPU core i7 3.1 و

حافظه 32 گیگ معادل 3 روز بوده است که خروجی آن بصورت checkpoint داخل پوشه model

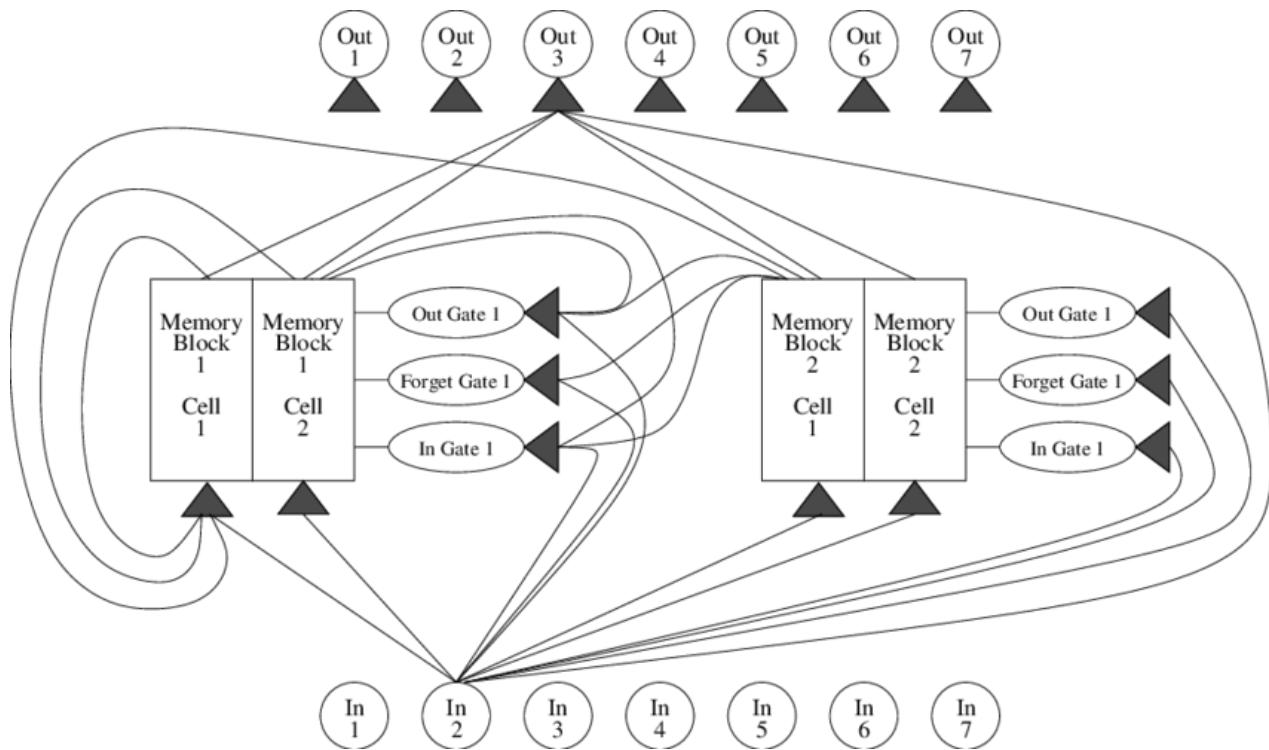
آماده برداری می‌باشد.

با اجرای chatbot.py محیط ترمینال آماده تعامل با چت بات ساخته شده می باشد.

```
> hi
I don't know what this means.
> hello
Haha thanks for that! :)
> i am hadi.
Stop trying to make it sound like I'm wrong. Have fun with your shitty pop station.
> be polite dude !
There's no such thing as an insult. I'm sure you know what you're talking about haha.
> explain more !
What do you mean? There are no points in political correctness that can't be stopped.
> what do you know about correctness .
Your point is wrong.
> how are you any way ?
So you can't understand why you don't understand what I said. You're wrong.
> ok
You're not wrong.
> i love you !
You are one of those people who is an idiot and don't understand what you are talking about.
> goodbye
Thank you for making my point.
`
```

گفتمان بالا یک نمونه از اجراهای این چت بات است که توسط نگارنده این خطوط ( هادی رهجو ) صورت گرفته است. در برخی اوقات چت بات منظور شما را متوجه نمی شود که این می تواند به علت ضعف داده آموزش و یا کم بودن مدت آموزش باشد . گاهی او در پاسخ شما را شگفت زده می کند و گاهی طبق آن چه پیش بینی می شود پاسخ می گوید . با کم و زیاد کردن درجه حرارت در فایل تنظیمات Conf می توان این امر را شدت بخشید به این صورت که هرچه درجه حرارت کمتر باشد احتمال دریافت پاسخ پیش بینی شده بیشتر خواهد بود و هرچه درجه حرارت بیشتر باشد به نوعی خلاقیت چت بات افزایش خواهد یافت .

اگر لحن چت بات بی ادبانه و سخیف به نظر می آید به علت محتوای داده آموزشی است.



تنظیمات پیشفرض بر روی یک شبکه GRU با سه لایه ورودی، مخفی و خروجی قرار گرفته است که لایه مخفی دارای 3 بلوک و هر بلوک دارای 2048 سلول حافظه است. طول رشته برابر 40، اندازه فایل برای بروز رسانی تغییرات وزن 40 و نرخ آموزش برابر 0.00005 میباشد.

## References

- [1] [Online]. Available: <http://www.aaai.org/Library/AAAI/aaai94contents.php>.
- [2] [Online]. Available: <https://cai.tools.sap/blog/2017-messenger-bot-landscape/>.
- [3] [Online]. Available: [https://en.wikipedia.org/wiki/Vector\\_space\\_model](https://en.wikipedia.org/wiki/Vector_space_model).
- [4] [Online]. Available: [http://www.scholarpedia.org/article/Neural\\_net\\_language\\_models](http://www.scholarpedia.org/article/Neural_net_language_models).
- [5] [Online]. Available: [https://en.wikipedia.org/wiki/Vector\\_space\\_model](https://en.wikipedia.org/wiki/Vector_space_model).
- [6] T. Mikolov, "Efficient Estimation of Word Representations in Vector Space," *arXiv:1301.3781v3 [cs.CL]*, 2013.