گزارش جامع کار آموزی

شرکت:

عامل‌های هوشمند دانا (عهد)

گردآورنده:

امیرعلی خطیب

۹۷۲۲۷۴۳۱۴۹

مسئول کارآموزی:

جناب آقای مهندس هادی کلماتی

استاد:

جناب آقای دکتر محمد رنجبر

مهر ۱۴۰۱


بسم الله الرحمن الرحيم

تقدیر:

از استاد بزرگوارم جناب آقای دکتر محمد رنجبر بابت اینکه این موفقیت و افتخار را داشته ام تا بتوانم به عنوان دانشجو و کارآموز در محضر حضرتعالی حضور داشته و از دانش، تجربیات و دانسته های ایشان بهره‌مند شوم بسیار به خود مفتخرم و لذا بر خود لازم می دانم مراتب قدردانی و تشکر خود را تقدیم نمایم. همچنین از جناب آقای مهندس کلماتی، مدیر عامل و هم بنیان گذار شرکت عهد و آقای مهندس فاتحی که با صبر و حوصله بسیار، در این مدت همراه بنده بودند و آموزه های خود را در اختیار بنده قرار دادند، کمال تشکر و قدردانی را دارم.

پیشگفتار:

امروزه با پیشرفت تکنولوژی و توسعه کسب و کارها در بستر وب، میزان تولید داده در جهان رشد چشم گیری داشته که به طرق مختلف توسط کاربران صفحات مجازی (لایک، کامنت، امتیاز دهی و ...)، سازمان‌های گوناگون اعم از دولتی و خصوصی تولید می‌شود. حال رقابت بین شرکت‌ها و غول‌های تجاری دنیا بر سر نحوه درست و بهینه استفاده از این داده‌ها به بهترین نحو می‌باشد به گونه‌ای که علاوه سود کلانی که شرکت از موضوع می‌برد، رضایت مشتریان را نیز در پی داشته باشد.

اهمیت داده شاید به صورت عینی برای ما قابل تشخیص نباشد. برای واضح تر شدن موضوع مثالی مطرح می‌کنم. فرض کنید شما به عنوان کاربر مدتی هست که به صفحه یک فروشگاه اینترنتی مراجعه می‌کنید. و از صفحات محصولات متعددی بازدید می‌کنید. تاریخچه بازدید یک کاربر در آن سایت یک مجموعه داده محسوب می‌شود. سازمان مربوطه اطلاعات شما رو تحلیل می‌کند. و شما خروجی این تحلیل را می‌توانید در بستر سایت فروشگاه و در قالب محصولات پیشنهادی سایت به شما، مشاهده نمایید. در واقع با استفاده روش‌های هوش مصنوعی و داده‌های مربوط به تاریخچه بازدید شما، علاقه مندی شما تشخیص داده شده و محصولات مشابه به شما پیشنهاد می‌شود.

با توجه به اهمیت مباحث این چنینی، تصمیمم بر آن شد که دوران کارآموزی را در شرکت عهد (عامل های هوشمند دانا) بگذرانم. که در حوزه سیستم‌های توصیه گر و پردازش متن فعال هستند. در این دوره کارآموزی تمرکز بر روی بحث آموزش الگوریتم های هوش مصنوعی، خزش در وب، کتابخانه های اصلی پایتون نظیر پانداز و نامپای و مباحث مقدماتی پردازش متن بود.

معرفی شرکت:

شرکت عامل‌های هوشمند دانای درنیکا (عهد)، یک تیم پژوهشی-صنعتی است که اعضای آن را پژوهشگران فعال در دو آزمایشگاه داده‌کاوی دانشگاه علم و صنعت تهران (با مدیریت دکتر بهروز مینائی بیدگلی) و همچنین قطب علمی رایانش نرم و پردازش هوشمند اطلاعات دانشگاه فردوسی (با مدیریت جناب پروفسور محمدرضا اکبرزاده توتونچی) تشکیل می‌دهند. تیم عهد در بخش تحقیق و توسعه، از یک مجموعه بسیار قدرتمند علمی شامل اساتید، دانشجویان پسادکتری، دکتری و پژوهشگران تحصیلات تکمیلی بهره میبرد. تیم عهد تجربه انجام کلان پروژههای مختلف را باز سازمانهایی همچون سازمان کل شهرداریهای کشور، سازمان فاوای شهرداری مشهد، آستان قدس رضوی، مرکز تحقیقات مخابرات ایران، بنیاد ملی بازیهای رایانهای کشور، دبیرخانه شورای عالی اطلاعرسانی، پژوهشکده آمار، شرکت سازههای اطلاعاتی راژمان و... در کارنامه دارد.

مشخصات شرکت مطابق جدول زیر می‌باشد:

جدول۱: مشخصات شرکت

نام شرکت:	عامل‌های هوشمند دانای درنیکا (عهد)
نوع کسب و کار:	نرم افزار
صنعت:	فناوری اطلاعات
ساختار مالکیت:	سهامی خاص
شناسه ملی:	۱۴۰۰۸۹۲۳۴۵۸
شماره ثبت:	۷۱۰۰۶
محل فیزیکی:	ساختمان ICT پارک علم و فناوری استان خراسان

تیم پژوهشی-صنعتی عهد قادر به ارائه خدمات زیر خواهد بود:

۱. طراحی و توسعه سیستم‌های مدیریت دانش سازمانی

۲. ارایه سیستم های نرم افزاری پردازش هوشمند متون فارسی

۳. طراحی سامانه پرسش و پاسخ خودکار

۴. تحلیل و پایش رسانه‌های خبری

۵. تحلیل شبکه‌های اجتماعی به منظور کمک به سیاست‌گذاری مدیران سازمان ها

۶. ارائه‌ی سامانه های پیشنهاد دهنده

فهرست مطالب

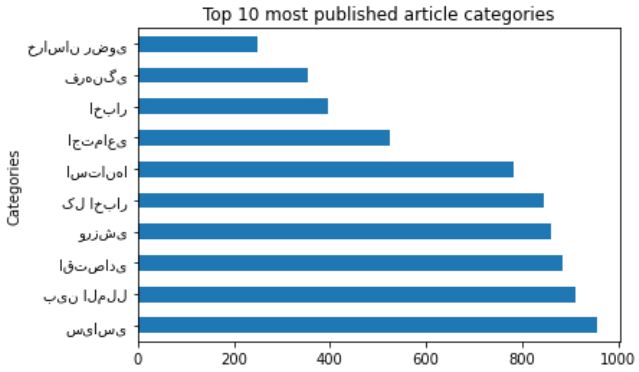
پروژه اول (تحلیل مقالات خبری):

هدف اصلی این پروژه کار با داده های حجیم می‌باشد. به گونه ای که به ما یک فایل txt با حجم ۱۳.۵ گیگ داده شده است. این فایل محتوی تعداد بالایی از مقالات خبری با موضوعات مختلف بود (مانند سیاسی، ورزشی، سلامت و ..). فایل مربوطه در قالب فایل زیپ با حجم ۱.۵ گیگ در Google drive ذخیره شده بود. که ما باید در بستر Google Colab (محیطی که ما در آن برنامه نویسی می کنیم Google Colab نام دارد . این سرویس این امکان را به ما می‌دهد که برنامه‌های خود را با پردازنده‌های قدرتمند گوگل اجرا کنیم. این مزیت را دارد که با پردازنده های قوی کد های پیچیده را اجرا کند و نیازی به صرف هزینه زیاد برای خرید یک سیستم قدرتمند نباشد. همچنین معمولا برای پیاده سازی الگوریتم‌های هوش مصنوعی از این بستر استفاده می‌شود و همچنین کتابخانه‌های مهم پایتون به صورت پیش فرض در آن نصب می‌باشد.) آن را بارگذاری کرده و از حالت زیپ خارج کنیم. فایل txt موجود در واقع شامل فایل json بوده است که به صورت string در آن ذخیره شده. باید آن را به json تبدیل کرده و پس از تبدیل فایل json به دیکشنری‌های مختلف، آن را درقالب یک دیتا فریم ذخیره می‌کنیم.

شکل ۱: مراحل شکل گیری دیتافریم آماده برای پردازش

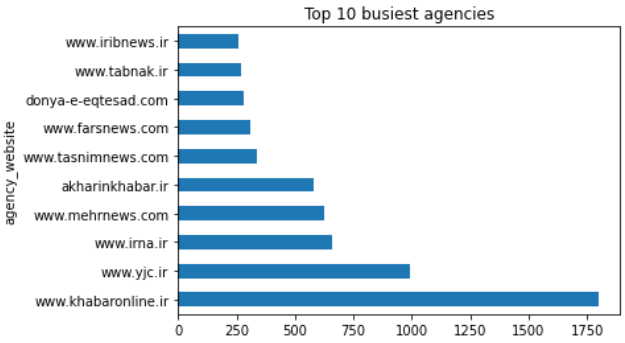
حال ما یک دیتا فریم داریم که در هر ردیف یک مقاله به خصوص قرار گرفته است. خواسته‌های این پروژه به صورت زیر می‌باشد:

۱. رسم نمودار فراوانی مقالات منتشر در ۱۰ حوزه پرتکرار.



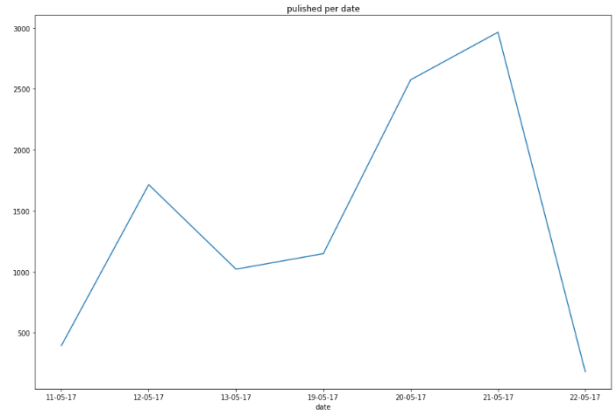
شکل ۲: نمودار میله‌ای فراوانی موضوعات مختلف (۱۰ موضوع پرتکرار)

۲. رسم نمودار میله‌ای برای فراوانی مقالات منتشر از هر پایگاه خبری.



شکل ۳: نمودار میله‌ای فراوانی مقالات منتشر شده توسط ۱۰ پایگاه خبری فعال

۳. رسم نمودار خطی برای برای میزان انشار مقالات خبری در طول زمان.



شکل ۴: تعداد مقالات خبری منتشر شده در حسب تاریخ

۴. انجام عملیات Normalization و Tokenization با استفاده از کتابخانه هضم (کتابخانه هضم یک مجموعه از توابع برای پردازش متون فارسی می‌باشد.) رو جملات تک تک مقالات و حذف تمامی stop words ها از جملات. و ایجاد یک دیتا فریم برای نمایش تعداد پر تکرار ترین bigram ها و کلمات در مقالات.

	bigram	count
314	(ریاست,جمهوری)	4736
8	(شورای,شهر)	3702
3	(انتخابات,ریاست)	3457
1548	(:ادامه,داد)	2216
2290	(:تصریح,کرد)	1988
1747	(رئیس,جمهور)	1969
7633	(اسلامی,شهر)	1930
334	(شورای,اسلامی)	1881
1998	(:اظهار,داشت)	1772
6799	(جمهوری,اسلامی)	1730

	word	count
4	انتخابات	18863
158	کشور	13989
118	ایران	12866
255	سال	12159
527	رای	11678
11	شهر	10333
400	گزارش	9801
259	دولت	9193
415	اسلامی	8688
608	استان	8341

شکل ۵: عکس جدول خروجی در کد مربوطه برای فراوانی ۱۰ کلمه و بایگرام پرتکرار

پروژه شماره ۱ به پیوست گزارش ارسال شده است.

توضیحاتی در خصوص اصطلاحات مطرح شده در گزارش پروژه ۱:

نرمال کردن متن (Normalization):

ابتدا متن را نرمال می‌کنیم به منظور ویرایش متن نظیر اصلاح نیم فاصله ها و غیره.

توکنایز کردن متن (Tokenization):

توکنایز به معنای تجزیه یک متن به جملات آن و تمامی کلمات موجود در آن می‌باشد. مانند شکل زیر با کتابخانه nltk (کتابخانه مرسوم برای پیش پردازش متون انگلیسی) پیاده سازی شده. در ادامه باید stop words ها (به کلمات و اجزای بی‌اهمیت متن می‌گویند مانند افعال کمکی یا نشانه نگارشی مانند نقطه ویرگول و پرانتز) نیز از میان خروجی توکنایز کلمات حذف کنیم که در مثال آورده نشده.

```
>> text = “Natural language processing (NLP) is a field “ + \
       “of computer science, artificial intelligence “ + \
       “and computational linguistics concerned with “ + \
       “the interactions between computers and human “ + \
       “(natural) languages, and, in particular, “ + \
       “concerned with programming computers to “ + \
       “fruitfully process large natural language “ + \
       “corpora. Challenges in natural language “ + \
       “processing frequently involve natural “ + \
       “language understanding, natural language” + \
       “generation frequently from formal, machine” + \
       “-readable logical forms), connecting language “ + \
       “and machine perception, managing human-“ + \
       “computer dialog systems, or some combination “ + \
       “thereof.”
```

```
>> print(sent_tokenize(text))

output[1]:

['Natural language processing (NLP) is a field of computer science,
artificial intelligence and computational linguistics concerned with
the interactions between computers and human (natural) languages,
and, in particular, concerned with programming computers to
fruitfully process large natural language corpora.', 'Challenges in
natural language processing frequently involve natural language
understanding, natural language generation (frequently from formal,
machine-readable logical forms), connecting language and machine
perception, managing human-computer dialog systems, or some
combination thereof.']

>> print(word_tokenize(text))

output[2]:

['Natural', 'language', 'processing', '(', 'NLP', ')', 'is', 'a',
'field', 'of', 'computer', 'science', ',', 'artificial',
'intelligence', 'and', 'computational', 'linguistics', 'concerned',
'with', 'the', 'interactions', 'between', 'computers', 'and',
'human', '(', 'natural', ')', 'languages', ',', 'and', ',', 'in',
'particular', ',', 'concerned', 'with', 'programming', 'computers',
'to', 'fruitfully', 'process', 'large', 'natural', 'language',
'corpora', '.', 'Challenges', 'in', 'natural', 'language',
'processing', 'frequently', 'involve', 'natural', 'language',
'understanding', ',', 'natural', 'language', 'generation', '(',
'frequently', 'from', 'formal', ',', 'machine-readable', 'logical',
'forms', ')', ',', 'connecting', 'language', 'and', 'machine',
'perception', ',', 'managing', 'human-computer', 'dialog', 'systems',
',', 'or', 'some', 'combination', 'thereof', '.']
```

بایگرام در پردازش متن (bigram):

یک زوج مرتب از کلمات موجود در یک جمله. به عنوان مثال جمله زیر را در نظر بگیرید:

بابا آب داد. نادان ادب ندارد.

حال بایگرام‌های جمله بالا مطابق زیر می‌باشد.

[(بابا، آب)، (آب، داد)، (داد، بابا)، (ندارد، ادب)، (نادان، ادب)، (ندارد، نادان)]

پروژه دوم (خزش در وب یا web scarping):

در اکثر قریب به اتفاق پروژه های صنعتی داده‌ی آماده و از پیش گردآوری شده نداریم و باید عمل جمع آوری داده را انجام بدهیم. یکی از روش‌های جمع‌آوری داده‌ها خزش در وب یا web scraping می‌باشد. به این صورت که ما داده‌های مورد نیازمان را با استفاده از یکسری دستورات از پیش توسعه داده شده از سطح صفحات وب دریافت می‌کنیم تا در نهایت یک دیتاست ایجاد کنیم. کتابخانه‌های گوناگونی در حوزه خزش در وب توسعه داده شده‌اند که می‌توان به معروف ترین آنها از جمله Scrapy (توجه: اسکرپی یک فریم ورک از و کتابخانه نمی‌باشد.)، Beautiful soup و Selenium اشاره کرد.

شکل ۶: کتابخانه‌ها و فریم ورک‌های معروف پایتون در حوزه خزش در وب

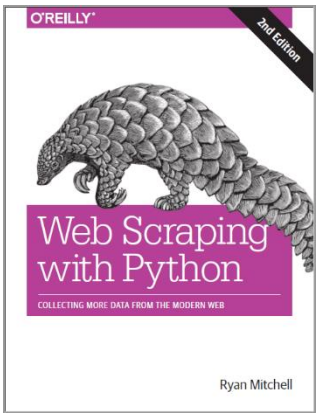
در پروژه ذکر شده ما باید به سایت منظوم (مرجع اطلاعات فیلم و سینما) مراجعه کرده و اطلاعات مربوط به بیوگرافی تمامی بازیگران ایرانی را از صفحات سایت دریافت کنیم و در قالب یک دیتاست با فرمت json ذخیره کنیم.

در ابتدای انجام این پروژه به دلیل عدم با آشنایی با زبان html، مجبور شدم به مدت حدودا ۵ روز مشغول یادگیری این زبان از سایت w3school.com شوم تا بتوانم کدهای موجود در صفحات وب را بخوانم و اطلاعات مورد نیازم را با آدرس‌دهی به وسیله تگ‌های html و دستورات موجود در کتابخانه های مربوطه دریافت کنم.

پس از یادگیری html پروژه را همزمان با یادگیری فریم ورک Scrapy شروع کردم این ابزار به دلیل سرعت بالا بسیار مورد توجه متخصصین این حوزه قرار گرفته است. ولی یکی از از معایب آن کاربر پسند نبودن آن است زیرا باید با استفاده از تکنیک های برنامه نویسی شی گرا از آن استفاده کنیم و قابل استفاده در بستر فایل‌های ipynb نمی‌باشد. به همین دلیل کار با آن کمی سخت تر است. که پس کار با آن ادامه پروژه با این کتابخانه در زمان موجود وقت‌گیر دیدم و تصمیم تغییر ابزارم کردم. کتابخانه Beautiful soup گزینه جایگزین من برای انجام پروژه بود. این کتابخانه به دلیل سهولت استفاده بسیار مورد توجه است. از معایب آن می‌توان به کند بودن آن اشاره داشت. در این پروژه من از کتابخانه Selenium هم استفاده کردم. کتابخانه Selenium علاوه بر قابلیت استفاده در وب اسکرپینگ، یک کنترلر جاوا اسکریپت به حساب می‌آید؛ به این معنی که کدهای آن همانند یک کاربر رفتار می کنند. برای مثال وقتی شما وارد یک

سایت می شوید و میخواهید کامنت های زیر یک مطلب را مشاهده کنید، تمامی کامنت به صورت کامل نماش داده نمی‌شود. شما باید روی گزینه "نمایش بیشتر" یا اصطلاحا "show more" برای مشاهده تعداد بیشتری از کامنت ها کلیک کنید. فرض کنید می‌خواهیم تمام نظرات کاربران برای همه بازیگران استخراج کنیم. ابتدا لازم تا جایی امکان دارد روی گزینه‌ی نمایش بیشتر کلیک کنیم تا تمامی کامنت‌های موجود نمایش داده شود. و سپس با کمک تگ‌های html آن‌ها و توابع کتابخانه مورد نظر، آن ها را دریافت و ذخیره می‌کنیم. همچنین یکی از چالش‌های دیگر جابجایی در بین صفحات وب بود. به این صورت که پس از ذخیره اطلاعات یک بازیگر، باید وارد صفحه اطلاعات مربوط به بازیگر دیگر شویم که خود کمی دشوار بود.

یکی از مراجعی در انجام این پروژه به آن مراجعه شد، کتابی از انتشارات O'REILLY با عنوان Web Scraping with Python می‌باشد که نحوه کار با Scrapy و Beautiful soup در آن گنجانده شده است.



شکل ۷: کتاب مورد استفاده در پروژه برای خزش در وب

در پروژه مربوطه ما برای تمام بازیگران به صفحه مشابه این لینک اطلاعات زیر را باید با عملیات خزش می‌آوردیم:

۱. نام
۲. تاریخ تولد
۳. کل متن بیوگرافی
٤. جوایز
٥. سوابق کاری (نام اثر , لینک صفحه اثر , سال تولید , امتیاز اثر)
٦. مشخصات (راههای ارتباطی , اطلاعات زندگی , تحصیلات , مهارت ها)
۷. حواشی
۸. نظرات مردم (نام کاربر , لینک صفحه کاربر , متن کامنت , تعداد لایک , تاریخ انتشار)

پروژه شماره ۲ نیز به پیوست گزارش ارسال شده است.

پروژه سوم (خوشه بندی متون پرسش و پاسخ دینی):

یکی از مجموعه الگوریتم‌های بدون نظارت خوشه‌بندی یا clustering نام دارد. خوشه بندی زمانی به کار گرفته می‌شود که داده‌های ما متغیر پاسخ ندارد. و با استفاده از ویژگی های آنها نزدیکترین داده ها را یکدیگر شناسایی شده و در یک گروه یا خوشه قرار می‌دهیم. از الگوریتم‌های مرسوم خوشه بندی k_means نام دارد. که در اینجا الگوریتم آن به اختصار توضیح داد می‌شود.

فرض کنید داده‌هایی با n ویژگی در اختیار داریم.

$$X = (x_1, x_2, \dots, x_n)$$

حال یک فضای هندسی n بعدی در نظر بگیرید که داده‌هایی مثل مورد ذکر شده در بالا در آن قرار گرفته باشند. هدف ما این است که تعدادی مشخصی گروه تعیین کنیم و داده‌هایی که بیشترین شباهت را به یکدیگر دارند در هر یک از این گروه‌ها یا به اصطلاح خوشه‌ها قرار گیرند. میزان شباهت معمولا با فاصله اقلیدسی سنجیده می‌شود. گام‌های الگوریتم به صورت زیر می‌باشد.

۱. تعیین تعداد خوشه مورد نظر(K)
۲. انتخاب تصادفی K تا از داده ها به عنوان نقاط اصلی هر خوشه. (centroid)
۳. محاسبه فاصله اقلیدسی (یا هر شاخص سنجش شباهت دیگری) هر یک از داده ها با centroid ها و تعیین خوشه مناسب (خوشه مناسب خوشه‌ای است که centroid متعلق به آن نسبت به centroid های دیگر به داده نزدیک تر باشد) برای داده مورد نظر.
۴. محاسبه میانگین داده‌های هر خوشه و قرار دادن centriod در محل میانگین آن خوشه.
۵. تکرار مورد ۳ و۴ تا جایی که داده ها خوشه عوض نکنند.

حال هدف ما در مسئله پرسش و پاسخ دینی این است که با استفاده از آنالیز متون پرسش و پاسخ بتوانیم محتواهای نزدیک به یکدیگر را در یک گروه یا خوشه قرار دهیم. برای این کار ابتدا لازم است متون پرسش و پاسخ را پیش پردازش کنیم و تبدیل به بردار کنیم و به عنوان ورودی به الگوریتم ذکر شده بدهیم. مثل پروژه شماره ۱ باید عملیات نرمال سازی، توکنایز کردن، استمینگ (stemming) و حذف کلمات توقف و علائم اضافی داده روی متون پرسش و پاسخ پیاده سازی کنیم. برای برداری کردن متون مسئله از ما سه روش خواسته که به شرح زیر می‌باشد:

۱. روش Bag of words:

فرض کنید ۳ جمله داریم که می خواهیم مدل Bag of words را برای آن ایجاد کنیم.

	she	loves	pizza	is	delicious	a	good	person	people	are	the	best
She loves pizza, pizza is delicious	1	1	2	1	1	0	0	0	0	0	0	0
She is a good person	1	0	0	1	0	1	1	1	0	0	0	0
good people are the best	0	0	0	0	0	0	1	0	1	1	1	1

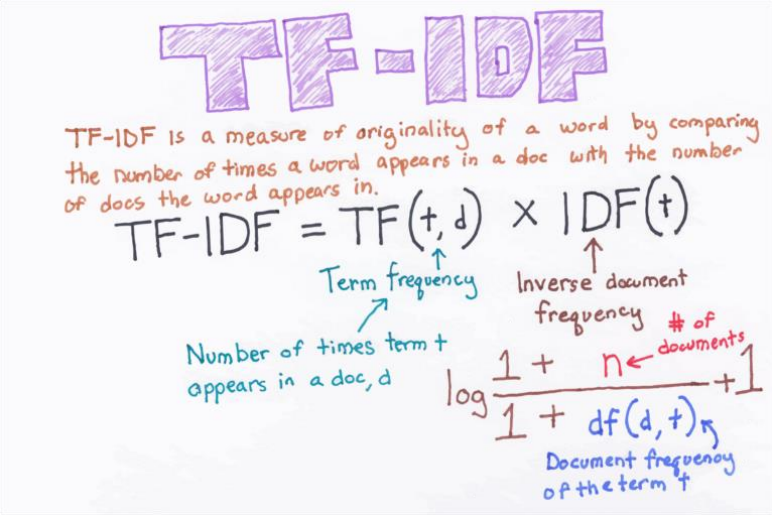
شکل ۸: نمونه پاده سازی شده در Bag of words روی جملات

در مجموعه داده پرسش و پاسخ دینی ما تمامی کلمات موجود در این مجموعه داده را بدست می‌آوریم. برای تمامی پرسش و پاسخ ها بردار ایجاد می‌کنیم. مانند چیزی که در شکل بالا مشاهده می‌کنید با این تفاوت که ابعاد بردارهای هر پرسش و پاسخ بسیار بلند است زیرا به ازای تمامی کلمات موجود در مجموعه داده طول بردار هر پرسش و پاسخ است. و این کار پردازش را سخت می‌کند. به این صورت کوله ای از کلمات در ماتریس ساخته می شود که این ماتریس می تواند به الگوریتم های بعدی برای عملیاتی مثل طبقه بندی یا خوشه بندی اعمال شود. در نهایت یک سری داده های غیر ساختار یافته را به یک داده های عددی به صورت ماتریس تبدیل کردیم.

۲. روش TF-IDF:

یک شاخص آماری است که ارتباط یک کلمه در داکیومنت(در مسئله ما یک پرسش و پاسخ) را در مجموعه‌ای از داکیومنت‌ها(تمام پرسش و پاسخ ها) ارزیابی می‌کند. این موضوع با ضرب دو مقایس انجام می‌گیرد

۱. چند بار یک لغت در یک داکیومنت تکرار شده است.
۲. نسبت تکرار شدن کلمه کلیدی به نسبت کل داکیومنت‌ها.



شکل ۹: روش tf-idf

۳. استفاده از بردارهای پیش‌ساخته Fasttext (Facebook word embedding on large data):

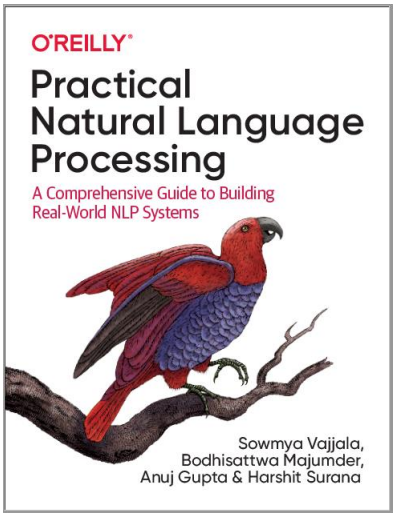
در این روش ما یک سری خروجی از پیش آماده شده داریم که ارتباط معنایی تمامی کلمات فارسی با سیصد کلمه گنجانده شده. این خروجی حاصل روشی ست به نام word embedding که با الگوریتم‌های هوش مصنوعی نظیر شبکه های عصبی روی حجم بالایی از متون (مثلا تمام ویکی پدیای فارسی) پردازش شده و هر کلمه یک بردار ۳۰۰ بعدی ست. هر کدام از آن ابعاد مختص یک واژه به خصوص می‌باشد. مقدار آن بیانگر ارتباط معنایی شان می‌باشد.

عملکرد این بردارهای معنایی به گونه ای ست که روابطی از این دست میان بردارهای کلمات ایجاد می‌شود.

$$king - men + women \sim queen$$

هر سه روش روی داده‌های پرسش پاسخ پیاده سازی شد که در نهایت روش سوم که از خروجی‌های fasttext استفاده ش بود، به دلیل رابطه معنایی قوی سرعت بسیار بالا در آموزش مدل خوشه بندی و عملکرد بسیار بهتر از سایر روش به عنوان بهترین روش انتخاب گردید.

یکی از مراجعی در انجام این پروژه به آن مراجعه شد، کتابی از انتشارات O’REILLY با عنوان Practical Natural Language Processing می‌باشد که مباحث مقدماتی کار با داده‌های متنی در آن گنجانده شده است.

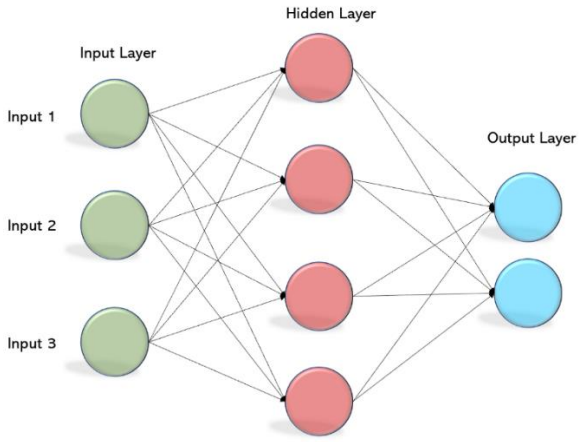


شکل ۱۰: کتاب مورد استفاده در پروژه برای پردازش متن

کد پروژه شماره ۳ به پیوست این گزارش ارسال می‌گردد.

پروژه چهارم (پیش بینی سری زمانی دما با استفاده از شبکه عصبی MLP):

یکی از الگوریتم‌های مهم حوزه هوش مصنوعی شبکه‌های عصبی نام دارند که از ایده اصلی آن مدل سازی ریاضی عملکرد مغز انسان بوده است. یکی از مدل‌های ابتدایی شبکه‌های عصبی، شبکه‌های عصبی mlp یا multi-layer perceptron ها می‌باشند. شکل یک شبکه عصبی mlp را نشان می‌دهد. این شبکه‌ها دارای یک لایه ورودی، یک لایه خروجی و حداکثر دو لایه پنهان می‌باشند.



شکل ۱۱: شبکه عصبی MLP

در پیکان های موجود بین نورون‌های شبکه یک وزن تعریف می‌شود. برای مثال در شبکه بالا ما به تعداد زیر وزن داریم:

$$n_{input\ layer} = 3\ ; n_{hidden\ layer} = 4\ ;\ \ n_{output\ layer} = 2$$

$$n_{weight} = n_{input\ layer} \times n_{hidden\ layer} + n_{hidden\ layer} \times n_{output\ layer} + \left(n_{hidden\ layer} + n_{output\ layer}\right) = 12 + 8 + 6 = 26$$

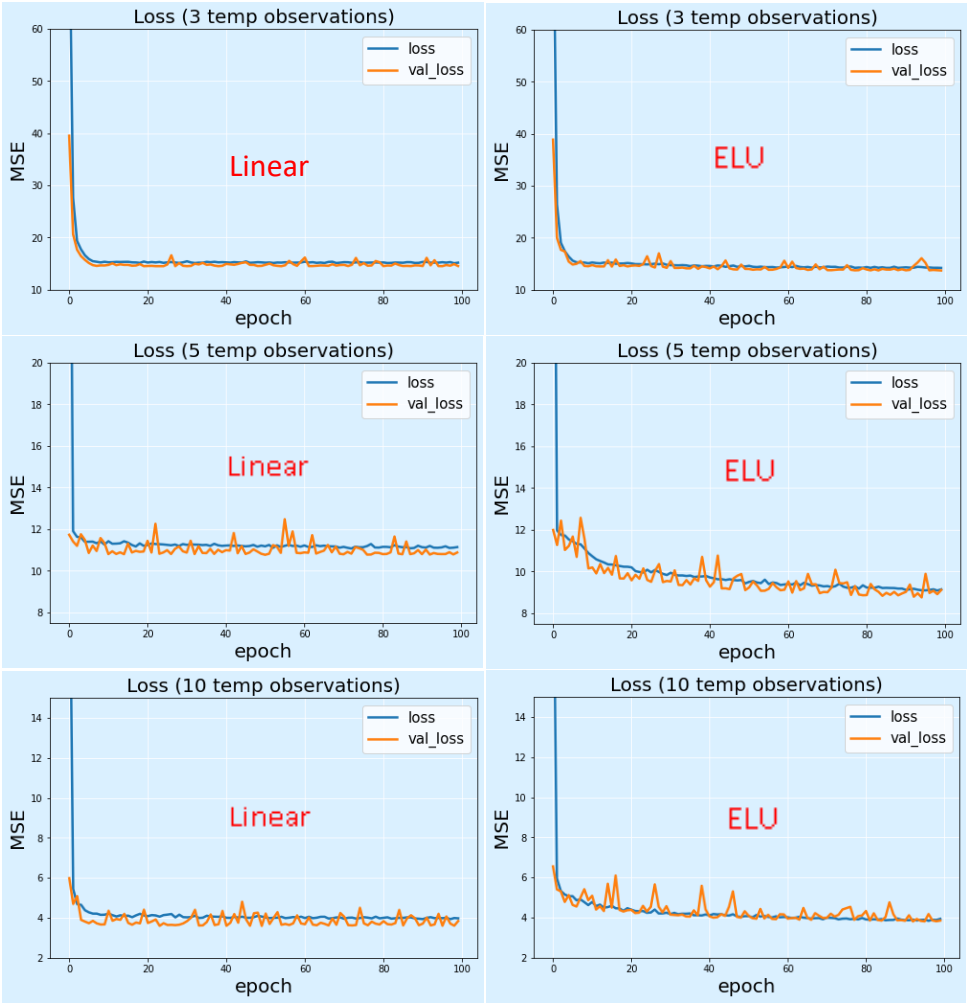
ما ابتدا با دادن مقادیر تصادفی به وزن‌ها الگوریتم آغاز می‌کنیم. و با دادن هر ورودی و مشاهده خروجی بدست آمده از شبکه، با توجه جواب داده ورودی یا مقداری واقعی آن در دیتاست وزن‌ها را آپدیت می‌کنیم (اصطلاحا به آن Back propagation می‌گوئیم.). این عمل آن قدر تکرار می‌شد تا با تظیم شدن وزن ها الگوریتم به عملکرد قابل قبولی برسد.

حال در این مسئله ما یک مجموعه داده زمانی در تغییرات دمای یک منطقه در دست داریم. هدفمان برسی عملکرد مدل شبکه عصبی با ۳، ۵ و ۱۰ ورودی با توابع فعال سازی ELU و Linear می‌باشد. منظور از تعداد ورودی این است که توالی مورد نظر را به دیتاستی تبدیل کنیم که هر داده آن شامل n ویژگی و ۱ پاسخ باشد. منظور از n ویژگی n زمان قبل زمان پیش‌بینی می‌باشد. برای مثال شکل رو به رو ۶ داده اول دیتاست ایجاد شده با ۵ ویژگی می‌باشد.

	t1	t2	t3	t4	t5	y
0	8.2	7.2	10.8	17.6	18.2	11.6
1	7.2	10.8	17.6	18.2	11.6	10.2
2	10.8	17.6	18.2	11.6	10.2	8.8
3	17.6	18.2	11.6	10.2	8.8	10.4
4	18.2	11.6	10.2	8.8	10.4	11.2
5	11.6	10.2	8.8	10.4	11.2	14.0

شکل ۱۲: تصویر دیتا فریم ایجاد شده در کد برای بیش‌بینی سری زمانی با ۵ ویژگی یا زمان

حال شبکه عصبی روی داده‌های مذکور آموزش داده شد. مقدار خطا در هر epoch در نمودارهای زیر نمایش داده شده است.



شکل ۱۳: نمودار های تغییرات خطای شبکه عصبی در هر epoch

عملکرد هر ۶ حالت آموزش داده شده در قالب جدول زیر قابل مشاهده می‌باشد.

	NeuralNet1(df3)	NeuralNet1(df5)	NeuralNet1(df10)
linear	15.513148	10.978749	4.099261
elu	14.856105	9.047754	3.765659

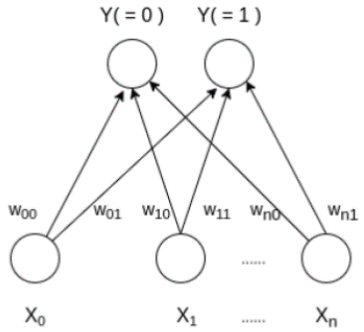
شکل ۱۴: تصویر دیتا فریم خروجی رد کد برای میزان دقت بدست آماده از ۶ شبکه عصبی آموزش داده شده

پروژه پنجم (پیاده سازی الگوریتم طبقه بندی LVQ و خوشه بندی SOM رو داده‌های MNIST):

پیش از پرداختن به دیتاست مربوطه نیاز است با الگوریتم طبقه بندی LVQ و خوشه‌بندی SOM آشنا شویم که به شرح زیر می باشد.

الگوریتم طبقه بندی LVQ:

این الگوریتم نوعی شبکه عصبی است که دارای یک لایه ورودی و یک لایه خروجی است. و به تعداد حاصل ضرب تعداد نورون‌های ورودی در نورون‌های خروجی می‌باشد. در شکل زیر معماری شبکه عصبی LVQ قابل مشاهده می‌باشد.



شکل ۱۵: شبکه عصبی LVQ

در الگوریتم فوق گام‌های زیر طی می‌شود:

۱. به وزن‌ها مقادیر تصادفی می دهیم یا اینکه به تعداد نورون های خروجی داده انتخاب می‌کنیم و مقادیر داده‌ها را به عنوان وزن قرار می دهیم (برای مقدار دهی اولیه وزن‌های منتهی به یک نورون خروجی که برای یک جواب بخصوص می‌باشد، باید داده‌ای انتخاب شود که جواب همان نورون را داشته باشد.)
۲. داده ورودی به شبکه می‌دهیم و فاصله آن با مجموعه وزن های متنهی به هر نورون خروجی مقایسه می‌شود

۳. کمترین فاصله با مجموعه وزن‌های منتهی به یک نورون به خصوص انتخاب و جواب موجود در نورون خروجی، پیش‌بینی شبکه از جواب داده مورد نظر می‌باشد.
۴. حال اگر جواب پیش بینی شده با جواب واقعی داده یکی باشد با رابطه زیر وزن‌ها می‌کنیم:

$$weight := weight + \alpha.(sample - weight)$$

در صورت عدم برابر از رابطه زیر استفاده می‌کنیم:

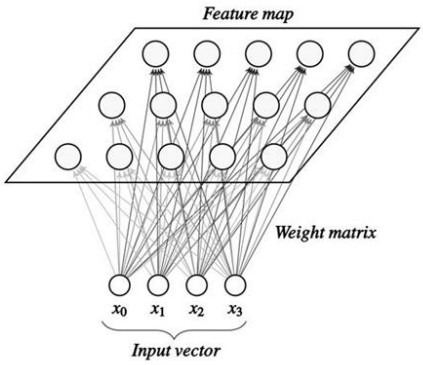
$$weight := weight - \alpha.(sample - weight)$$

۵. تا رسیدن با دقت قابل قبول گام‌های ۲ تا ۴ را تکرار می‌کنیم.

این روش یک روش نظارت شده می‌باشد.

الگوریتم خوشه بندی SOM:

شبکه‌های عصبی SOM یا Self-Organizing Map که با نام شبکه کوهونن (Kohonen Network) نیز شناخته می‌شوند، یک روش غیرنظارت‌شده (Unsupervised Learning) برای استخراج ویژگی و کاهش ابعاد است که با وجود سادگی، توانایی زیادی از خود نشان داده است. در این شبکه، تعدادی نورون با موقعیت اولیه تصادفی انتخاب می‌شوند که این نورون‌ها در یک شبکه منظم به نام Lattice در کنار هم قرار گرفته‌اند. در طول آموزش، نورون‌های شبکه به مکان‌هایی با چگالی بیشتر داده حرکت می‌کنند و فرم نهایی Lattice حاصل می‌شود.



شکل ۱۶: شبکه عصبی SOM

در الگوریتم فوق گام‌های زیر طی می‌شود:

۱. به وزن‌های منتهی به هر نورون موجود در lattice مقدار تصادفی می‌دهیم.
۲. یک داده وارد شبکه می‌شود. فاصله همه بردار وزن‌های منتهی به نورون‌ها از بردار ورودی محاسبه می‌شود.
۳. نزدیک‌ترین نورون به بردار ورودی تعیین و به‌عنوان نورون برنده انتخاب می‌شود. و موقعیت نورون برنده با استفاده از رابطه زیر بروز می‌شود:

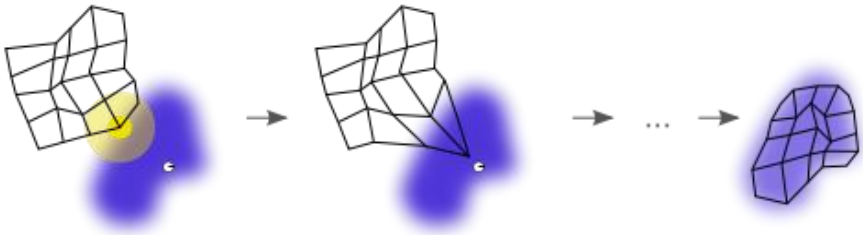
$$weight := weight + \alpha.(sample - weight)$$

۴. موقعیت نورون‌های موجود در همسایگی نورون برنده با استفاده از رابطه زیر بروز می‌شود:

$$weights := weights + \theta.\alpha(sample - weights)$$

مقدار θ نیز تعیین کننده بحث همسایگی می‌باشد که مقداردهی اولیه پارامتر آن در عملکرد مسئله بسیار تأثیرگذار است.

۵. تا زمان رسیدن به عملکد مناسب برای الگوریتم، گام‌های ۲ تا ۴ را تکرار می‌کنیم.



شکل ۱۷: تغییرات lattice طی تکرار های الگوریتم SOM

حال دیتاست Mnist مربوط تصاویر ارقام دست نویس به الگوریتم‌ها داده شده. که کدهای آن به پیوست این تمرین ارسال می‌گردد.

شکل ۱۸: نمونه‌ای از دیتاست Mnist

کدهای مربوط به دو الگوریتم به پیوست این گزارش ارسال می‌گردد.

نتیجه گیری:

در طول این کارآموزی من با طیف گسترده ای از الگوریتم های یادگیری ماشین اما از الگوریتم های مبتنی بر یادگیری بدون نظارت نظیر خوشه بندی و مزارت شده نظیر رگرسیون و طبقه بندی و همچنین استفاده از شبکه های عصبی در هر دو حوزه آشنا شدم. در کنار این مباحث با ابزارهای استخراج داده از صفحات وب آشنا شوم و بتوانم در صورت نیاز به گردآوری داده از آنها استفاده کنم. تمامی موارد ذکر شده در بالا جزوه مهارت های مورد نیاز یک متخصص داده در شرکت های مختلف تلقی می شود. امید است که بتوانم با قوی تر کردن مهارت هایم در حوزه هوش مصنوعی و داده کاوی بتوانم سیستمهای هوشمندی را برای سهولت بخشیدن به امور بشر توسعه دهم.

انتقادات:

محتوای به اشتراک گذاشته توسط شرکت برای موضوع خزش در وب کمی از لحاظ صدا دارای مشکل بود که خود انتقال مطالب کمی سخت می‌کرد.

پیشنهادات:

بهتر است در حین کارآموزی کارآموزان تشویق به مطالعه کتاب تکست بکنید و خارج از محتواهای به اشتراک گذاشته شده، منابع بیشتری معرفی کنید

فهرست منابع:

الف) کتاب‌ها:

1. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition (Aurelien Geron)
2. Practical Natural Language Processing A Comprehensive Guide to Building Real-World NLP Systems (Sowmya Vajjala, Bodhisattwa Majumder etc.)
3. The StatQuest illustrated guide to machine learning (Josh Starmer)
4. Python for Data Analysis, 2nd Edition (Wes McKinney)

ب) یوتیوب:

1. Krish Naik – NLP playlist
2. Morita Data Land – NLP playlist
3. Morita Data Land – web scraping playlist
4. Free Code Camp - selenium

ج) وب‌سایت:

1. Toward Data science
2. HTML tutorial – W3schools
3. Fasttext