



دانشگاه صنعتی امیرکبیر  
(پلی تکنیک تهران)  
دانشکده مهندسی کامپیوتر

پروژه کارشناسی  
گرایش نرم افزار

طراحی و پیاده سازی سیستم پرسش و پاسخ خودکار  
سوالات ساده فارسی

نگارش

علیرضا ترابیان

استاد راهنما

دکتر سعیده ممتازی

شهریور ۱۳۹۹

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

# صفحه فرم ارزیابی و تصویب پایان نامه - فرم تأیید اعضاء کمیته دفاع

در این صفحه فرم دفاع یا تأیید و تصویب پایان نامه موسوم به فرم کمیته دفاع - موجود در پرونده آموزشی - را قرار دهید.

## نکات مهم:

- نگارش پایان نامه/رساله باید به **زبان فارسی** و بر اساس آخرین نسخه دستورالعمل و راهنمای تدوین پایان نامه های دانشگاه صنعتی امیرکبیر باشد.(دستورالعمل و راهنمای حاضر)
- رنگ جلد پایان نامه/رساله چاپی کارشناسی، کارشناسی ارشد و دکترا باید به ترتیب مشکی، طوسی و سفید رنگ باشد.
- چاپ و صحافی پایان نامه/رساله بصورت **پشت و رو(دورو)** بلامانع است و انجام آن توصیه می شود.



دانشگاه صنعتی امیرکبیر  
(پلی تکنیک تهران)

به نام خدا

## تعهدنامه اصالت اثر

تاریخ: شهریور ۱۳۹۹

اینجانب **علیرضا ترابیان** متعهد می‌شوم که مطالب مندرج در این پایان‌نامه حاصل کار پژوهشی اینجانب تحت نظارت و راهنمایی اساتید دانشگاه صنعتی امیرکبیر بوده و به دستاوردهای دیگران که در این پژوهش از آنها استفاده شده است مطابق مقررات و روال متعارف ارجاع و در فهرست منابع و مآخذ ذکر گردیده است. این پایان‌نامه قبلاً برای احراز هیچ مدرک هم‌سطح یا بالاتر ارائه نگردیده است. در صورت اثبات تخلف در هر زمان، مدرک تحصیلی صادر شده توسط دانشگاه از درجه اعتبار ساقط بوده و دانشگاه حق پیگیری قانونی خواهد داشت.

کلیه نتایج و حقوق حاصل از این پایان‌نامه متعلق به دانشگاه صنعتی امیرکبیر می‌باشد. هرگونه استفاده از نتایج علمی و عملی، واگذاری اطلاعات به دیگران یا چاپ و تکثیر، نسخه‌برداری، ترجمه و اقتباس از این پایان‌نامه بدون موافقت کتبی دانشگاه صنعتی امیرکبیر ممنوع است. نقل مطالب با ذکر مآخذ بلامانع است.

**علیرضا ترابیان**

امضا

تقدیم بہ  
خالق بی نہایت،  
پرستگارِ خلاق،  
و غواصانِ گشتی ناپذیرِ دریایِ پانچ ماہ۔

# سپاس‌گزاری

از پدرم که چون کوهی استوار و مادرم که چون دریای محبت در فراز و نشیب زندگی دلسوزانه  
همراهم بوده‌اند؛  
از استاد بزرگوار سرکار خانم دکتر سعیده ممتازی که در کمال سعه صدر، با حسن خلق و فروتنی،  
رهنمون من شده‌اند؛  
و از سایر عزیزانی که در کنارشان این نتیجه حاصل آمد کمال تشکر و قدردانی را دارم.

علیرضا تریان  
شهریور ۱۳۹۹

## چکیده

امروزه با در دسترس قرار گرفتن انبوهی از اطلاعات یکی از مشکلات پیش رو چگونگی جستجو و یافتن اطلاعات مد نظر در میان این انبوه داده می‌باشد. از راحت‌ترین راه‌های جستجو برای یک کاربر عادی پرسش اطلاعات مد نظر خود است. به همین منظور سیستم‌های پرسش و پاسخ یکی از مهم‌ترین کاربردهای موجود در جستجوی اطلاعات می‌باشند. هدف از این پروژه طراحی سامانه‌ای است که در آن امکان پرسش سوالات ساده فارسی فراهم شده باشد. این سوالات می‌توانند در دسته‌بندی‌های مختلف از جمله پایتخت کشور، کارگردان فیلم یا سریال و ... باشد و پاسخ آنها با استفاده از گراف دانش فارسی فراهم می‌گردد. این پروژه مبتنی بر زبان برنامه‌نویسی پایتون پیاده‌سازی و ارزیابی می‌شود. مدل‌های پیاده‌سازی شده در این پژوهش بر اساس روش‌های ماشین بردار پشتیبان و شبکه‌ی عصبی پیچشی می‌باشد. هر دو روش مذکور در آزمون‌های مختلف بررسی و خطاهای هر کدام مورد بررسی قرار گرفته شده است. تمامی آزمایش‌های این پروژه به صورت با نظارت انجام گرفته است. علاوه بر موارد فوق اثر تغییر مدل‌های مختلف برای نگاشت جملات به بردار ویژگی نیز بررسی شده است. نتایج آزمایش‌ها بر روی دادگانی که در راستای همین پروژه تهیه شده است نشان می‌دهد که دقت روش پیشنهادی در پاسخ‌گویی به سوالات فارسی با استفاده از اطلاعات گراف دانش فارسی ۲۹.۰۷ درصد می‌باشد.

## واژه‌های کلیدی:

سیستم پرسش و پاسخ، شبکه‌ی عصبی پیچشی، ماشین بردار پشتیبان، گراف دانش، سوال ساده‌ی فارسی

# فهرست مطالب

صفحه

عنوان

۱	مقدمه	۱
۲	۱-۱ مقدمه	۲
۲	۲-۱ تعریف مسئله	۲
۳	۳-۱ تعریف زیرمسئله‌ها	۳
۳	۱-۳-۱ تشخیص رابطه	۳
۴	۲-۳-۱ تشخیص موجودیت‌ها	۴
۴	۳-۳-۱ تولید کوئری	۴
۴	۴-۱ خلاصه فصل‌های بعد	۴
۵	۲ مفاهیم پایه	۵
۶	۱-۲ پردازش زبان طبیعی	۶
۶	۲-۲ بازبانی اطلاعات	۶
۷	۳-۲ گراف دانش	۷
۸	۴-۲ سیستم پرسش و پاسخ	۸
۹	۵-۲ شبکه‌های عصبی	۹
۹	۱-۵-۲ تعریف پایه	۹
۱۰	۲-۵-۲ شبکه‌های عصبی پرسپترون و پرسپترون چندلایه	۱۰
۱۲	۳-۵-۲ شبکه‌ی عصبی پیچشی	۱۲
۱۴	۴-۵-۲ شبکه‌ی عصبی بازگشتی	۱۴
۱۶	۵-۵-۲ شبکه‌ی عصبی دنباله به دنباله	۱۶
۱۷	۶-۲ تابع هزینه‌ی Categorical Cross-Entropy	۱۷
۱۸	۳ مروری بر کارهای گذشته	۱۸
۱۹	۱-۳ روش‌های مبتنی بر ارائه‌ی معنایی	۱۹
۲۰	۲-۳ روش‌های مبتنی بر تشخیص رابطه	۲۰
۲۰	۳-۳ روش‌های تولیدکننده‌ی جمله‌ی پاسخ	۲۰



۴-۳	جمع‌بندی	۲۱
۴	سیستم پیشنهادی برای پرسش و پاسخ فارسی	۲۲
۴-۱	گراف دانش فارسیس	۲۳
۴-۲	مجموعه دادگان	۲۴
۴-۳	تشخیص رابطه	۲۶
۴-۳-۱	تعبیه‌سازی کلمات	۲۷
۴-۳-۲	دسته‌بند ماشین بردار پشتیبان	۲۷
۴-۳-۳	دسته‌بند شبکه عصبی پیچشی	۲۸
۴-۴	تشخیص موجودیت‌ها	۲۹
۴-۵	تولید کوئری	۲۹
۵	تحلیل، طراحی و پیاده‌سازی سیستم	۳۲
۵-۱	الگوی سیستم	۳۳
۵-۲	تحلیل نیازمندی‌های سیستم	۳۴
۵-۳	طراحی سیستم	۳۴
۵-۴	پیاده‌سازی سیستم	۳۵
۵-۴-۱	هسته سیستم	۳۶
۵-۴-۲	اتصال به پایگاه دانش	۳۶
۵-۴-۳	سرویس و رابط کاربری تحت وب	۳۷
۶	آزمایش و ارزیابی سیستم	۴۱
۶-۱	معیارهای ارزیابی	۴۲
۶-۱-۱	صحت	۴۲
۶-۱-۲	دقت و فراخوانی و امتیاز $F_1$	۴۳
۶-۲	روش اعتبارسنجی متقابل	۴۵
۶-۲-۱	اعتبارسنجی متقابل چندتایی	۴۵
۶-۳	نتایج ارزیابی سیستم	۴۶
۶-۳-۱	ارزیابی دسته‌بندهای رابطه	۴۶

۵۰	.....	۲-۳-۶ ارزیابی کتابخانه‌ی تشخیص دهنده موجودیت‌های نام‌دار
۵۲	.....	۳-۳-۶ ارزیابی یکپارچه‌ی سیستم
۵۴	.....	۴-۳-۶ ارزیابی مدل دنباله به دنباله
۵۷	.....	۷ جمع‌بندی و نتیجه‌گیری و پیشنهادات
۵۸	.....	۱-۷ جمع‌بندی و نتیجه‌گیری
۵۸	.....	۲-۷ کارهای آینده
۶۰	.....	منابع و مراجع

## فهرست اشکال

صفحه

شکل

۱-۲	ساختار انتزاعی گراف RDF	۸
۲-۲	نمونه‌ای از یک سه‌تایی RDF	۸
۳-۲	اتصالات یک شبکه‌ی عصبی پرسپترون با دو لایه‌ی پنهان	۱۱
۴-۲	نمودار تابع فعال‌ساز غیرخطی ReLU مورد استفاده در شبکه‌های پیچشی	۱۳
۵-۲	معماری کلی یک شبکه‌ی عصبی پیچشی	۱۴
۶-۲	معماری کلی یک شبکه‌ی عصبی بازگشتی	۱۵
۷-۲	نمونه‌ای از شبکه‌ی دنباله به دنباله	۱۷
۱-۴	تابع هسته‌ی دایره‌ای در ماشین بردار پشتیبان	۲۸
۲-۴	لایه‌های دسته‌بند شبکه عصبی پیچشی	۳۰
۳-۴	لایه‌های هسته‌ی دسته‌بند شبکه عصبی پیچشی	۳۱
۱-۵	نمودار Use case	۳۵
۲-۵	معماری سیستم	۳۶
۳-۵	صفحه کاربر عادی	۳۸
۴-۵	صفحه کاربر عادی همراه روابط غیرمنتخب	۳۹
۵-۵	صفحه کاربر تحلیل‌گر	۳۹
۶-۵	صفحه کاربر تحلیل‌گر همراه چند موجودیت	۴۰
۷-۵	صفحه کاربر تحلیل‌گر همراه چند موجودیت و روابط غیرمنتخب	۴۰
۱-۶	صحت شبکه دسته‌بند	۴۹
۲-۶	هزینه‌ی شبکه دسته‌بند	۴۹

## فهرست جداول

صفحه

جدول

۲۶	۱-۴ عنوان روابط و شناسه آنها در گراف دانش فارس بیس
۲۶	۲-۴ نمونه‌هایی از جملات الگو برای رابطه‌ی "پایتخت یک کشور"
۴۲	۱-۶ ماتریس درهم‌ریختگی
۴۷	۲-۶ ارزیابی دسته‌بند ماشین بردار پشتیبان
۴۸	۳-۶ نمونه‌های مشکل‌دار دسته‌بند ماشین بردار پشتیبان
۴۸	۴-۶ صحت و هزینه‌ی شبکه عصبی پیچشی
۵۰	۵-۶ ارزیابی دسته‌بندها در روش تعبیه‌سازی word2vec
۵۱	۶-۶ نمونه‌های مشکل‌دار دسته‌بند شبکه عصبی پیچشی
۵۱	۷-۶ ارزیابی کتابخانه‌ی تشخیص‌دهنده موجودیت‌های نام‌دار در سطح توکن
۵۲	۸-۶ ارزیابی کتابخانه‌ی تشخیص‌دهنده موجودیت‌های نام‌دار در سطح عبارت
۵۲	۹-۶ نمونه‌های مشکل‌دار کتابخانه‌ی تشخیص‌دهنده موجودیت‌های نام‌دار
۵۳	۱۰-۶ ارزیابی یکپارچه‌ی سیستم
۵۵	۱۱-۶ مقایسه‌ی مدل دنباله به دنباله و دسته‌بندهای سیستم در تشخیص رابطه
۵۶	۱۲-۶ ارزیابی مدل دنباله به دنباله



# فصل اول

## مقدمه

## ۱-۱ مقدمه

هوش مصنوعی یا هوش ماشینی به هوشمندی سیستم‌هایی گفته می‌شود که می‌توانند عملکردی مشابه با رفتارهای هوشمندانه‌ی انسان داشته باشند. این عبارت که در مقابل هوش طبیعی موجود در انسان بوجود آمده توسط آقای مک‌کارتی<sup>۱</sup>، استاد دانشگاه دارتموث<sup>۲</sup> در سال ۱۹۵۵ بیان گردیده است. رفتارهای هوش مصنوعی شامل درک شرایط، شبیه‌سازی فرایند تفکر، توانایی استدلال و یادگیری و کسب دانش می‌شود. ساختار مغزی انسان به قدری پیچیده است که چندین حوزه‌ی تخصصی برای بررسی، آموزش و تحلیل مسائل مرتبط با آن بوجود آمده است. به جرأت می‌توان گفت که الفاظی نظیر یادگیری، آموزش و تقلید (که همگی بر یادگیری تاکید دارند) از جذاب‌ترین حوزه‌هایی هستند که بشر به مطالعه آنها پرداخته و از عجایب و شگفتی‌های آن پرده برمی‌دارد. در این میان، دسته‌بندی‌هایی جزئی‌تر برای بررسی دقیق‌تر هریک از مسائل مرتبط با موارد فوق پایه‌گذاری شده‌اند. این دسته‌بندی شامل سه حوزه‌ی بینایی، پردازش زبان و مطالعه‌ی سیستم گفتاری انسان است. اگر موجودی شبیه انسان است، باید این سه ویژگی را به بهترین شکل در ساختار وجودی خود به همراه داشته باشد. این سه ویژگی، هرچند انسان کامل (از لحاظ درک) را نتیجه نخواهند داد، اما تا حد مطلوبی یک تقریب از انسان را نتیجه می‌دهد. علاوه بر این با دست‌یافتن به ساختارهایی مجهز به عملکرد شبیه به مغز انسان، می‌توان محصولات جدیدی را خلق کرد که بخش زیادی از نیازهای انسان را پاسخ‌گو خواهند بود. دستیابی به چنین ساختارهایی اگرچه جذاب، بسیار دشوار می‌باشد. یکی از اصلی‌ترین چالش‌های موجود بحث درک<sup>۳</sup> و فهماندن یا آموزش آن به ساختارهای مجهز به تکنولوژی هوش مصنوعی است. درک ابعاد مختلفی دارد و از آن میان می‌توان به درک تصویر، درک زبان طبیعی و درک گفتار اشاره کرد. آنچه که در این پروژه مورد بررسی قرار می‌گیرد در زیر دسته‌ی اصلی پردازش زبان طبیعی قرار دارد. در ادامه به تعریف مسئله پرداخته می‌شود.

## ۲-۱ تعریف مسئله

جستجوی اطلاعات یکی از فعالیت‌های اصلی کاربران در فضای وب است و وقتی اطلاعات مد نظر خاص و کوتاه می‌شود موتورهای جستجو نمی‌توانند پاسخگوی نیاز کاربران باشند. در این شرایط سیستم‌های پرسش و پاسخ<sup>۴</sup> به بهترین نحو می‌توانند نیاز کاربر را برآورده کنند. سیستم‌های پرسش و پاسخ در دو

<sup>۱</sup> John McCarthy

<sup>۲</sup> Dartmouth College

<sup>۳</sup> Perception

<sup>۴</sup> Question Answering System

نوع دسته‌بندی می‌شوند. نوع اول سیستم‌هایی هستند که از یک متن پاسخ سوال را استخراج می‌کنند و نوع دوم سیستم‌هایی هستند که از یک گراف دانش<sup>۵</sup> به عنوان پایگاه داده استفاده می‌کنند و پاسخ را از این گراف استخراج می‌کنند. محصول مد نظر این پروژه از نوع دوم بوده و گراف دانش فارسی فارس‌بیس اساس کار آن می‌باشد. گراف دانش یک پایگاه داده است که اطلاعات را به صورت یک گراف بیان می‌کند. در این گراف هر یک از گره‌ها بیانگر یک موجودیت<sup>۶</sup> است. بین هر دو موجودیت ممکن است رابطه‌ای وجود داشته باشد که این رابطه بوسیله‌ی اتصال یک یال بین دو گره‌ی مربوطه بیان می‌شود. سوالات گراف دانش به دو دسته‌ی سوالات ساده و پیچیده تقسیم می‌شوند. سوالات ساده سوال‌هایی هستند که برای پاسخ‌دهی به آنها تنها نیاز به یک رابطه است ولی سوالات پیچیده نیاز به استنتاج و ترکیب چند رابطه دارند. هدف از این پروژه طراحی سامانه‌ای است که کاربر بتواند بوسیله‌ی پرسش سوال اطلاعات مد نظر خود را کسب کند. این سیستم محدود به سوالات ساده‌ی فارسی می‌باشد و سوالات پیچیده در آن پشتیبانی نمی‌شود.

### ۳-۱ تعریف زیرمسئله‌ها

روش کار این سامانه به این صورت است که ابتدا از سوال ورودی اطلاعات مورد نیاز را استخراج می‌کند، سپس بوسیله‌ی این اطلاعات کوئری مورد نظر برای درخواست از گراف دانش را تولید کرده و پاسخ را از گراف دریافت برای کاربر نمایش می‌دهد. هسته این سامانه دارای سه بخش زیر می‌باشد:

- تشخیص رابطه<sup>۷</sup>

- تشخیص موجودیت‌ها<sup>۸</sup>

- تولید کوئری<sup>۹</sup>

#### ۱-۳-۱ تشخیص رابطه

این سامانه از دسته‌های "پایتخت یک کشور"، "کارگردان یک فیلم"، "همسر یک شخص" و ۴۵ دسته‌ی دیگر پشتیبانی می‌کند. هر کدام از این دسته‌ها یک نگاشت از یک نوع رابطه در گراف دانش مورد

---

Knowledge Graph<sup>۵</sup>

Entity<sup>۶</sup>

Relation Classification<sup>۷</sup>

Entity Recognition<sup>۸</sup>

Query Generation<sup>۹</sup>



استفاده می‌باشد. به عنوان مثال پرسش "پایتخت ایران کجاست؟" از دسته‌ی "پایتخت یک کشور" می‌باشد که نماینده‌ی رابطه‌ی "capital" در گراف دانش مورد استفاده می‌باشد. وظیفه‌ی اول سامانه تشخیص رابطه‌ی مورد پرسش در سؤال است که توسط یک دسته‌بند<sup>۱۰</sup> قابل انجام می‌باشد. برای انجام این بخش از دو دسته‌بند ماشین بردار پشتیبان<sup>۱۱</sup> و شبکه‌ی عصبی پیچشی<sup>۱۲</sup> استفاده شده و هر کدام به صورت جداگانه مورد آزمایش قرار گرفته شده‌اند.

### ۲-۳-۱ تشخیص موجودیت‌ها

وظیفه‌ی دوم سامانه استخراج موجودیت مورد پرسش سؤال می‌باشد. برای این منظور بایستی که تمامی موجودیت‌های نام‌دار سؤال تشخیص داده شوند. منظور از موجودیت نام‌دار انواع اسامی از جمله اشخاص، آثار، سازمان‌ها و مکان‌ها می‌باشد. به عنوان مثال جمله‌ی روبرو را در نظر بگیرید: "علی دایی متولد اردبیل است و سابقه‌ی بازی در پرسپولیس را دارد." مجموعه کلمات "علی دایی"، "اردبیل" و "پرسپولیس" موجودیت‌های نام‌دار این جمله می‌باشند.

با تشخیص موجودیت‌های نام‌دار جمله‌ی سؤال می‌توان موجودیت مورد پرسش و گره‌ی مربوط به آن در گراف دانش را استخراج نمود. به عنوان مثال در جمله‌ی "پایتخت ایران کجاست؟" کلمه‌ی "ایران" تنها موجودیت نام‌دار جمله است که همان موجودیت مورد پرسش سؤال می‌باشد.

### ۳-۳-۱ تولید کوئری

سامانه در مرحله‌ی سوم از رابطه و موجودیت استخراج شده از سؤال استفاده و کوئری SPARQL مناسب را تولید کرده و بوسیله‌ی آن پاسخ سؤال را از گراف دانش استخراج می‌نماید.

## ۴-۱ خلاصه فصل‌های بعد

در فصل بعدی با مفاهیم اولیه هوش مصنوعی و مدل‌های یادگیری ماشین مرتبط آشنا خواهیم شد. در ادامه تعدادی از مطالعات گذشته مرور شده و به توضیح سیستم پیشنهادی این پروژه می‌پردازیم. در نهایت قسمت‌های مختلف سیستم مورد ارزیابی قرار گرفته و پیشنهادات برای بهبود آن مطرح می‌شوند.

---

<sup>۱۰</sup> Classifier

<sup>۱۱</sup> Support Vector Machine

<sup>۱۲</sup> Convolutional Neural Network

# فصل دوم

## مفاهیم پایه

در این فصل به توضیح و مرور مفاهیم مقدماتی و پایه‌ای مرتبط با این پروژه می‌پردازیم. با توجه به اینکه بخشی از روش مورد استفاده بر مبنای شبکه عصبی پیچشی بوده و همچنین در قسمت ارزیابی از یک مدل دنباله به دنباله استفاده می‌شود، توضیحاتی در مورد شبکه‌های عصبی در انتهای این بخش ارائه خواهد شد.

## ۱-۲ پردازش زبان طبیعی

پردازش زبان طبیعی<sup>۱</sup> یکی از زیرشاخه‌های مهم حوزه‌ی هوش مصنوعی و دانش زبان‌شناسی است. تلاش عمده در این زمینه، ایجاد توانایی درک مفاهیم بیان شده در یک زبان طبیعی برای یک ماشین می‌باشد که می‌توان آنرا به دو قسمت پردازش زبان گفتار و نوشتار تقسیم نمود. با ورود فناوری رایانه‌ای به زندگی بشر، پردازش زبان طبیعی از مهمترین امور مورد توجه بوده است. برای ایجاد توانایی فهم زبان طبیعی توسط ماشین نیاز به دانش وسیعی از زبان می‌باشد که موجب نیاز به دانش زبان‌شناسان علاوه بر محققان علوم رایانه می‌گردد. کاربردهای پردازش زبان طبیعی به دو دسته‌ی کاربردهای نوشتاری و کاربردهای گفتاری قابل تقسیم است. از کاربردهای نوشتاری آن می‌توان به استخراج اطلاعاتی خاص از یک متن، ترجمه یک متن به زبانی دیگر یا یافتن مستندات خاص در یک پایگاه داده نوشتاری و از کاربردهای گفتاری آن می‌توان به سرویس‌های خودکار ارتباط با مشتری از طریق تلفن و سیستم‌های کنترلی توسط صدا اشاره کرد. سیستمی که در این پروژه بررسی می‌شود، از دسته‌ی کاربردهای نوشتاری این زمینه می‌باشد.

## ۲-۲ بازیابی اطلاعات

به استخراج اطلاعات مورد نیاز از یک منبع اطلاعاتی، بازیابی اطلاعات<sup>۲</sup> می‌گویند. این استخراج بر پایه‌ی یک جستجو انجام می‌شود که می‌تواند روی کل داده‌های متنی، فراداده‌ها و یا پایگاه‌های داده انجام گیرد. سیستم مورد بررسی در این پروژه از یک گراف دانش به عنوان منبع اطلاعات استفاده کرده و بوسیله‌ی کوئری‌های SPARQL اطلاعات مورد نیازش را از آن استخراج می‌کند.

<sup>۱</sup> Natural Language Processing  
<sup>۲</sup> Information Retrieval

## ۳-۲ گراف دانش

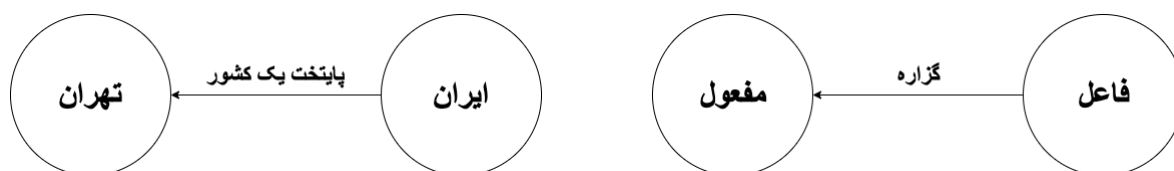
گراف‌های دانش مجموعه‌ی بزرگی از موجودیت‌های<sup>۳</sup> بهم مرتبط هستند که به وسیله‌ی برچسب‌های معنایی غنی شده‌اند. در واقع گراف دانش، پایگاه دانشی از حقیقت‌ها<sup>۴</sup> راجع به موجودیت‌هاست که از جمله‌ی آن‌ها می‌توان به ویکی‌دیتا<sup>۵</sup>، فریبیس<sup>۶</sup> و یاگو<sup>۷</sup> اشاره کرد که معمولاً از دانشنامه‌هایی مانند ویکی‌پدیا<sup>۸</sup>، جئونیمز<sup>۹</sup> و وردنت<sup>۱۰</sup> استخراج می‌گردند. از زمینه‌هایی که گراف دانش در آن‌ها کاربرد دارد می‌توان به موتورهای جستجو، پردازش زبان‌های طبیعی، استخراج آزاد اطلاعات و سامانه‌های پرسش و پاسخ اشاره کرد. بسیاری از سیستم‌های اطلاعاتی روی وب که نیازمند دسترسی به دانش ساخت‌یافته می‌باشند، از گراف دانش استفاده می‌کنند. وب معنایی<sup>۱۱</sup> در ابتدا دانش را بر مبنای گراف ارائه نمود که گره‌های آن، موجودیت‌ها و یال‌های آن رابطه‌ی میان موجودیت‌ها می‌باشند. با مطرح شدن داده‌های پیوندی<sup>۱۲</sup>، اتصال مجموعه داده‌های مختلف به یکدیگر در وب معنایی عنوان شد. بنابراین آینده‌ی وب معنایی، یک پایگاه دانش جهانی بسیار بزرگ و مستقل از زبان شامل موجودیت‌هایی است که به‌طور معنایی به‌هم مرتبط شدند.

اساس وب معنایی، مدل داده‌ای چارچوب توصیف منابع<sup>۱۳</sup> یا به اختصار RDF است. در فناوری وب معنایی، بیشتر داده‌ها از قبیل داده‌های پیوندی، به‌صورت RDF ذخیره و نمایش داده می‌شوند. از این جهت شاید بتوان گفت که مدل داده‌ای وب معنایی، RDF هست. در RDF هر منبع یا موجودیت دارای یک شناسه به عنوان نام یا آدرس یکتا است. این آدرس یکتا باید از طریق پروتکل HTTP قابل جستجو بوده و اطلاعات مفیدی را به شکل استاندارد فراهم نماید. این مدل، همانند مدل داده‌ای جدولی در پایگاه‌های داده‌ای رابطه‌ای نیست، همچنین مانند ساختار درختی XML نیز نیست بلکه RDF یک گراف است. داده‌های وب معنایی معمولاً به صورت سه‌تایی<sup>۱۴</sup> توصیف می‌شوند که شامل سه جزء فاعل<sup>۱۵</sup>،

Entity<sup>۳</sup>Facts<sup>۴</sup>Wikidata<sup>۵</sup>Freebase<sup>۶</sup>Yago<sup>۷</sup>Wikipedia<sup>۸</sup>GeoNames<sup>۹</sup>WordNet<sup>۱۰</sup>Semantic Web<sup>۱۱</sup>Linked Data<sup>۱۲</sup>Resource Description Framework<sup>۱۳</sup>Triple<sup>۱۴</sup>Subject<sup>۱۵</sup>

گزاره<sup>۱۶</sup> و مفعول<sup>۱۷</sup> است [۱۴].

هر سه‌تایی را می‌توان به صورت یک زنجیر گره-یال-گره مانند شکل ۱-۲ نمایش داد. به عنوان مثال حقیقتی مانند «پایتخت ایران شهر تهران است» را می‌توان به صورت یک سه‌تایی توصیف نمود. این توصیف براساس ساختار انتزاعی مذکور، در شکل ۲-۲ نمایش داده شده است.



شکل ۲-۲: نمونه‌ای از یک سه‌تایی RDF

شکل ۱-۲: ساختار انتزاعی گراف RDF

## ۴-۲ سیستم پرسش و پاسخ

سیستم پرسش و پاسخ<sup>۱۸</sup> به سیستمی گفته می‌شود که بتواند سوالات مطرح شده توسط یک انسان در زبان طبیعی را بررسی کرده و به آن به صورت خودکار پاسخ دهد. اینگونه سیستم‌ها در زمینه‌ی پردازش زبان طبیعی و بازیابی اطلاعات بررسی می‌شوند.

یک روش دسته‌بندی سیستم‌های پرسش و پاسخ مبتنی بر گراف دانش، بر اساس تعداد روابطی از گراف است که برای پاسخ‌دهی به سوال استفاده می‌شود.

دسته اول سوالات ساده هستند که برای پاسخ‌دهی به این سوالات فقط نیاز به یک رابطه از گراف دانش می‌باشد. به عنوان مثال در سوال "نام آثار سعدی چه هستند؟" با بررسی رابطه‌ی "آثار" از موجودیت "سعدی" در گراف دانش می‌توان پاسخ‌های این سوال را استخراج نمود. سیستمی که در این پروژه بررسی می‌شود از این دسته می‌باشد.

دسته دوم سوالات پیچیده هستند که برای پاسخ دادن به آنها به بیش از یک رابطه از گراف دانش نیاز است. به عنوان مثال برای پاسخ دادن به سوال "علی دایی در کدام تیم‌های لیگ آلمان بازی کرده است؟" ابتدا باید رابطه‌ی "باشگاه" از موجودیت "علی دایی" بررسی شده و سپس رابطه‌ی "لیگ" روی هر کدام از نتایج رابطه‌ی اول بررسی شود.

<sup>۱۶</sup> Predicate

<sup>۱۷</sup> Object

<sup>۱۸</sup> Question Answering System

## ۵-۲ شبکه‌های عصبی

در این قسمت به توضیح شبکه‌های عصبی و پایه‌ی ریاضی آنها و همچنین تشابه آنها با ریشه‌ی زیستی و طبیعی این شبکه‌ها پرداخته می‌شود. در ادامه نیز با انواعی از اینگونه شبکه‌ها آشنا خواهیم شد. یکی از روش‌های نوین در یادگیری ماشینی<sup>۱۹</sup>، شبکه‌های عصبی یا به عبارتی شبکه‌های عصبی مصنوعی<sup>۲۰</sup> بوده که می‌توان از آنها برای برای اعمال دانش بدست آمده از سامانه‌های پیچیده استفاده نمود. ساختار این شبکه‌ها مشابه ساختار زیستی مغز طبیعی بوده و از آن الهام گرفته است. هسته‌ی اصلی پردازشی این شبکه‌ها نرون‌های مصنوعی<sup>۲۱</sup> هستند که در شمار فراوان در این شبکه‌ها ظاهر شده و می‌توانند به صورت کاملاً به هم پیوسته<sup>۲۲</sup> یا غیرکاملاً به هم پیوسته<sup>۲۳</sup> کنار هم قرار گیرند. در شبکه‌های طبیعی، ارتباطات کاملاً به هم پیوسته بوده و در صورت آسیب دیدن یک نرون، نرون دیگری می‌تواند کار آن را به عهده بگیرد. نرون‌های مصنوعی همانند گونه‌ی طبیعی خود قابلیت یادگیری دارند. یادگیری یک شبکه عصبی مصنوعی می‌تواند به صورت با نظارت<sup>۲۴</sup> و یا بدون نظارت<sup>۲۵</sup> انجام گیرد. شبکه‌هایی که در ادامه بررسی می‌شوند تحت روش یادگیری با نظارت آموزش می‌بینند.

### ۱-۵-۲ تعریف پایه

یک شبکه‌ی عصبی دارای چندین لایه از نرون‌ها می‌باشد. عموماً هر شبکه دارای سه لایه‌ی اصلی ورودی، پردازش و خروجی است. به طور معمول نرون‌های هر لایه به تمامی نرون‌های لایه‌های قبل و بعد خود متصل می‌شود مگر آنکه به طور مخصوص این ارتباط محدود گردد. در عین حال هیچ نرونی با نرون‌های هم‌لایه‌ی خود اتصالی برقرار نمی‌کند. نرون کوچکترین واحد پردازشگر در شبکه است و یک شبکه مجموعه‌ای از نرون‌ها می‌باشد که با معماری خاصی کنار هم قرار گرفته و متصل شده‌اند. رفتار یک شبکه حاصل از مجموع رفتار همگی نرون‌های آن شبکه است که به طور مستقل در حال پردازش می‌باشند. هر نرون می‌تواند یک تابع ریاضی خطی یا غیرخطی باشد. در نتیجه یک شبکه‌ی عصبی می‌تواند روابط ریاضی بسیار پیچیده‌ای را برای استفاده فراهم نماید.

<sup>۱۹</sup> Machine Learning

<sup>۲۰</sup> Artificial Neural Networks

<sup>۲۱</sup> Artificial Neurons

<sup>۲۲</sup> Fully Connected

<sup>۲۳</sup> Not Fully Connected

<sup>۲۴</sup> Supervised Learning

<sup>۲۵</sup> Unsupervised Learning

## ۲-۵-۲ شبکه‌های عصبی پرسپترون و پرسپترون چندلایه

پرسپترون ساده‌ترین نوع شبکه‌های عصبی است که یک دسته‌بند دودویی<sup>۲۶</sup> به‌شمار می‌آید. فرض کنید که ورودی‌های شبکه‌ی عصبی  $d$  بعدی بوده و به‌تعداد  $n$  داده در اختیار داشته باشیم. در این صورت یک پرسپترون ورودی خود  $x \in R^d$  را به مقدار خروجی  $f(x) \in R$  با رابطه‌ی زیر تبدیل می‌کند:

$$f(x) = \begin{cases} 1 & \text{if } \langle w, x \rangle + b > 0 \\ 0 & \text{otherwise} \end{cases} \quad (1-2)$$

که مقدار  $w \in R^d$  بردار وزن شبکه و  $b$  نشان‌دهنده‌ی بایاس آن است که وظیفه‌ی آن جابجایی مرز تصمیم‌گیری از مبدا است. نماد  $\langle w, x \rangle$  نشان‌دهنده‌ی ضرب داخلی دو بردار است که مقدار آن برابر با  $\sum_{i=1}^m w_i x_i$  می‌باشد. طی مرحله‌ی یادگیری، شبکه‌ی پرسپترون تلاش می‌کند تا با توجه به داده‌هایی که در اختیار می‌گیرد، مقدار بهینه برای بردار وزنش یا همان  $w$  و مقدار بایاسش یا همان  $b$  را پیدا کند.

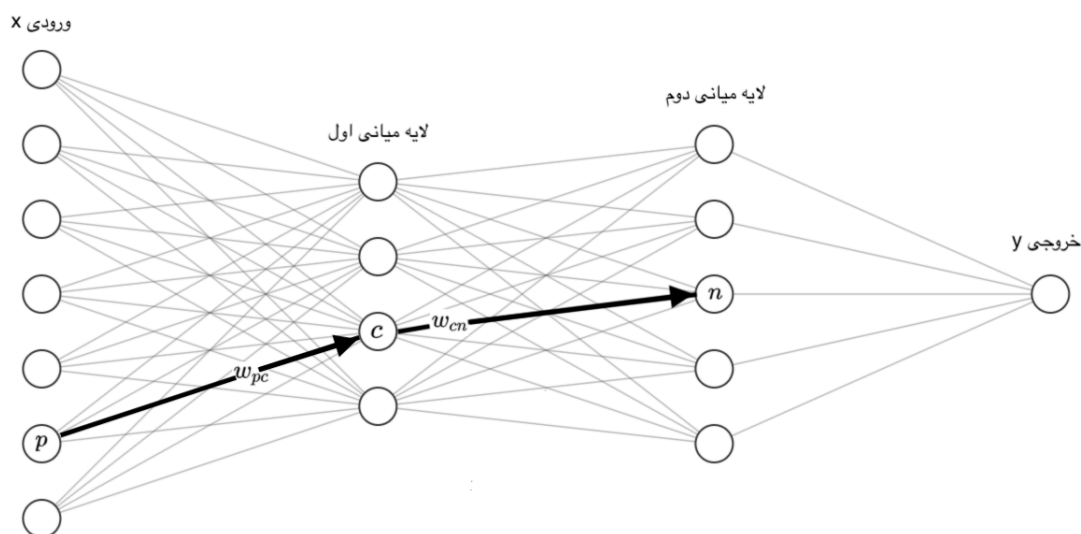
### پرسپترون چندلایه

یک پرسپترون چندلایه<sup>۲۷</sup> شامل حداقل سه‌لایه ورودی، پردازش (پنهان) و خروجی می‌باشد. در واقع هر شبکه‌ی عصبی با سه‌لایه یا بیشتر به یک پرسپترون چندلایه نام‌گذاری می‌شود. برای مثال در شکل ۲-۳ نمایشی از یک شبکه‌ی پرسپترون با دولایه‌ی پنهان آمده است. مقدار خروجی از نرون  $p$  را با  $b_p$  و وزن شبکه از نرون  $p$  به  $c$  را با  $w_{pc}$  نمایش می‌دهیم. همانطور که در شکل ۲-۳ مشخص است، نرون  $c$  مقادیر تمامی نرون‌های لایه‌ی ورودی را دریافت کرده و روی آنها پردازش انجام می‌دهد. مجموع ضرب ورودی‌های نرون  $c$  با وزن‌هایشان را با  $a_c$  نمایش می‌دهیم:

$$a_c = \sum_p w_{pc} b_p \quad (2-2)$$

---

Binary Classifier<sup>۲۶</sup>  
Multi-Layer Perceptron<sup>۲۷</sup>



شکل ۳-۲: اتصالات یک شبکه‌ی عصبی پرسپترون با دو لایه‌ی پنهان. گرادیان وزن‌های هر لایه وابسته به گرادیان وزن‌های لایه‌ی بالاتر می‌باشد.

حال جهت ایجاد امکان مدل کردن روابط غیرخطی در ورودی‌ها، باید روی  $a_c$  تابعی غیرخطی اعمال شود. در این صورت خواهیم داشت:

$$b_c = \theta_c(a_c) \quad (۳-۲)$$

که در آن  $\theta_c$  تابع غیرخطی اعمال شده و  $b_c$  خروجی جدید بدست آمده است. حال اگر تمامی وزن‌های شبکه‌ی پرسپترون را با  $W$  نمایش دهیم و خروجی شبکه پرسپترون به ازای داده‌ی  $x_i$  را  $h_W(x_i)$  فرض کنیم، تابع هدف به صورت:

$$Q(W) = \sum_{i=1}^n l(h_W(x_i), y_i) \quad (۴-۲)$$

خواهد بود که برای  $l$  می‌توان از خطای مربعات<sup>۲۸</sup> و یا منفی لگاریتم درست‌نمایی<sup>۲۹</sup> استفاده کرد. برای بدست آوردن کمینه‌ی  $Q(W)$  باید از روش گرادیان کاهشی<sup>۳۰</sup> استفاده کرد. به این معنی که گرادیان تابع را حساب کرده، کمی در خلاف جهت آن حرکت کرده و این کار را آنقدر ادامه داد تا تابع

<sup>۲۸</sup> Least Squared Error

<sup>۲۹</sup> Negative Log Likelihood

<sup>۳۰</sup> Gradient Descent



هزینه خیلی کوچک شود. روش انتشار معکوس<sup>۳۱</sup> در واقع روشی برای پیدا کردن گرادیان تابع  $Q(W)$  نسبت به تمامی وزن‌های شبکه می‌باشد.

## ۳-۵-۲ شبکه‌ی عصبی پیچشی

شبکه‌های پرسپترون چندلایه که در بخش قبل معرفی شدند با آنکه قابلیت یادگیری روی داده‌های غیرخطی را دارند اما در مسائلی با داده‌های حجیم مانند تصاویر دچار مشکل شده و با چنین ساختاری نیاز به تعداد پارامترهای زیادی دارند که موجب مشکل شدن فرایند یادگیری در اینگونه مسائل خواهد شد. برای رفع این مشکل از مفهوم و عملیات کانولوشن<sup>۳۲</sup> استفاده شده و شبکه‌هایی با نام شبکه‌ی عصبی پیچشی<sup>۳۳</sup> معرفی می‌شوند. عملیات پایه‌ای اینگونه شبکه‌ها عملیات کانولوشن می‌باشد. در این ساختار تعدادی هسته یا فیلتر وجود دارد که به آنها هسته‌ی لغزشی گفته می‌شود. این هسته‌ها همانند عملیات کانولوشن بر روی هر قسمت از داده‌ها قرار گرفته و پردازش انجام می‌دهند و این کار را روی همه‌ی قسمت‌های تصویر تکرار می‌کنند. هرکدام از این هسته‌ها دارای وزن‌هایی هستند که در مرحله‌ی یادگیری مورد آموزش قرار می‌گیرند. رابطه‌ی ۵-۲ نشان دهنده‌ی این عملیات است. در این رابطه  $x$  تصویر ورودی،  $y$  ترم بایاس،  $K$  اندازه‌ی هسته لغزشی،  $w$  وزن‌های آموزش (هسته‌ی لغزشی) و  $z_{i,j}$  خروجی عملیات کانولوشن است.

$$Z_{i,j} = y_{i,j} + \sum_{m=0}^{K-1} \sum_{n=0}^{K-1} x_{i+m,j+n} w_{m,n} \quad (5-2)$$

ترکیب این عملیات با اعمالی که در ادامه به آنها اشاره می‌کنیم، شبکه‌ی عصبی با قابلیت بالا در پردازش تصاویر را نتیجه خواهد داد. این اعمال عبارتند از:

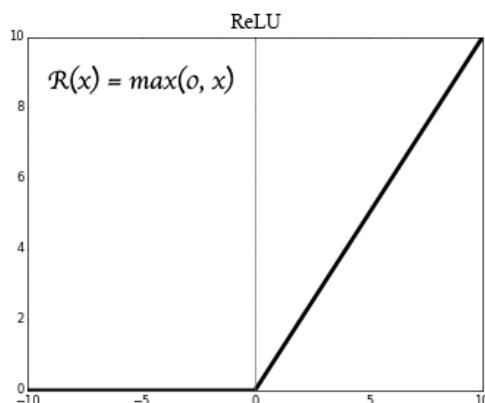
- پولینگ<sup>۳۴</sup>: عمل پولینگ یکی از اعمال اصلی شبکه‌های عصبی پیچشی بوده و به طور معمول از آن استفاده می‌شود. از این عملیات برای کاهش بعد داده هنگام پردازش استفاده می‌شود. این عمل روی قسمت‌های مختلف داده قرار گرفته و داده‌های آن قسمت را یکی می‌کند. برای انجام

<sup>۳۱</sup> Backpropagation

<sup>۳۲</sup> Convolution

<sup>۳۳</sup> Convolutional Neural Network

<sup>۳۴</sup> Pooling



شکل ۲-۴: نمودار تابع فعال‌ساز غیرخطی ReLU مورد استفاده در شبکه‌های پیچشی

آن روش‌های مختلفی وجود دارد که از انواع آن می‌توان به پولینگ بیشینه<sup>۳۵</sup> و پولینگ میانگین<sup>۳۶</sup> اشاره نمود. از مزیت‌های این عملیات بی‌اثر کردن شبکه عصبی پیچشی نسبت به تغییرات تصاویر نظیر حرکات انتقالی<sup>۳۷</sup> می‌باشد.

• توابع فعال‌ساز غیرخطی<sup>۳۸</sup>: این توابع مختص شبکه‌های پیچشی نیستند و در شبکه‌های پرسپترون نیز از قابلیت غیرخطی‌ای که فراهم می‌کنند استفاده می‌شود، هرچند کاربرد آنها در شبکه‌های عصبی پیچشی اهمیت بسیار بالایی پیدا می‌کند. از جمله مهمترین توابع فعال‌ساز غیرخطی مورد استفاده در شبکه‌های عصبی پیچشی می‌توان به تابع خطی رقیق شده<sup>۳۹</sup> یا به‌طور اختصاری ReLU اشاره داشت. ویژگی خاص این تابع که نمودار آن در ۲-۴ نشان داده شده است در این است که در عین حال که غیرخطی بوده و می‌تواند به یادگیری الگوهای پیچیده غیرخطی کمک کند، پردازش آن راحت بوده و مشتق آن بسیار ساده محاسبه می‌شود و استفاده از آن به‌صرفه‌تر از توابع غیرخطی‌ای نظیر تانژانت هایپربولیک<sup>۴۰</sup> می‌باشد.

نهایتاً با ترکیب عملیات کانولوشن و پولینگ، و تکرار این امر در زمان آموزش، هسته‌های شبکه‌ی پیچشی مانند الگوریتم‌های مبتنی بر هسته‌های دست‌ساز در پردازش تصاویر یادگرفته و عمل می‌کنند. به عنوان مثال، در دسته‌بندی تصاویر چهره‌ی انسان، هسته‌هایی برای شناسایی رنگ مو، حالت ابرو و حالت بینی چهره شکل می‌گیرد. این ویژگی‌ها با اینکه به لطف عملیات پولینگ ابعاد پایینی دارند اما

<sup>۳۵</sup> Max Pooling

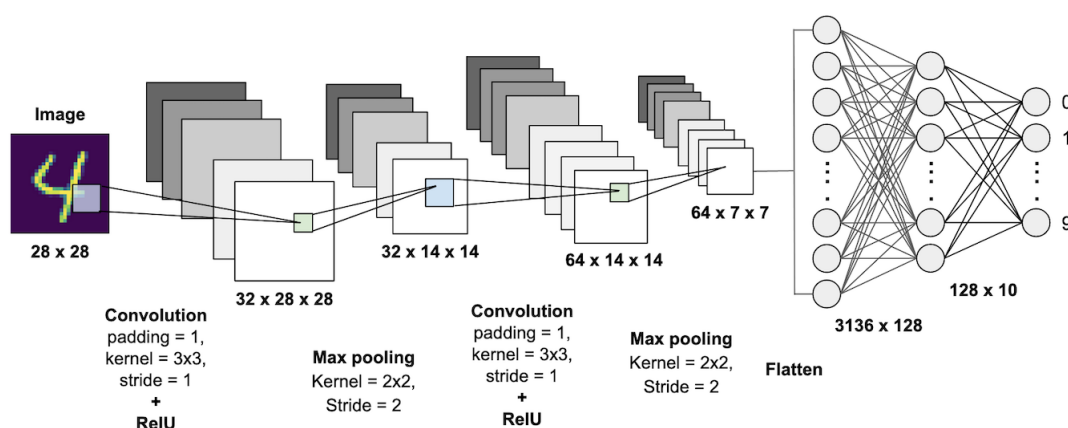
<sup>۳۶</sup> Average Pooling

<sup>۳۷</sup> Translation

<sup>۳۸</sup> Non-Linear Activation Functions

<sup>۳۹</sup> Rectified Linear Unit

<sup>۴۰</sup> Tanh



شکل ۲-۵: معماری کلی یک شبکه‌ی پیچشی در دسته‌بندی اعداد دست‌نوشته

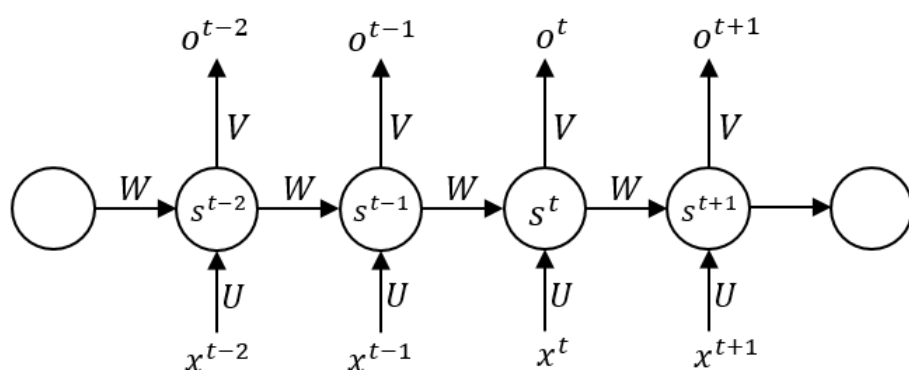
برای تشخیص چهره بسیار کاربردی بوده و استفاده می‌شوند. بنابراین در انتهای شبکه عصبی پیچشی از یک دسته‌بند مبتنی بر شبکه عصبی برای تشخیص خروجی استفاده می‌شود. نمونه‌ی چنین شبکه‌ای را که برای تشخیص اعداد دست‌نوشته استفاده می‌شود، در شکل ۲-۵ مشاهده می‌کنید. با توجه به وجود ۱۰ کلاس مختلف در مسئله‌ی تشخیص عدد دست‌نوشته، لایه‌ی دسته‌بند این شبکه که در انتهای آن قرار دارد دارای طول ۱۰ می‌باشد [۹].

## ۲-۵-۴ شبکه‌ی عصبی بازگشتی

ضعفی که در شبکه‌های معرفی شده در قسمت‌های قبل وجود دارد، ثابت بودن طول داده‌ی ورودی این شبکه‌ها می‌باشد. در برخورد با داده‌های زبانی، صوت و یا فیلم، توالی و ترتیب اطلاعات اهمیت بالایی پیدا می‌کند. شبکه‌ی بازگشتی<sup>۴۱</sup> این امکان را به ما می‌دهد تا یک توالی با طول نامشخص را به یک بردار با اندازه ثابت بازنمایی کنیم و در عین حال بسیاری از خواص نحوی و ساختاری توالی ورودی را حفظ کنیم. به طور کلی یک شبکه‌ی بازگشتی تابعی است که یک ورودی با طول غیرثابت (مثلاً جمله) را به صورت دنباله‌ای از  $n$  بردار با ابعاد  $R^{d_{in}}$  دریافت و یک بردار خروجی  $y$  با ابعاد  $R^{d_{out}}$  باز می‌گرداند. رابطه‌ی زیر این موضوع را به‌خوبی بیان می‌کند:

$$y_n = RNN(x_{1:n}) \quad (۲-۶)$$

Recurrent Neural Network<sup>۴۱</sup>



شکل ۶-۲: نمایی از معماری کلی یک شبکه‌ی عصبی بازگشتی متشکل از حالت‌های پنهان و خروجی‌ها

که در آن  $x_{1:n}$  جمله (دنباله) ورودی و  $y_n$  بردار خروجی است. شکل ۶-۲ نمایشی از یک شبکه‌ی عصبی بازگشتی با سلول‌های حافظه‌ای ساده است. در واقع این شبکه از دو بخش اصلی تشکیل شده است:

۱. **بخش بازگشتی:** این بخش با رابطه‌ی زیر نشان داده می‌شود و به آن حالت درونی یا پنهان شبکه در زمان  $t$  نیز گفته می‌شود:

$$s_t = R(x_t, s_{t-1}) \quad (۷-۲)$$

۲. **خروجی:** که با اعمال تابع فعال‌ساز بر حالت بخش بازگشتی بدست می‌آید:

$$o_t = O(s_t) \quad (۸-۲)$$

به طور خلاصه، شبکه بازگشتی هر کلمه را به یک بردار  $y_i$  تبدیل می‌کند ولی چون بردار مرحله آخر برگرفته از تمام اطلاعات قبلی است از آن به عنوان بازنمایی جمله استفاده می‌شود. روابط ۷-۲ و ۸-۲ را به طور دقیق‌تر می‌توان اینگونه بیان کرد:

$$s_t = \tanh(Ws_{t-1} + Ux_{t-1}) \quad (۹-۲)$$

$$o_t = y_t = \text{softmax}(Vs_t) \quad (۱۰-۲)$$

واضح است که خروجی زمان  $t$  در یک شبکه‌ی عصبی بازگشتی به ورودی‌ها و حالت‌های پنهان زمان‌های قبلی وابسته است. به همین علت، برای محاسبه‌ی گرادیان‌ها (تغییرات خطا نسبت به تغییر وزن‌ها) از الگوریتم انتشار معکوس در زمان<sup>۴۲</sup> استفاده می‌شود.

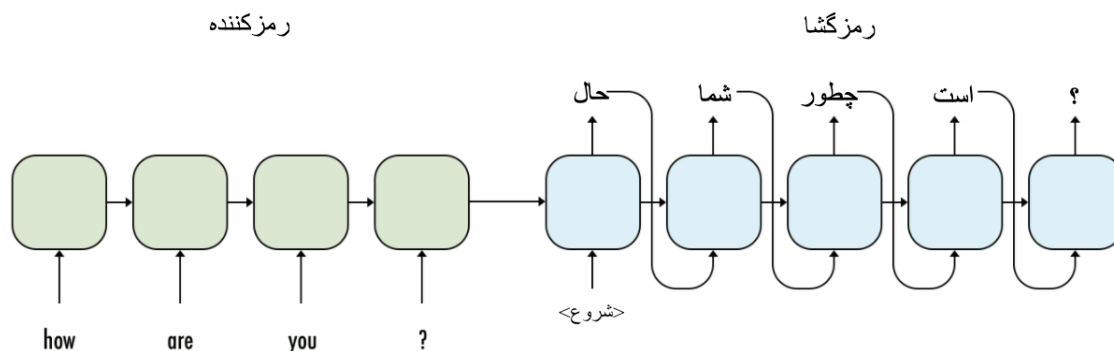
## ۵-۵-۲ شبکه‌ی عصبی دنباله به دنباله

در شبکه بازگشتی بوسیله‌ی یک حلقه بر روی بردار ورودی جلو می‌رفتیم و هر بار یک خروجی ایجاد می‌کردیم. مشکلی که وجود دارد در مواقعی است که بایستی در سطح کل بردار پیش برویم تا شبکه به درستی آموزش ببیند. به طور مثال وقتی بخواهیم مفهوم یک جمله را برداشت کنیم تا انتهای آن باید پیش برویم، بنابراین تنها خروجی و یا وضعیت آخر برای ما مهم است. موردی که یک شبکه‌ی بازگشتی قادر به انجام آن نیست، تولید یک دنباله از خروجی‌ها با توجه به تمامی اطلاعات ورودی می‌باشد. در چنین وضعیتی از یک شبکه‌ی عصبی دنباله به دنباله<sup>۴۳</sup> استفاده می‌کنیم [۵].

### ساختار شبکه

یک شبکه‌ی دنباله به دنباله از ترکیب دو شبکه‌ی بازگشتی ساخته می‌شود. شبکه اول که کدگذار<sup>۴۴</sup> نام دارد، دنباله‌ی بردار ورودی را به یک بردار وضعیت انتهایی تبدیل می‌کند و سپس آنرا به عنوان وضعیت اولیه برای شبکه بازگشتی دوم که کدگشا<sup>۴۵</sup> نام دارد، در نظر می‌گیرد. شبکه‌ی کدگشا با هر گام که جلو می‌رود یک خروجی تولید می‌کند. بدین صورت دنباله‌ی خروجی مورد نظر ساخته می‌شود. به عنوان مثال یک سیستم ترجمه‌کننده می‌تواند از این شبکه استفاده کند. به این صورت که ابتدا شبکه بازگشتی اول جمله زبان مبدأ را گرفته و آنرا به یک بردار وضعیت که نشان‌دهنده‌ی مفهوم جمله است، تبدیل می‌کند. سپس شبکه بازگشتی دوم این بردار وضعیت را دریافت کرده و جمله زبان مقصد را تولید می‌کند. در شکل ۷-۲ نمونه‌ای از این ماشین را مشاهده می‌کنید.

Backpropagation Through Time<sup>۴۲</sup>  
Sequence to Sequence Neural Network<sup>۴۳</sup>  
Encoder<sup>۴۴</sup>  
Decoder<sup>۴۵</sup>



شکل ۷-۲: نمونه‌ی یک ماشین ترجمه که از شبکه‌ی دنباله به دنباله تشکیل شده است

## ۶-۲ تابع هزینه‌ی Categorical Cross-Entropy

تابع Categorical Cross-Entropy یک تابع هزینه<sup>۴۶</sup> مناسب برای دسته‌بندی‌های تک کلاسه<sup>۴۷</sup> است. دسته‌بندی‌های تک کلاسه دسته‌بندی‌هایی هستند که هر داده‌ی آنها فقط متعلق به یک کلاس می‌باشد. این تابع هزینه به ازای هر داده بوسیله‌ی رابطه‌ی ۱۱-۲ بدست می‌آید که در آن  $\hat{y}$  نماد بردار احتمالات پیش‌بینی شده توسط دسته‌بند و  $y$  نماد بردار هدف می‌باشد.

$$L(y, \hat{y}) = - \sum_{i=0}^n (y_i * \log(\hat{y}_i)) \quad (۱۱-۲)$$

<sup>۴۶</sup> Loss Function  
<sup>۴۷</sup> One-Class Classification

## فصل سوم

### مروری بر کارهای گذشته

از آنجا که سیستم‌های پرسش و پاسخ در تمامی زبان‌ها از اهمیت بالایی برخوردار هستند تحقیقات و پروژه‌های بسیاری با روش‌های متنوع برای پاسخ‌دهی به این مسئله ارائه شده است. در ادامه به بررسی تعدادی از روش‌های پیاده‌سازی شده که سعی در ارائه‌ی پاسخ و یا بهبود اینگونه سیستم‌ها را داشته‌اند می‌پردازیم. برخی از این روش‌ها برای پاسخ‌دهی به سوالات ساده و برخی برای سوالات پیچیده طراحی شده‌اند و همگی آنها مبتنی بر پایگاه دانش می‌باشند.

### ۳-۱ روش‌های مبتنی بر ارائه‌ی معنایی

در این روش‌ها تلاش می‌شود تا از سوال ورودی بازنمایی‌ای<sup>۱</sup> استخراج شود که حاوی معنای نهفته در سوال باشد تا بر اساس آن بتوان به کوئری متناسب با سوال رسید. در یکی از این روش‌ها از تجزیه‌کننده‌های<sup>۲</sup> موجود برای زبان انگلیسی استفاده می‌شود. هدف تولید یک کوئری الگو<sup>۳</sup> متناسب با بازنمایی معنایی<sup>۴</sup> تولید شده می‌باشد. منظور از کوئری الگو یک کوئری SPARQL ناقص است که قالب آن کامل بوده اما بدون شناسه می‌باشد. یک کوئری الگو دارای جایگاه‌های مشخص برای قرار گرفتن شناسه‌های مرتبط با سوال می‌باشد. در این روش در تجزیه‌کننده‌ی مورد استفاده تغییراتی ایجاد شده که در حین تجزیه‌ی جمله بتواند کوئری الگو متناسب را تولید نماید. در مرحله‌ی بعدی موجودیت‌های جمله تشخیص داده شده و در جایگاه‌های مربوطه قرار می‌گیرند تا کوئری‌های نهایی تولید شوند. با امتیازدهی این کوئری‌ها برترین آنها انتخاب شده و استفاده می‌شود [۱].

در روشی دیگر بدون استفاده از تجزیه‌کننده‌های موجود یک بازنمایی معنایی تولید می‌شود. بدین منظور برخلاف روش قبلی، ابتدا موجودیت‌های جمله تشخیص داده شده و استخراج می‌شوند. سپس به روش تکرارشونده<sup>۵</sup> ارائه‌ی مورد نظر به صورت یک گراف تولید می‌شود. در هر تکرار این مرحله با توجه به گراف و اجزای جمله و موجودیت‌های تشخیص داده شده تعدادی عمل قابل انجام است که بوسیله‌ی آنها گره‌هایی به گراف اضافه می‌شوند. در نهایت با استفاده از این گراف یک کوئری تولید می‌شود [۲].

---

<sup>۱</sup> Representation

<sup>۲</sup> Parser

<sup>۳</sup> Template

<sup>۴</sup> Semantic

<sup>۵</sup> Iterative



## ۲-۳ روش‌های مبتنی بر تشخیص رابطه

همانطور که در فصل پیشین توضیح داده شد، گراف‌های دانش مجموعه‌ای از موجودیت‌ها هستند که توسط روابطی بهم متصل شده‌اند. در بسیاری از سیستم‌های پرسش و پاسخ هدف سیستم تشخیص موجودیت و رابطه‌ی مورد نظر سوال می‌باشد تا بوسیله‌ی آن بتوان با ایجاد کوئری موجودیت پاسخ را دریافت نمود. در یکی از روش‌های موجود کل فرایند به چهار قسمت شامل تشخیص موجودیت، پیوند دادن موجودیت، تشخیص رابطه و دریافت پاسخ تقسیم شده است که قسمت تشخیص رابطه به صورت یک دسته‌بند عمل می‌کند. در چنین روشی هر کدام از مراحل مستقل از بقیه‌ی مراحل می‌توانند بررسی شوند. هر کدام از سه مرحله‌ی اول توسط روش‌های مبتنی بر شبکه عصبی و یا غیر شبکه عصبی مورد آزمایش قرار گرفته‌اند و نتایج نشان از عملکرد نزدیک این دو دسته روش‌ها در این مسائل می‌دهد. در مرحله‌ی آخر این روش، از روابط و موجودیت‌های کاندید استفاده می‌شود تا موجودیت پاسخ مورد نظر یافت شود. این روش تنها توانایی پاسخ دادن به سوالات ساده را دارد [۱۱].

در روشی مشابه نیز ابتدا روابط و موجودیت‌های کاندید استخراج می‌شوند، اما برای توانایی پاسخ دادن به سوالات پیچیده از همگی این روابط و موجودیت‌ها هم‌زمان استفاده می‌شود. بدین منظور این روش تلاش می‌کند مسیری در گراف دانشی که استفاده می‌کند پیدا کند که همگی روابط و موجودیت‌های استخراج شده از سوال در آن مسیر وجود داشته باشد. چنین مسیری گره‌های دیگری را نیز دربر می‌گیرد که همان پاسخ‌های کاندید برای سوال می‌باشند [۴].

یکی از ایرادهای وارد بر دو روش فوق در این است که به دلیل مستقل بودن مراحل تشخیص موجودیت و تشخیص رابطه امکان تشخیص اشتباه بالا می‌رود. به عنوان مثال برای کلماتی که چندین موجودیت مشابه به آن‌ها در گراف دانش یافت می‌شود، با توجه به رابطه‌ی مورد پرسش می‌توان موجودیت مناسب‌تر را تشخیص داد. به همین دلیل در روشی دیگر روابط، موجودیت‌ها و سایر گره‌های گراف دانش همگی تحت عنوان یک عنصر در نظر گرفته شده و سعی می‌شود عناصر مرتبط با پرسش تولید شده و یک مسیر صحیح در گراف دانش متناسب با پرسش مورد نظر یافت شود [۱۳].

## ۳-۳ روش‌های تولیدکننده‌ی جمله‌ی پاسخ

در روش‌های معرفی شده در دو قسمت قبلی خروجی نهایی مدلهای یک موجودیت از پایگاه دانش است که به عنوان پاسخ به پرسش ورودی برگردانده می‌شود. اما در برخی روش‌ها به این پاسخ قناعت نشده

و تلاش می‌شود یک جمله‌ی کامل به عنوان پاسخ بازگردانده شود. به عنوان مثال در پاسخ به پرسش "پایتخت ایران کجاست" به جای کلمه یا موجودیت "تهران"، جمله‌ی "پایتخت ایران تهران است" نمایش داده می‌شود. برای پیاده‌سازی چنین سیستمی، در [۶] از یک شبکه‌ی دنباله به دنباله استفاده می‌شود. همانطور که مشخص است برخی از کلمات خروجی از پایگاه دانش استخراج می‌شوند و این شبکه نمی‌تواند به تنهایی تمامی جمله‌ی پاسخ را تولید نماید. کلمات خروجی را می‌توان به دو دسته تقسیم نمود؛ کلمات پایگاه که از پایگاه دانش استخراج می‌شوند و کلمات مشترک که برای تکمیل جمله تولید می‌شوند و نیازی به پایگاه برای تولید آنها نمی‌باشد. برای استفاده هم‌زمان این شبکه و پایگاه دانش، تغییراتی در قسمت کدگشای شبکه‌ی دنباله به دنباله داده شده است. در این قسمت در هر گام علاوه بر کلمه‌ی خروجی یک احتمال نیز تولید می‌شود که نشان می‌دهد آیا این کلمه از کلمات مشترک یا از کلمات پایگاه می‌باشد. در صورتی که تشخیص داده شود که در این گام یک کلمه‌ی پایگاه بایستی تولید شود، از پایگاه دانش استفاده شده و بوسیله‌ی موجودیت‌های یافت شده در جمله‌ی ورودی موجودیت مربوطه به جمله‌ی خروجی استخراج می‌گردد.

### ۴-۳ جمع‌بندی

همانطور که در این فصل توضیح داده شد یکی از رویکردهای متداول در این حوزه استفاده از دسته‌بندها برای تشخیص رابطه می‌باشد. در این پژوهش نیز سیستمی با این رویکرد پیشنهاد داده می‌شود که در ادامه به معرفی آن می‌پردازیم.

## فصل چهارم

# سیستم پیشنهادی برای پرسش و پاسخ فارسی

در این فصل سیستمی مناسب برای پاسخ‌دهی به مسائل شرح داده شده در فصل یک معرفی می‌شود. برای چنین سیستمی نیاز به یک پایگاه دانش می‌باشد تا اطلاعات مورد نیاز سیستم را فراهم نماید. برای فراهم کردن این اطلاعات از گراف دانش فارس بیس استفاده می‌شود. در ابتدای این فصل به معرفی این گراف دانش پرداخته شده و سپس بخش‌های دیگر سیستم معرفی می‌شوند.

## ۴-۱ گراف دانش فارس بیس

پایگاه دانش فارس بیس اولین پایگاه دانش طراحی شده مخصوص زبان فارسی می‌باشد. فارس بیس از مدل فراداده‌ای چارچوب توصیف منابع برای ذخیره‌ی دانش استفاده می‌کند. برای اتصال به این پایگاه دانش و دریافت موجودیت پاسخ از زبان SPARQL<sup>۱</sup> استفاده می‌شود. برخلاف پایگاه‌های دانش انگلیسی زبان همانند Wikidata که افراد مختلف بسیاری در گسترش آن کمک می‌کنند، در مورد پایگاه‌های فارسی زبان چنین اجتماع گسترده‌ای وجود ندارد و نمی‌توان به آن اتکا نمود؛ به همین سبب برای تولید پایگاه دانش فارس بیس از داده‌های خام متنی داخل وب استفاده شده و از طریق خزش وب دانش‌های موجود در آنها استخراج می‌گردند. دانش‌های استخراج شده به صورت ترکیب‌های سه‌تایی در پایگاه فارس بیس ذخیره می‌شوند. این ترکیب‌های سه‌تایی شامل فاعل، مفعول و رابطه‌ی بین آنها می‌باشد. به عنوان مثال با استخراج این دانش که مادرید پایتخت کشور اسپانیا می‌باشد، سه‌تایی «اسپانیا، مادرید، پایتخت یک کشور» در پایگاه دانش ایجاد می‌شود. سیستم استخراج‌کننده‌ی دانش فارس بیس دارای معماری‌ای با چندین لایه می‌باشد. استخراج دانش از ترکیب دو روش اصلی استخراج رابطه<sup>۲</sup> و استخراج اطلاعات<sup>۳</sup> صورت می‌گیرد. برای استخراج رابطه از دو ماژول و برای استخراج اطلاعات از چهار ماژول مختلف توسعه‌یافته برای زبان فارسی استفاده می‌شود. همچنین در هر دو روش یک مرحله‌ی پیوند موجودیت‌ها<sup>۴</sup> وجود دارد که فاعل و مفعول سه‌تایی مربوطه را ابهام‌زدایی کرده و به موجودیت‌های گراف دانش متصل می‌نماید. به موازات این مرحله، کانونی‌سازی<sup>۵</sup> نیز انجام می‌شود. در مرحله‌ی کانونی‌سازی رابطه‌ی بین فاعل و مفعول در اطلاعات استخراج شده بدست آورده می‌شود. این مرحله فقط در روش استخراج اطلاعات مورد نیاز است. در نهایت، همه‌ی سه‌تایی‌های دانش کاندید که استخراج شده‌اند، پس از بازبینی و تایید شدن توسط یک متخصص انسانی به پایگاه دانش فارس بیس افزوده می‌شوند [۸].

<sup>۱</sup> SPARQL Protocol and RDF Query Language

<sup>۲</sup> Relation Extraction

<sup>۳</sup> Information Extraction

<sup>۴</sup> Entity Linking

<sup>۵</sup> Canonicalization

سیستم پرسش و پاسخ فارسی مورد نظر از این پایگاه دانش استفاده می کند تا با داشتن رابطه و موجودیت مورد پرسش، سه تایی مربوطه در پایگاه دانش را پیدا کرده و پاسخ سوال را از آن استخراج نماید. برای انجام این کار نیاز است که سه زیرمسئله‌ی عنوان شده در فصل یک شامل تشخیص رابطه، تشخیص موجودیت‌ها و تولید کوئری توسط سیستم پاسخ داده شوند. برای پاسخ دادن به این زیرمسئله‌ها نیاز به مجموعه دادگان وجود دارد و با توجه به نبود دادگان برای پرسش و پاسخ فارسی، دادگان مورد نیاز در این پروژه تهیه شده‌اند. در ادامه با دادگان جمع‌آوری شده آشنا شده و سپس به شرح چگونگی انجام و حل سه زیرمسئله‌ی عنوان شده که سه بخش اصلی سیستم را تشکیل می‌دهند، می‌پردازیم.

## ۲-۴ مجموعه دادگان

به تعداد ۴۶ رابطه از گراف دانش فارسی‌بیس استخراج شده است که آنها را در جدول ۴-۱ مشاهده می‌نمایید. این روابط جز پرکاربردترین روابط در گراف دانش می‌باشند که توسط تهیه‌کنندگان گراف دانش فارسی مشخص شده‌اند.

عنوان	شناسه
پایتخت یک کشور	<a href="http://fkg.iust.ac.ir/ontology/capital">http://fkg.iust.ac.ir/ontology/capital</a>
واحد پول یک شرکت	<a href="http://fkg.iust.ac.ir/ontology/currency">http://fkg.iust.ac.ir/ontology/currency</a>
نوع حکومت یک کشور	<a href="http://fkg.iust.ac.ir/ontology/governmentType">http://fkg.iust.ac.ir/ontology/governmentType</a>
زبان رسمی یک کشور	<a href="http://fkg.iust.ac.ir/ontology/language">http://fkg.iust.ac.ir/ontology/language</a>
مساحت یک کشور	<a href="http://fkg.iust.ac.ir/ontology/areaTotal">http://fkg.iust.ac.ir/ontology/areaTotal</a>
جمعیت یک کشور	<a href="http://fkg.iust.ac.ir/ontology/populationTotal">http://fkg.iust.ac.ir/ontology/populationTotal</a>
کارگردان یک فیلم	<a href="http://fkg.iust.ac.ir/ontology/director">http://fkg.iust.ac.ir/ontology/director</a>
بازیگران یک فیلم	<a href="http://fkg.iust.ac.ir/ontology/starring">http://fkg.iust.ac.ir/ontology/starring</a>
نویسنده یک فیلم	<a href="http://fkg.iust.ac.ir/ontology/writer">http://fkg.iust.ac.ir/ontology/writer</a>
آهنگساز یک فیلم	<a href="http://fkg.iust.ac.ir/ontology/musicComposer">http://fkg.iust.ac.ir/ontology/musicComposer</a>
بودجه یک فیلم	<a href="http://fkg.iust.ac.ir/ontology/budget">http://fkg.iust.ac.ir/ontology/budget</a>
میزان فروش یک فیلم	<a href="http://fkg.iust.ac.ir/ontology/gross">http://fkg.iust.ac.ir/ontology/gross</a>
ترکیبات یک غذا	<a href="http://fkg.iust.ac.ir/ontology/ingredient">http://fkg.iust.ac.ir/ontology/ingredient</a>
آثار یک شخص	<a href="http://fkg.iust.ac.ir/ontology/notableWork">http://fkg.iust.ac.ir/ontology/notableWork</a>

<a href="http://fkg.iust.ac.ir/ontology/club">http://fkg.iust.ac.ir/ontology/club</a>	باشگاه های یک ورزشکار
<a href="http://fkg.iust.ac.ir/ontology/almaMater">http://fkg.iust.ac.ir/ontology/almaMater</a>	محل تحصیل یک شخص
<a href="http://fkg.iust.ac.ir/ontology/governorGeneral">http://fkg.iust.ac.ir/ontology/governorGeneral</a>	استاندار یک استان
<a href="http://fkg.iust.ac.ir/ontology/deathPlace">http://fkg.iust.ac.ir/ontology/deathPlace</a>	محل دفن یک شخص
<a href="http://fkg.iust.ac.ir/ontology/deathDate">http://fkg.iust.ac.ir/ontology/deathDate</a>	تاریخ مرگ یک شخص
<a href="http://fkg.iust.ac.ir/ontology/child">http://fkg.iust.ac.ir/ontology/child</a>	فرزند یک شخص
<a href="http://fkg.iust.ac.ir/ontology/property">http://fkg.iust.ac.ir/ontology/property</a>	سرمربی یک تیم ورزشی
<a href="http://fkg.iust.ac.ir/ontology/capital">http://fkg.iust.ac.ir/ontology/capital</a>	مرکز یک استان
<a href="http://fkg.iust.ac.ir/ontology/director">http://fkg.iust.ac.ir/ontology/director</a>	کارگردان یک سریال تلویزیونی
<a href="http://fkg.iust.ac.ir/ontology/starring">http://fkg.iust.ac.ir/ontology/starring</a>	بازیگران یک سریال تلویزیونی
<a href="http://fkg.iust.ac.ir/ontology/author">http://fkg.iust.ac.ir/ontology/author</a>	نویسنده یک سریال تلویزیونی
<a href="http://fkg.iust.ac.ir/ontology/Person/height">http://fkg.iust.ac.ir/ontology/Person/height</a>	قد یک ورزشکار
<a href="http://fkg.iust.ac.ir/ontology/areaTotal">http://fkg.iust.ac.ir/ontology/areaTotal</a>	مساحت یک دریا یا اقیانوس
<a href="http://fkg.iust.ac.ir/ontology/era">http://fkg.iust.ac.ir/ontology/era</a>	دوره ساخت یک مکان یا بنا
<a href="http://fkg.iust.ac.ir/ontology/spouse">http://fkg.iust.ac.ir/ontology/spouse</a>	همسر یک شخص
<a href="http://fkg.iust.ac.ir/ontology/county">http://fkg.iust.ac.ir/ontology/county</a>	محل یک مکان یا بنا
<a href="http://fkg.iust.ac.ir/ontology/nationality">http://fkg.iust.ac.ir/ontology/nationality</a>	ملیت یک شخص
<a href="http://fkg.iust.ac.ir/ontology/author">http://fkg.iust.ac.ir/ontology/author</a>	نویسنده کتاب
<a href="http://fkg.iust.ac.ir/ontology/publisher">http://fkg.iust.ac.ir/ontology/publisher</a>	ناشر کتاب
<a href="http://fkg.iust.ac.ir/ontology/areaCode">http://fkg.iust.ac.ir/ontology/areaCode</a>	پیش شماره
<a href="http://fkg.iust.ac.ir/ontology/areaTotal">http://fkg.iust.ac.ir/ontology/areaTotal</a>	مساحت
<a href="http://fkg.iust.ac.ir/ontology/populationTotal">http://fkg.iust.ac.ir/ontology/populationTotal</a>	جمعیت
<a href="http://fkg.iust.ac.ir/ontology/leaderName">http://fkg.iust.ac.ir/ontology/leaderName</a>	شهردار
<a href="http://fkg.iust.ac.ir/ontology/service">http://fkg.iust.ac.ir/ontology/service</a>	محصولات شرکت
<a href="http://xmlns.com/foaf/o.1/homepage">http://xmlns.com/foaf/o.1/homepage</a>	سایت یک شرکت
<a href="http://fkg.iust.ac.ir/ontology/ceo">http://fkg.iust.ac.ir/ontology/ceo</a>	مدیر عامل شرکت
<a href="http://fkg.iust.ac.ir/ontology/revenue">http://fkg.iust.ac.ir/ontology/revenue</a>	درآمد

<a href="http://fkg.iust.ac.ir/ontology/vehicleCode">http://fkg.iust.ac.ir/ontology/vehicleCode</a>	پلاک
<a href="http://fkg.iust.ac.ir/ontology/بزرگترین_شهر">http://fkg.iust.ac.ir/ontology/بزرگترین_شهر</a>	بزرگترین شهر یک کشور
<a href="http://fkg.iust.ac.ir/ontology/leader">http://fkg.iust.ac.ir/ontology/leader</a>	وزیر یک وزارتخانه
<a href="http://fkg.iust.ac.ir/ontology/numberOfStaff">http://fkg.iust.ac.ir/ontology/numberOfStaff</a>	تعداد کارکنان یک شرکت یا ارگان
<a href="http://fkg.iust.ac.ir/ontology/ground">http://fkg.iust.ac.ir/ontology/ground</a>	نام ورزشگاه یک تیم ورزشی

جدول ۴-۱: عنوان روابط و شناسه آنها در گراف دانش فارس بیس

به ازای هر کدام از روابط فوق مجموعه‌ای از جملات الگو به منظور آموزش به مدل‌ها در فایل rel\_q.csv جمع‌آوری شده‌اند. تعدادی از الگوهای رابطه‌ی "پایتخت یک کشور" را در جدول ۴-۲ مشاهده می‌نمایید.

پایتخت کشور ررر چه نام دارد
پایتخت ررر چه نام دارد
شهر ششش پایتخت کدام کشور است
نام کشوری که شهر ششش پایتخت آن است چیست
پایتخت کشور ررر چه می‌باشد

جدول ۴-۲: نمونه‌هایی از جملات الگو برای رابطه‌ی "پایتخت یک کشور"

در جملات الگو به جای موجودیت‌ها نماد آنها جایگزین شده است. به عنوان مثال در جملات جدول ۴-۲ "ررر" نماد موجودیت کشور و "ششش" نماد موجودیت شهر می‌باشد. هدف از انتخاب این نمادها اطمینان از عدم وجود آنها در واژگان زبان فارسی می‌باشد. به ازای هر یک از این نمادها تعدادی موجودیت مرتبط از پایگاه دانش فارس بیس استخراج شده و در جملات الگو قرار داده می‌شوند و از این جملات تولید شده برای آموزش مدل‌ها استفاده می‌شود.

## ۳-۴ تشخیص رابطه

این بخش از سیستم وظیفه‌ی تشخیص رابطه‌ی موجود در سوال مورد پرسش کاربر را دارد که بوسیله‌ی یک دسته‌بند قابل انجام است. به این منظور دسته‌بندهای ماشین بردار پشتیبان و شبکه عصبی پیچشی که در فصل دوم با آن آشنا شدیم بکار گرفته شده‌اند. در این بخش به طور مختصر چگونگی استفاده از

این دسته‌بندها برای این سیستم توضیح داده می‌شود. این دسته‌بندها در پکیج<sup>۶</sup> classifiers پیاده‌سازی شده‌اند. در این پکیج یک کلاس انتزاعی<sup>۷</sup> پیاده شده و کلاس‌های مربوط به مدل‌های ماشین بردار پشتیبان و شبکه عصبی پیچشی از آن ارث‌بری می‌کنند. برای استفاده از این دسته‌بندها نیاز به یک مرحله پیش‌پردازش روی سوالات ورودی می‌باشد که در ادامه به شرح آن می‌پردازیم.

### ۱-۳-۴ تعبیه‌سازی کلمات

برای استفاده از دسته‌بندها ابتدا باید سوالات ورودی به بردارهایی نگاشت شوند و از آن بردارها به عنوان ورودی دسته‌بند استفاده شود. برای این منظور از چهار روش تعبیه‌سازی<sup>۸</sup> کلمات شامل کوله‌ی کلمات<sup>۹</sup> که آنرا با نماد BOW نمایش می‌دهیم، روش TF-IDF<sup>۱۰</sup> و یک مدل word2vec استفاده شده است. روش چهارم استفاده از همان مدل word2vec به صورت وزن‌دار است که وزن‌های آن همان مقادیر معکوس فراوانی متن استفاده شده در روش TF-IDF برای هر کلمه می‌باشد. این مدل‌ها در فایل feature\_extractor.py پیاده‌سازی شده‌اند.

مدل ماشین بردار پشتیبان با هر چهار روش فوق و مدل شبکه عصبی پیچشی با روش word2vec بدون وزن قابل استفاده بوده و مورد آزمایش قرار گرفته‌اند.

### ۲-۳-۴ دسته‌بند ماشین بردار پشتیبان

ماشین بردار پشتیبان یک روش یادگیری با نظارت است که از آن برای دسته‌بندی و رگرسیون<sup>۱۱</sup> می‌توان استفاده کرد. مبنای کار ماشین بردار پشتیبان دسته‌بندی خطی داده‌ها است به طوری که خطی به عنوان تقسیم‌کننده استفاده شود که بیشترین حاشیه اطمینان را داشته باشد. حل این مسئله منجر به یک مسئله‌ی بهینه‌سازی برنامه‌ریزی درجه دوم<sup>۱۲</sup> مقید می‌گردد. همچنین می‌توان از این ماشین برای دسته‌بندی داده‌های جدایی‌ناپذیر خطی هم استفاده کرد؛ به طوریکه قبل از استفاده از ماشین بردار پشتیبان، داده‌ها توسط تابعی تحت عنوان تابع هسته<sup>۱۳</sup> به فضایی دیگر انتقال داده می‌شوند که در آن

<sup>۶</sup>Package

<sup>۷</sup>Abstract

<sup>۸</sup>Embedding

<sup>۹</sup>Bag of Words

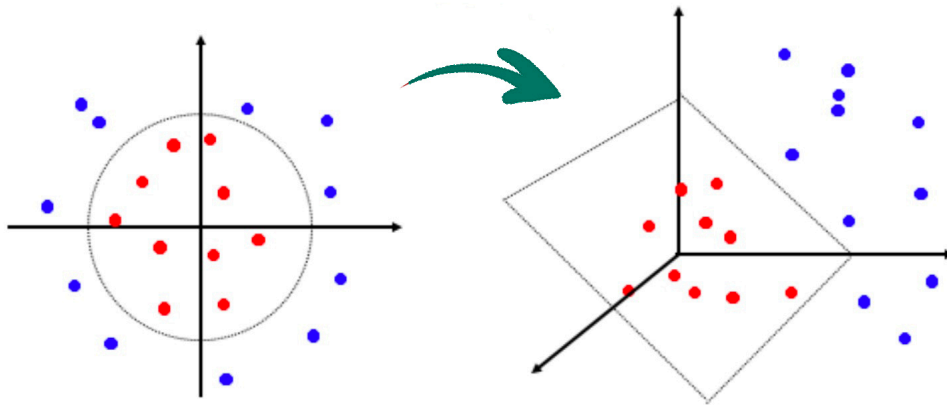
<sup>۱۰</sup>Term Frequency-Inverse Document Frequency

<sup>۱۱</sup>Regression

<sup>۱۲</sup>Quadratic Programming

<sup>۱۳</sup>Kernel





شکل ۴-۱: جداپذیری خطی کردن داده‌ها با انتقالشان از فضای دوبعدی به سه‌بعدی بوسیله‌ی یک هسته‌ی تابع پایه شعاعی

فضا جدایی‌پذیر خطی باشند. یک نمونه از این انتقال در شکل ۴-۱ نمایش داده شده است. در این سیستم از تابع پایه شعاعی<sup>۱۴</sup> به عنوان هسته‌ی این دسته‌بند استفاده می‌شود.

### ۴-۳-۳ دسته‌بند شبکه عصبی پیچشی

پایه‌سازی این شبکه توسط کتابخانه‌ی متن‌باز Keras انجام شده است. این شبکه یک جمله را به صورت یک بردار تک‌بعدی دریافت می‌کند که هر خانه‌ی این آرایه کد کلمه‌ی متناظر در لیست واژگان را مشخص می‌کند. خروجی شبکه نیز یک بردار تک‌بعدی به طول ۴۶ می‌باشد که احتمال هر رابطه را برای جمله‌ی ورودی مشخص می‌نماید. همانطور که در شکل ۴-۲ مشاهده می‌کنید، اولین لایه‌ی شبکه یک لایه‌ی تعبیه‌ساز است که از مدل word2vec توضیح داده شده استفاده می‌کند تا هر کلمه را به بردار متناسب آن نگاشت کند. لایه‌ی سوم که هسته‌ی این شبکه می‌باشد در ادامه توضیح داده خواهد شد. در ادامه دو لایه شبکه از نوروں‌ها (Dense) همراه با تابع فعال‌ساز<sup>۱۵</sup> ReLU حضور دارند که در نهایت بوسیله‌ی یک لایه‌ی فعال‌ساز Softmax احتمالات هر یک از کلاس‌ها که خروجی دسته‌بند است تولید می‌شود. در میان این لایه‌ها دو لایه‌ی حذفی تصادفی<sup>۱۶</sup> حضور دارند که فقط هنگام آموزش فعال هستند و هدف آنها کاهش بیش‌بردارش<sup>۱۷</sup> در هنگام آموزش می‌باشد. هسته‌ی این شبکه، یک شبکه‌ی پیچشی است که وظیفه‌ی آن تشخیص الگوهای جملات ورودی است.

<sup>۱۴</sup>Radial Basis Function (RBF)

<sup>۱۵</sup>Activation

<sup>۱۶</sup>Dropout

<sup>۱۷</sup>Overfitting

همانطور که در شکل ۴-۳ مشاهده می‌کنید، این شبکه از سه قسمت موازی مشابه تشکیل می‌شود. در هر قسمت ابتدا یک لایه شبکه‌ی کانولوشن وجود دارد که در هر کدام از آنها ظرفیت لازم برای تشخیص ۱۵۰ فیلتر در نظر گرفته شده است. اما اندازه‌ی هسته یا فیلتر آنها متفاوت بوده و طول آن در هر کدام به ترتیب سه، چهار و پنج کلمه در نظر گرفته شده است. در ادامه در هر سه قسمت مقادیر به یک لایه‌ی ادغام حداکثر<sup>۱۸</sup> به طول دو وارد شده و سپس تمامی مقادیر بهم الحاق شده و از این شبکه‌ی داخلی خارج می‌شوند [۷].

از تابع هزینه‌ی Categorical Cross-Entropy برای آموزش این شبکه استفاده شده است.

## ۴-۴ تشخیص موجودیت‌ها

برای تشخیص موجودیت‌های جمله‌ی مورد پرسش دو امکان در این سیستم موجود است. امکان اول استفاده از رابط‌های برنامه‌نویسی کاربردی<sup>۱۹</sup> موجود در سیستم فارس‌بیس است که امکان تشخیص موجودیت‌های موجود در جمله را دارند. امکان دوم استفاده از کتابخانه‌ی تشخیص‌دهنده موجودیت‌های نام‌دار<sup>۲۰</sup> پیاده‌سازی شده در آزمایشگاه پردازش زبان طبیعی امیرکبیر است که در پکیج `NER_module` موجودند. در این سیستم خروجی هر دو روش گرفته شده و از آنهایی که گره‌ی مرتبط در گراف فارس‌بیس دارند استفاده می‌شود. روندهای مربوط به این قسمت در فایل `NER_provider` قابل مشاهده می‌باشند. در ادامه خروجی این دو روش را برای جمله‌ی "پایتخت کشور آلمان کجاست" مشاهده می‌کنید. خروجی اول مربوط به رابط برنامه‌نویسی فارس‌بیس و خروجی دوم مربوط به کتابخانه‌ی تشخیص‌دهنده می‌باشد:

['Beginning :Thing', 'Beginning :Country', 'Inside :Country', 'Outside']

['o', 'o', 'b-loc', 'o'].

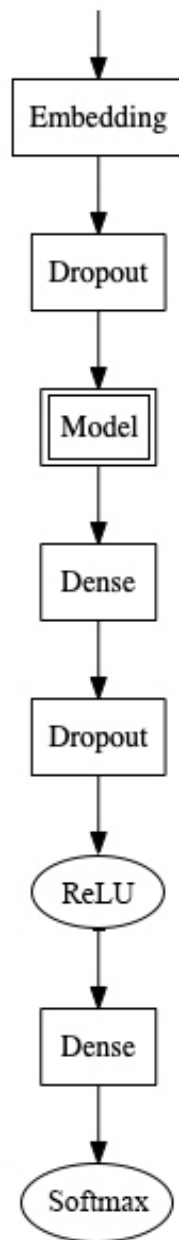
## ۴-۵ تولید کوئری

رابطه و موجودیت تشخیص داده شده توسط بخش‌های دیگر سیستم که توضیح داده شدند، در این بخش استفاده شده و کوئری مناسب تولید می‌شود. از آنجا که پایگاه دانش از یک گراف جهت‌دار ساخته شده

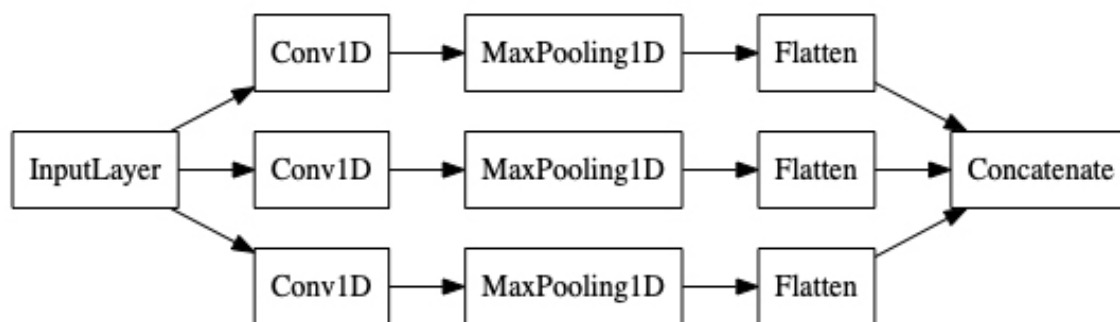
<sup>۱۸</sup>Max Pooling

<sup>۱۹</sup>API

<sup>۲۰</sup>Named-Entity Recognition



شکل ۴-۲: لایه‌های دسته‌بند شبکه عصبی پیچشی



شکل ۳-۴: لایه‌های هسته‌ی دسته‌بند شبکه عصبی پیچشی

است، موجودیت سوال می‌تواند در ابتدا و یا انتهای رابطه باشد. بر همین اساس کوئری ارسالی به دو صورت می‌تواند ساخته شود که هر دو قالب را در ادامه مشاهده می‌کنید:

<pre> select distinct(?o) where { ?o &lt;relation_uri&gt; &lt;entity_uri&gt;. }         </pre>	<pre> select distinct(?o) where { &lt;entity_uri&gt; &lt;relation_uri&gt; ?o. }         </pre>
------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------

به عنوان مثال برای پرسش "پایتخت آلمان کجاست" موجودیت "آلمان" و رابطه‌ی "پایتخت یک کشور" بدست آورده شده‌اند. با استفاده از این اطلاعات و بدست آوردن جهت رابطه در این سوال کوئری زیر تولید می‌شود:

```

select distinct(?o) where {
<http://fkg.iust.ac.ir/resource/آلمان> <http://fkg.iust.ac.ir/ontology/capital> ?o.
}
    
```

اگر برای پرسش ورودی چندین رابطه با احتمال بالا و یا چندین موجودیت تشخیص داده شود، از ترکیب آنها چندین کوئری ساخته شده و در مرحله‌ی ارسال از همه‌ی آنها استفاده می‌شود و چندین پاسخ برای کاربر بازگردانده می‌شود.

## فصل پنجم

### تحلیل، طراحی و پیاده‌سازی سیستم

در این فصل به توضیحات مربوط به تحلیل، طراحی و پیاده‌سازی سیستم پرداخته می‌شود. ابتدا متدولوژی مورد استفاده در این سیستم توضیح داده شده و سپس نیازمندی‌های سیستم بررسی می‌شوند. در ادامه با معماری این سیستم و بخش‌های مختلف آن آشنا شده و چگونگی پیاده‌سازی هر یک از بخش‌های سیستم جداگانه توضیح داده می‌شود.

## ۱-۵ الگوی سیستم

یک متدولوژی مجموعه‌ای از روش‌ها، توصیه‌ها و قالب‌ها است که به همراه راهبرد مشخص و طی مراحل مختلف، در توسعه‌ی سیستم به کار گرفته می‌شود. براساس رویکردهای مختلف متدولوژی‌های مختلفی وجود دارد. به عنوان مثال در رویکرد توسعه چابک<sup>۱</sup>، متدولوژی‌های اسکرام<sup>۲</sup> و کانبان<sup>۳</sup> وجود دارند. یا در رویکرد شیء‌گرا<sup>۴</sup> متدولوژی RUP<sup>۵</sup> تعریف شده است. انتخاب متدولوژی وابسته به عوامل بسیاری چون نیازمندی‌های نرم‌افزار، تعداد اعضای تیم، زمان‌بندی پروژه و ... می‌باشد. از آنجا که نیازمندی‌های سیستم به طور دقیق در ابتدا مشخص نبوده‌اند از الگوی چابک<sup>۶</sup> استفاده شده است. این الگو در واقع همان الگوی آبشاری<sup>۷</sup> است که در چندین تناوب تکرار می‌شود و مراحل پیاده‌سازی و آزمایش همزمان با مراحل تحلیل و طراحی سیستم مورد نظر انجام می‌گیرد. در این الگو هر آنچه تولید می‌شود بلافاصله مورد آزمایش قرار می‌گیرد و سپس با تحلیل دوباره‌ی نیازمندی‌ها مراحل تکرار می‌شوند [۱۰]. در الگوی چابک استفاده شده در این پروژه، مراحل زیر به ترتیب و به صورت تکرارشونده انجام شده‌اند:

۱. تحلیل نیازمندی‌ها

۲. برنامه‌ریزی

۳. طراحی سیستم

۴. پیاده‌سازی

۵. آزمایش

Agile<sup>۱</sup>

Scrum<sup>۲</sup>

Kanban<sup>۳</sup>

Object Oriented<sup>۴</sup>

Rational Unified Process<sup>۵</sup>

Agile<sup>۶</sup>

Waterfall Model<sup>۷</sup>

در ادامه توضیحات مربوط به تحلیل، طراحی و پیاده‌سازی نرم‌افزار آورده شده است. توضیحات مربوط به آزمایش و اعتبارسنجی قسمت‌های مختلف سیستم در فصل‌های مربوط به خودشان توضیح داده خواهد شد.

## ۵-۲ تحلیل نیازمندی‌های سیستم

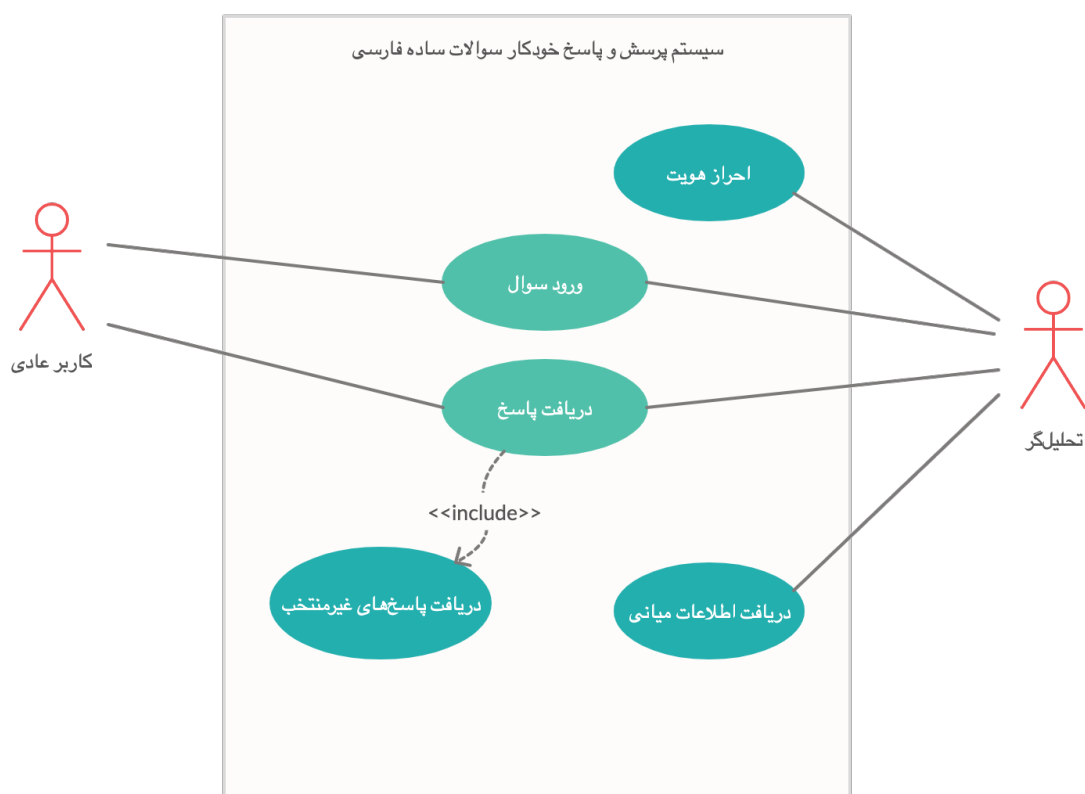
این سیستم دو دسته استفاده کننده دارد. دسته‌ی اول کاربر عادی سیستم است که نیازمندی‌های آن به شرح زیر است:

- ورود سوال: کاربر عادی باید بتواند یک جمله را به عنوان پرسش مورد نظرش در سیستم ثبت کند تا سیستم آنرا بررسی کند.
- دریافت پاسخ: کاربر عادی باید بتواند پاسخی که سیستم بدست آورده را مشاهده نماید.
- دریافت پاسخ‌های غیرمنتخب: این کاربر باید بتواند پاسخ‌های غیرمنتخب تشخیص داده شده توسط سیستم را مشاهده کند. این نیازمندی زیرمجموعه‌ی نیازمندی دریافت پاسخ می‌باشد.
- دسته‌ی دوم کاربر تحلیل‌گر است. تحلیل‌گر تمامی نیازمندی‌های ذکر شده برای کاربر عادی را شامل شده و نیازمندی‌های زیر را نیز دارد:
- احراز هویت: تحلیل‌گر باید بتواند خود را به سیستم بشناساند و تحت عنوان تحلیل‌گر به سیستم ورود یا خروج کند.
- دریافت اطلاعات میانی: تحلیل‌گر می‌تواند اطلاعات میانی سیستم که شامل کوئری‌های ارسال شده توسط سیستم است را مشاهده نماید.

نیازمندی‌های کاربردی سیستم توسط نمودار Use case نمایش داده می‌شود. در شکل ۵-۱ این نمودار را مشاهده می‌کنید. همانطور که گفته شد این سیستم دارای دو نقش کاربر عادی و تحلیل‌گر می‌باشد.

## ۵-۳ طراحی سیستم

در شکل ۵-۲ معماری این سیستم را مشاهده می‌کنید که از الگوی سرویس‌دهنده-سرویس‌گیرنده برای طراحی آن استفاده شده است. این سیستم بوسیله‌ی یک رابط کاربری تحت وب برای کاربر ارائه



شکل ۵-۱: نمودار Use case

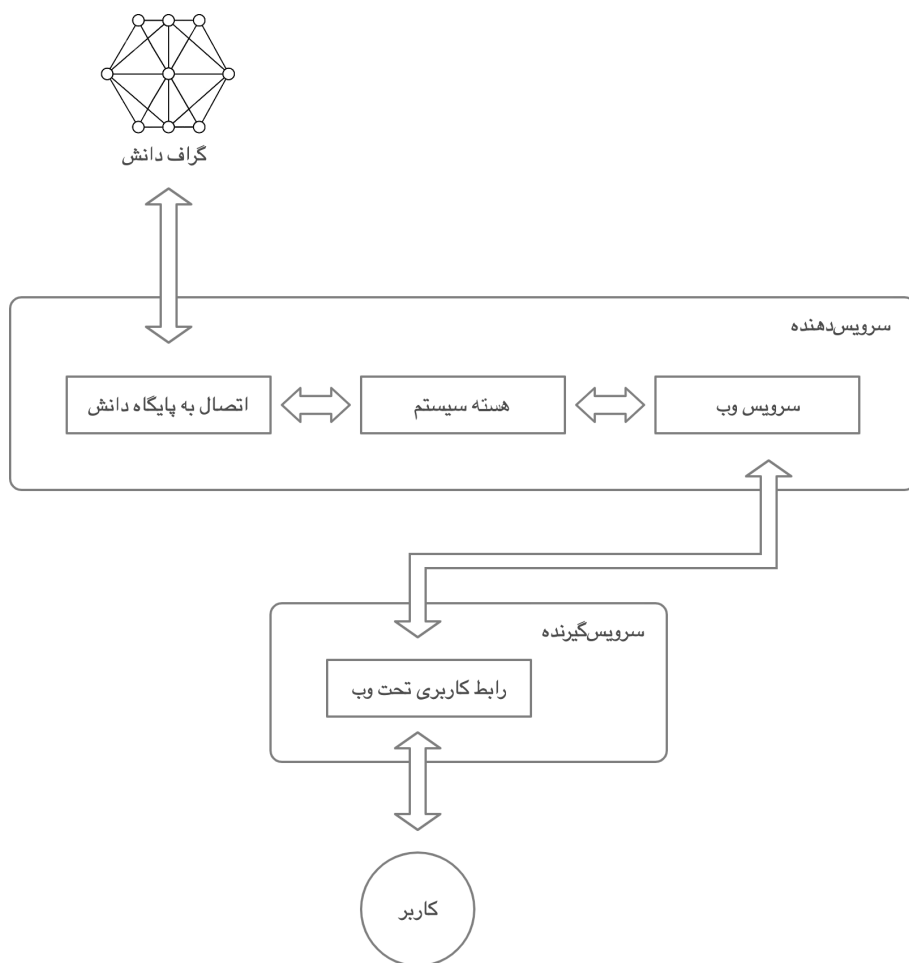
می‌شود. بنابراین بخش‌های اصلی سیستم شامل بخش سرویس‌دهنده و سرویس‌گیرنده می‌باشد. بخش سرویس‌دهنده دارای سه بخش اصلی هسته سیستم، سرویس وب و اتصال به پایگاه دانش می‌باشد. هسته سیستم بخش اصلی سیستم است و شامل مدل‌های یادگیری ماشین برای تشخیص رابطه و موجودیت‌های داخل سوال بوده و وظیفه‌ی تولید کوئری مرتبط را نیز دارد. بخش اتصال به پایگاه داده وظیفه‌ی ارسال کوئری و گرفتن پاسخ از گراف دانش را انجام می‌دهد. کاربر بوسیله‌ی رابط کاربری تحت وب با سرویس‌گیرنده و سرویس‌گیرنده بوسیله‌ی سرویس وب با سرویس‌دهنده در ارتباط است.

## ۴-۵ پیاده‌سازی سیستم

بخش عمده‌ی این سیستم بوسیله‌ی زبان پایتون<sup>۸</sup> نسخه‌ی ۷.۶.۳ پیاده‌سازی شده است. در این بخش هر کدام از قسمت‌های سیستم که در بخش طراحی ذکر شده‌اند توضیح داده خواهند شد.

<sup>۸</sup>Python





شکل ۵-۲: معماری سیستم

## ۱-۴-۵ هسته سیستم

هسته سیستم بخش اصلی این سیستم است که وظیفه‌ی آن پاسخ‌دهی به سه زیرمسئله‌ی اصلی سیستم شامل تشخیص رابطه، تشخیص موجودیت و تولید کوئری می‌باشد. این بخش همان سیستم پیشنهادی توضیح داده شده در فصل پیشین است. فایل اصلی این قسمت `answer_generator.py` است که وظیفه‌ی آن آماده‌سازی روند پاسخ‌دهنده‌ی سوالات کاربران می‌باشد. در این فایل روندهای مورد نیاز برای سه زیرمسئله‌ی بیان شده فراخوانی شده و استفاده می‌شوند.

## ۲-۴-۵ اتصال به پایگاه دانش

در فصل قبل چگونگی تولید کوئری مورد نیاز برای ارسال به پایگاه دانش فارس‌بیس شرح داده شد. بخش اتصال به پایگاه دانش وظیفه دارد که کوئری‌های تولید شده توسط سیستم را به پایگاه دانش

ارسال کرده و پاسخ مرتبط را از پایگاه دریافت نموده و به سیستم بازگرداند. پایگاه دانش فارس‌بیس چنین درخواست‌هایی را از طریق آدرس `http://farsbase.net:8890/sparql` پاسخ می‌دهد. این درخواست‌ها با استفاده از متد `Post` در پروتکل `HTTP` ارسال می‌شوند. پاسخ دریافتی از پایگاه دانش که به صورت `XML` است می‌تواند در برگیرنده‌ی چندین موجودیت به عنوان پاسخ باشد. این موجودیت‌ها می‌توانند دارای گره در پایگاه دانش باشند که در این صورت آدرس آنها در پایگاه بازگردانده می‌شود. روندهای مربوط به این قسمت در فایل `graph_extractor.py` پیاده‌سازی شده‌اند.

### ۵-۴-۳ سرویس و رابط کاربری تحت وب

تمامی فایل‌های مربوط به این دو بخش از سیستم در پکیج `web_interface` قرار دارند.

#### سرویس

برای اتصال هسته سیستم به رابط کاربری وب از کتابخانه‌ی `Flask` در پایتون استفاده شده است. بوسیله‌ی این کتابخانه رابط‌های برنامه‌نویسی کاربردی مورد نیاز فراهم شده و بوسیله‌ی آنها روندهای متناسب در هسته سیستم فراخوانی می‌شوند.

دو مسیر<sup>۹</sup> در این سرویس ارائه می‌شود. مسیر اول به آدرس `"/"` صفحه‌ی اصلی رابط کاربری را برای کاربر نمایش می‌دهد که کاربر می‌تواند در این صفحه پرسش مورد نظر خود را وارد کرده و پاسخ را مشاهده کند. مسیر دوم به آدرس `"/answer"` می‌تواند بوسیله‌ی متد `Post`<sup>۱۰</sup> در پروتکل `HTTP` فراخوانی شود. این مسیر پرسش وارد شده توسط کاربر را همراه با پارامترهای دیگر دریافت کرده و با فراخوانی روند مورد نیاز در هسته سیستم پاسخ را در فرمت `JSON` برمی‌گرداند. این سرویس در فایل `interface.py` پیاده‌سازی شده است.

#### رابط کاربری

در رابط کاربری کاربران می‌توانند پرسش خود را وارد کنند و رابطه و موجودیت تشخیص داده شده برای آن و پاسخ خود را دریافت نمایند. رابط کاربری تحت وب با فراخوانی رابط‌های برنامه‌نویسی کاربردی پیاده‌سازی شده در سرویس وب با سرویس‌دهنده ارتباط برقرار می‌کند. برای کاربران این گزینه وجود دارد که اگر بخواهند سیستم روابط غیر منتخب را نیز برایشان بررسی کند که در این صورت احتمال

<sup>۹</sup> Route  
<sup>۱۰</sup> Method

درستی هر کدام از روابط تشخیص داده شده نیز نمایش داده می‌شود. کاربر تحلیل‌گر می‌تواند اطلاعات میانی سیستم که شامل کوئری ارسالی می‌شود را نیز مشاهده کند. تصاویری از رابط کاربری را همراه با پرسش‌ها و حالات مختلف مشاهده می‌کنید.

The screenshot shows a web application interface with a red header bar. On the left, there is a dark grey button labeled 'ورود ادمین'. In the center, there is a toggle switch labeled 'بررسی روابط غیرمنتخب' which is currently turned off. On the right, there is a white search bar with the text 'ترکیبات غذای قرمه سبزی چیست؟'. Below the header, there is a blue box containing the text 'رابطه‌ی تشخیص داده شده " ترکیبات یک غذا " برای موجودیت " قرمه سبزی "'. Below this, there is a white box containing the text 'تره، جعفری، شنبلیله، لوبیا، گوشت قرمز، لیمو امانی'.

شکل ۵-۳: صفحه کاربر عادی

ورود ادمین

زبان و پایتخت کشور ایران چیست ؟

☒ بررسی روابط غیرمنتخب

احتمال ۶۴٪

رابطه‌ی تشخیص داده شده " پایتخت یک کشور "

برای موجودیت " ایران "

تهران

احتمال ۳۱٪

رابطه‌ی تشخیص داده شده " زبان رسمی یک کشور "

برای موجودیت " ایران "

زبان فارسی

شکل ۴-۵: صفحه کاربر عادی همراه روابط غیرمنتخب

خروج

☒ نمایش کوئری ارسال شده

☐ بررسی روابط غیرمنتخب

آل پاچینو در چه فیلم‌هایی بازی کرده است ؟

رابطه‌ی تشخیص داده شده " بازیگران یک فیلم "

برای موجودیت " آل پاچینو "

راه کارلینو

بوی خوش زن (فیلم ۱۹۹۲)

مترسک (فیلم ۱۹۷۳)

دانش براسکو

جک و جیل

ناتالی و من

نیو عدالت برای همه (فیلم)

کوئری ارسال شده

```

select distinct(?o) where { ?o
<http://4kg.iust.ac.ir/ontology/starring>
<http://4kg.iust.ac.ir/resource/آل_پاچینو> }

```

شکل ۵-۵: صفحه کاربر تحلیل‌گر

خروج

☒ نمایش کوثری ارسال شده
☐ بررسی روابط غیرمنتخب

آل پاچینو و رابرت دنیرو در چه فیلم‌هایی بازی کرده‌اند؟

رابطه‌ی تشخیص داده شده " بازیگران یک فیلم " برای موجودیت " رابرت دنیرو "

امتیاز (فیلم)  
نامحدود (فیلم)  
راننده تاکسی  
زندگی این پسر  
غبار ستاره (فیلم ۲۰۰۷)  
ماشته  
مردان افتخار

کوثری ارسال شده  

```
select distinct(?o) where { ?o <http://fkg.iust.ac.ir/ontology/starring> <http://fkg.iust.ac.ir/resource/رابرت_دنیرو> ?o. }
```

رابطه‌ی تشخیص داده شده " بازیگران یک فیلم " برای موجودیت " آل پاچینو "

راه کارلیتو  
بوی خوش زن (فیلم ۱۹۹۲)  
مترسک (فیلم ۱۹۷۳)  
دانش براسکو  
جک و جیل  
ناتالی و من  
...و عدالت برای همه (فیلم)

کوثری ارسال شده  

```
select distinct(?o) where { ?o <http://fkg.iust.ac.ir/ontology/starring> <http://fkg.iust.ac.ir/resource/آل_پاچینو> ?o. }
```

شکل ۵-۶: صفحه کاربر تحلیل گر همراه چند موجودیت

خروج

☒ نمایش کوثری ارسال شده
☒ بررسی روابط غیرمنتخب

پایتخت کشور هلند و ایران چه شهرهایی است و به چه زبانی صحبت می‌کنند؟

رابطه‌ی تشخیص داده شده " زبان رسمی یک کشور " برای موجودیت " ایران "

زبان فارسی

کوثری ارسال شده  

```
select distinct(?o) where { <http://fkg.iust.ac.ir/resource/ایران> <http://fkg.iust.ac.ir/ontology/language> ?o. }
```

رابطه‌ی تشخیص داده شده " زبان رسمی یک کشور " برای موجودیت " هلند "

زبان هلندی

کوثری ارسال شده  

```
select distinct(?o) where { <http://fkg.iust.ac.ir/resource/هلند> <http://fkg.iust.ac.ir/ontology/language> ?o. }
```

رابطه‌ی تشخیص داده شده " پایتخت یک کشور " برای موجودیت " ایران "

تهران

کوثری ارسال شده  

```
select distinct(?o) where { <http://fkg.iust.ac.ir/resource/ایران> <http://fkg.iust.ac.ir/ontology/capital> ?o. }
```

رابطه‌ی تشخیص داده شده " پایتخت یک کشور " برای موجودیت " هلند "

آمستردام

کوثری ارسال شده  

```
select distinct(?o) where { <http://fkg.iust.ac.ir/resource/هلند> <http://fkg.iust.ac.ir/ontology/capital> ?o. }
```

شکل ۵-۷: صفحه کاربر تحلیل گر همراه چند موجودیت و روابط غیرمنتخب

## فصل ششم

# آزمایش و ارزیابی سیستم

در فصل قبل قسمت‌های مختلف سیستم همراه با مدل‌های مورد استفاده بررسی شدند. در این فصل ابتدا با معیارهای معمول ارزیابی آشنا شده، سپس بخش‌های مختلف موجود در سیستم مورد آزمایش قرار گرفته و ارزیابی می‌شوند.

## ۱-۶ معیارهای ارزیابی

برای ارزیابی یک دسته‌بند دودویی امکان رخداد چهار حالت شامل منفی درست<sup>۱</sup>، منفی نادرست<sup>۲</sup>، مثبت درست<sup>۳</sup> و مثبت نادرست<sup>۴</sup> وجود دارد که دو حالت از آنها خطا محسوب می‌شوند. این دو حالت حالتی هستند که خروجی تشخیص داده شده با خروجی واقعی در تضاد باشد و در علم آمار به خطای نوع اول و دوم مشهور می‌باشند. این حالات را در جدول ۱-۶ با نام ماتریس درهم‌ریختگی<sup>۵</sup> مشاهده می‌کنید. در ادامه برای اختصار از نمادهای  $TP$ ،  $FP$ ،  $TN$  و  $FN$  به ترتیب به نمایندگی از تعداد موارد مثبت درست، مثبت نادرست (خطای نوع اول)، منفی درست و منفی نادرست (خطای نوع دوم) استفاده می‌کنیم.

واقعاً مثبت	واقعاً منفی	
منفی نادرست (خطای نوع دوم)	منفی درست	پیش‌بینی منفی
مثبت درست	مثبت نادرست (خطای نوع اول)	پیش‌بینی مثبت

جدول ۱-۶: ماتریس درهم‌ریختگی

### ۱-۱-۶ صحت

یکی از ساده‌ترین معیارها برای ارزیابی درستی یک سیستم استفاده از معیار صحت<sup>۶</sup> می‌باشد. درصد صحت سیستم با نسبت گرفتن تعداد پاسخ‌های درست بر تعداد کل پاسخ‌ها بدست می‌آید که رابطه‌ی

- 
- True Negative (TN)<sup>۱</sup>
  - False Negative (FN)<sup>۲</sup>
  - True Positive (TP)<sup>۳</sup>
  - False Positive (FP)<sup>۴</sup>
  - Confusion Matrix<sup>۵</sup>
  - Accuracy<sup>۶</sup>

آنرا در ۱-۶ مشاهده می‌کنید.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (۱-۶)$$

## ۲-۱-۶ دقت و فراخوانی و امتیاز $F_1$

دقت<sup>۷</sup> و فراخوانی<sup>۸</sup> دو معیار ارزیابی هستند که بر اساس ماتریس درهم‌ریختگی بدست می‌آیند. دقت به این سوال پاسخ می‌دهد که چه نسبتی از موارد انتخابی، مرتبط هستند. مقدار دقت برابر است با نسبت تعداد مواردی که سیستم به درستی تشخیص داده بر تعداد کل مواردی که سیستم تشخیص داده است.

فراخوانی به این سوال پاسخ می‌دهد که چه نسبتی از موارد مرتبط، انتخاب شده‌اند. مقدار فراخوانی برابر است با نسبت تعداد مواردی که سیستم به درستی تشخیص داده بر تعداد کل مواردی که سیستم باید تشخیص می‌داده است.

روابط دقت و فراخوانی را به ترتیب در ۲-۶ و ۳-۶ مشاهده می‌کنید.

$$Recall = \frac{TP}{TP + FN} \quad (۳-۶) \quad Precision = \frac{TP}{TP + FP} \quad (۲-۶)$$

در بسیاری از موارد نمی‌توان با استناد به معیار صحت درستی یک سیستم را بررسی کرد. به عنوان مثال سیستمی را در نظر بگیرید که هدف آن تشخیص یک بیماری خاص می‌باشد. از آنجا که درصد بسیار کمی از افراد دارای این بیماری هستند اگر سیستم به ازای همه‌ی ورودی‌ها عدم بیماری را تشخیص دهد، درصد صحت بسیار بالایی خواهد داشت در صورتی که چنین سیستمی اصلاً کارایی ندارد. در چنین سیستم‌هایی معیارهای دقت و فراخوانی کاربردی هستند.

معیار  $F_1$ <sup>۹</sup> نوعی میانگین بین معیارهای دقت و فراخوانی است. برای مقایسه‌ی دو سیستم با معیار دقت و فراخوانی می‌توان در یک نگاه معیار  $F_1$  دو سیستم را مقایسه کرد. رابطه‌ی این معیار را در ۴-۶ مشاهده

Precision<sup>۷</sup>

Recall<sup>۸</sup>

F1-Score<sup>۹</sup>



می‌نمایید.

$$F_1 = \frac{2 * Precision * Recall}{Precision + Recall} \quad (۴-۶)$$

### دقت و فراخوانی و امتیاز $F_1$ برای دسته‌بندی با چند کلاس

برای مسائلی با چند کلاس می‌توان معیارهای دقت و فراخوانی را برای هر کلاس به صورت جداگانه محاسبه نمود. در این مسائل مقدار دقت یک کلاس برابر است با نسبت تعداد مواردی که سیستم به درستی آن کلاس را تشخیص داده بر تعداد کل مواردی که سیستم برای آن کلاس تشخیص داده است. مقدار فراخوانی برابر است با نسبت تعداد مواردی که سیستم به درستی برای آن کلاس تشخیص داده بر تعداد کل مواردی که عضو در این کلاس می‌باشد.

برای بدست آورد یک مقدار واحد برای دقت یا فراخوانی می‌توان میانگین میکرو<sup>۱۰</sup> یا میانگین ماکرو<sup>۱۱</sup> این معیارها را محاسبه نمود. برای بدست آوردن میانگین ماکرو ابتدا مقدار دقت و یا فراخوانی هر کلاس جداگانه حساب شده و میانگین آنها محاسبه می‌شود. اما برای محاسبه‌ی میانگین میکرو تعداد موارد مثبت یا منفی درست یا نادرست برای هر کلاس محاسبه شده و از مجموع همگی آنها طبق فرمول دقت و فراخوانی استفاده می‌شود. در ادامه روابط این دو میانگین برای معیارهای دقت و فراخوانی را برای یک دسته‌بند با  $n$  کلاس مشاهده می‌کنید. در این روابط برای تفکیک تعداد موارد هر حالت از پاسخ‌های هر کلاس از اندیس استفاده شده است. به عنوان مثال منظور از  $TP_i$  تعداد موارد مثبت درست برای کلاس  $i$ ام می‌باشد.

$$\left( \sum_{i=1}^n \frac{TP_i}{TP_i + FN_i} \right) / n \quad (۶-۶)$$

میانگین ماکرو معیار فراخوانی

$$\left( \sum_{i=1}^n \frac{TP_i}{TP_i + FP_i} \right) / n \quad (۵-۶)$$

میانگین ماکرو معیار دقت

---

Micro Average<sup>۱۰</sup>

Macro Average<sup>۱۱</sup>

$$\frac{\sum_{i=1}^n TP_i}{\sum_{i=1}^n (TP_i + FN_i)} \quad (۸-۶)$$

میانگین میکرو معیار فراخوانی

$$\frac{\sum_{i=1}^n TP_i}{\sum_{i=1}^n (TP_i + FP_i)} \quad (۷-۶)$$

میانگین میکرو معیار دقت

در بررسی چنین مسائلی با چند کلاس باز هم می‌توان از مقدار بدست آمده از میانگین دقت و فراخوانی استفاده کرد و طبق فرمول گفته شده امتیاز  $F_1$  را برای آن مسئله محاسبه نمود. در ارزیابی دسته‌بندهای این پروژه از میانگین ماکرو استفاده می‌شود.

## ۲-۶ روش اعتبارسنجی متقابل

اعتبارسنجی متقابل<sup>۱۲</sup> یک روش ارزیابی مدل است که تعیین می‌نماید نتایج یک تحلیل آماری بر روی یک مجموعه داده تا چه اندازه قابل تعمیم و مستقل از داده‌های آموزشی است و چه مقدار می‌تواند در مقابل داده‌های جدید به درستی عمل کند. این روش به طور ویژه در کاربردهای پیش‌بینی مورد استفاده قرار می‌گیرد تا مشخص شود مدل مورد نظر تا چه اندازه در عمل مفید خواهد بود. به طور کلی یک دور از اعتبارسنجی متقابل شامل افراز داده‌ها به دو زیرمجموعه مکمل تحت عنوان مجموعه‌ی آموزش و مجموعه‌ی اعتبارسنجی یا آزمایش، انجام آموزش روی زیرمجموعه‌ی آموزش و سپس ارزیابی توسط مجموعه‌ی دیگر است. برای کاهش پراکندگی<sup>۱۳</sup> در نتایج، عمل اعتبارسنجی چندین بار با افرازهای مختلف انجام شده و از نتایج اعتبارسنجی‌ها میانگین گرفته می‌شود. با افزایش تعداد افرازها پراکندگی نتایج کاهش میابد [۳].

### ۱-۲-۶ اعتبارسنجی متقابل چندتایی

در روش اعتبارسنجی متقابل چندتایی یا  $k$ -تایی<sup>۱۴</sup> داده‌های اصلی به صورت رندم به  $k$  زیرمجموعه با اندازه‌ی مساوی تقسیم می‌شود. فرایند آزمایش در  $k$  مرحله انجام می‌شود. در مرحله‌ی  $i$ ام زیرمجموعه‌ی  $i$ ام به عنوان مجموعه‌ی آزمایش و سایر داده‌ها به عنوان مجموعه‌ی آموزش در نظر گرفته شده و آموزش

Cross Validation<sup>۱۲</sup>Variability<sup>۱۳</sup>K-Fold<sup>۱۴</sup>

مدل و سپس ارزیابی آن انجام می‌شود. در نهایت میانگین نتایج این آزمایش‌ها محاسبه شده و به عنوان معیار ارزیابی مدل استفاده می‌شود. مزیت این روش در این است که هر یک از داده‌ها به طور برابر برای آموزش و آزمایش استفاده می‌شوند.

## ۳-۶ نتایج ارزیابی سیستم

سیستم پیاده‌سازی شده در این پروژه دارای دو قسمت اصلی دسته‌بند رابطه و تشخیص‌دهنده‌ی موجودیت می‌باشد. در این بخش هر کدام از این بخش‌ها و همچنین کل سیستم به صورت یکپارچه مورد آزمایش قرار گرفته شده و داده‌هایی که منجر به نتیجه‌ی غلط می‌شدند، استخراج شده و مورد تحلیل قرار گرفته‌اند. برنامه‌های مربوط به ارزیابی بخش‌های مختلف سیستم در پکیج evaluation پیاده‌سازی شده‌اند.

### ۱-۳-۶ ارزیابی دسته‌بندهای رابطه

همانطور که در فصل‌های قبل توضیح داده شد در این سیستم از دو دسته‌بند ماشین بردار پشتیبان و شبکه عصبی پیچشی استفاده می‌شود. نتایج آزمایش هر دو دسته‌بند را در ادامه مشاهده می‌کنید. هر کدام از دسته‌بندها همراه روش تعبیه‌سازی خود در پنج روش مورد آزمایش قرار گرفته‌اند. روش اول روش اعتبارسنجی متقابل چندتایی با تعداد ۵ زیرمجموعه است. برای چهار روش بعدی مجموعه داده‌های اصلی با نسبت ۴، ۱، ۱ به سه قسمت آموزش، اعتبارسنجی و آزمایش تقسیم شده‌اند. در اولین این روش‌ها مجموعه داده‌ها به صورت کاملاً رندم تقسیم می‌شوند. در روش بعدی مجموعه داده‌ها طوری تقسیم می‌شوند که الگوی داده‌های مجموعه‌ی آموزش در دو مجموعه‌ی دیگر تکرار نشود. دو روش بعدی به ترتیب همانند روش‌های ذکر شده هستند اما با این تفاوت که روابط یک جهت‌یافته‌اند. به عنوان مثال در مجموعه‌ی عادی رابطه‌ی "پایتخت یک کشور" وجود دارد اما در روابط یک‌جهته به ازای آن، دو رابطه‌ی "پایتخت یک کشور ۱" و "پایتخت یک کشور ۲" وجود دارد که تفاوت آنها در این است که یکی از آنها رابطه از موجودیت کشور به شهر است و دیگری برعکس. در چنین حالتی تشخیص برای دسته‌بند سخت‌تر می‌شود. این پنج روش را به طور خلاصه در ادامه مشاهده می‌نمایید:

• ۵تایی<sup>۱۵</sup>

• مجموعه داده

<sup>۱۵</sup>5-Fold

• مجموعه داده - الگو متمایز

• مجموعه داده - یک جهته

• مجموعه داده - یک جهته - الگو متمایز

### ماشین بردار پشتیبان

در جدول ۲-۶ نتایج ارزیابی دسته‌بند ماشین بردار پشتیبان را مشاهده می‌کنید.

امتیاز $F_1$	فراخوانی (میانگین ماکرو)	دقت (میانگین ماکرو)	صحت	روش تعبیه‌سازی	روش آزمایش
۹۹.۱۲	۹۸.۸۸	۹۹.۳۵	۹۹.۴۹	BOW	۵ تایی
۹۹.۳۱	۹۹.۱۵	۹۹.۴۷	۹۹.۶۲	TF-IDF	
۹۹.۳۳	۹۹.۳۱	۹۹.۳۵	۹۹.۶۳	word2vec	
۹۹.۲۲	۹۹.۱۰	۹۹.۳۴	۹۹.۵۷	word2vec وزن‌دار	
۹۸.۹۵	۹۸.۵۸	۹۹.۳۲	۹۹.۳۶	BOW	مجموعه داده
۹۹.۱۰	۹۸.۸۷	۹۹.۳۴	۹۹.۴۸	TF-IDF	
۹۹.۰۲	۹۸.۸۹	۹۹.۱۶	۹۹.۴۶	word2vec	
۹۹.۱۲	۹۸.۸۹	۹۹.۳۶	۹۹.۴۹	word2vec وزن‌دار	
۹۷.۹۶	۹۷.۵۸	۹۸.۳۳	۹۸.۷۶	BOW	مجموعه داده الگو متمایز
۹۷.۱۹	۹۷.۰۰	۹۷.۳۸	۹۸.۶۳	TF-IDF	
۹۷.۳۶	۹۷.۱۹	۹۷.۵۴	۹۸.۵۷	word2vec	
۹۶.۲۰	۹۵.۸۸	۹۶.۵۲	۹۸.۱۰	word2vec وزن‌دار	
۹۷.۰۴	۹۶.۱۷	۹۷.۹۳	۹۸.۲۷	BOW	مجموعه داده یک جهته
۹۸.۲۹	۹۸.۰۵	۹۸.۵۲	۹۹.۰۸	TF-IDF	
۹۸.۲۳	۹۸.۱۰	۹۸.۳۷	۹۸.۹۵	word2vec	
۹۷.۷۸	۹۷.۴۹	۹۸.۰۷	۹۸.۶۶	word2vec وزن‌دار	
۹۴.۳۱	۹۳.۴۷	۹۵.۱۷	۹۶.۶۷	BOW	مجموعه داده یک جهته الگو متمایز
۹۴.۲۸	۹۴.۰۱	۹۴.۵۵	۹۷.۰۶	TF-IDF	
۹۲.۷۶	۹۲.۸۶	۹۲.۶۷	۹۵.۹۶	word2vec	
۹۰.۹۲	۹۰.۶۵	۹۱.۱۹	۹۵.۳۸	word2vec وزن‌دار	

جدول ۲-۶: ارزیابی دسته‌بند ماشین بردار پشتیبان

در جدول ۳-۶ نمونه‌هایی از داده‌های مشکل‌دار مربوط به این دسته‌بند را مشاهده می‌کنید. عمده

مشکل این دسته‌بند در تشخیص داده‌های کلاس‌هایی بوده که حاوی کلمات مشترک زیادی در جملات آموزششان بوده‌اند.

پرسش	رابطه‌ی هدف	رابطه‌ی تشخیص داده شده
سوتن کورو نویسنده اش چه نام دارد	نویسنده یک فیلم	نویسنده کتاب
چه کسی فیلم جزیره مرده ۲ را کارگردانی می کند	کارگردان یک سریال تلویزیونی	کارگردان یک فیلم

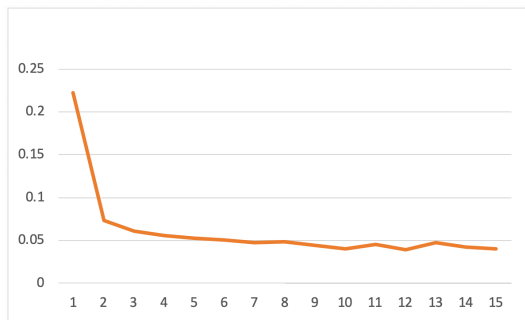
جدول ۳-۶: نمونه‌های مشکل‌دار دسته‌بند ماشین بردار پشتیبان

### شبکه عصبی پیچشی

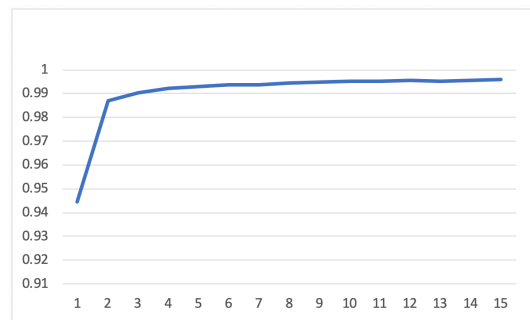
برای آموزش نهایی این شبکه فرایند آموزش روی داده‌ها در ۱۰ دوره<sup>۱۶</sup> انجام شده است. برای تحلیل عملکرد این شبکه مقدار صحت و تابع هزینه‌ی این شبکه طی دوره‌های ۱ تا ۱۵ استخراج شده‌اند که در جدول ۴-۶ قرار گرفته‌اند. همانطور که در فصل گذشته گفته شده بود از تابع هزینه‌ی Categorical Cross-Entropy برای آموزش این شبکه می‌شود. نمودار صحت و هزینه را در شکل‌های ۱-۶ و ۲-۶ مشاهده می‌کنید.

دوره	صحت	هزینه			
۱	۹۴.۴۵	۰.۲۲۲۴	۹	۹۹.۴۸	۰.۰۴۴۲
۲	۹۸.۷۱	۰.۰۷۳۴	۱۰	۹۹.۵۲	۰.۰۴۰۵
۳	۹۹.۰۵	۰.۰۶۰۷	۱۱	۹۹.۵۲	۰.۰۴۵۵
۴	۹۹.۲۲	۰.۰۵۵۳	۱۲	۹۹.۵۵	۰.۰۳۹۴
۵	۹۹.۳۱	۰.۰۵۲۵	۱۳	۹۹.۵۳	۰.۰۴۷۰
۶	۹۹.۳۵	۰.۰۵۰۱	۱۴	۹۹.۵۷	۰.۰۴۲۴
۷	۹۹.۳۸	۰.۰۴۷۷	۱۵	۹۹.۶۱	۰.۰۴۰۱

جدول ۴-۶: صحت و هزینه‌ی شبکه عصبی پیچشی



شکل ۶-۲: هزینه‌ی شبکه دسته‌بند



شکل ۶-۱: صحت شبکه دسته‌بند

در جدول ۵-۶ نتایج ارزیابی دسته‌بند شبکه عصبی پیچشی را مشاهده می‌کنید. از آنجا که روش تعبیه‌سازی استفاده شده در این دسته‌بند یعنی روش word2vec، یکی از روش‌های تعبیه‌سازی مورد استفاده در دسته‌بند ماشین بردار پشتیبان نیز می‌باشد، برای مقایسه، نتایج آزمایش مربوط به این روش تعبیه‌سازی در دسته‌بند ماشین بردار پشتیبان که در جدول ۲-۶ بیان شدند در جدول ۵-۶ نیز آورده شده‌اند. با مقایسه‌ی نتایج این دو دسته‌بند مشخص می‌شود که در اکثر آزمایش‌ها و روش‌ها شبکه عصبی پیچشی عملکرد نسبتاً بهتری را داشته است.

نمونه‌هایی از داده‌های مشکل دار دسته‌بند شبکه عصبی پیچشی استخراج شده‌اند که در جدول ۶-۶ قرار دارند. مشکل نمونه‌ی دوم و آخر مشابه بودن کلمات مربوط به هر دو رابطه است. اما مشکل نمونه‌های اول، سوم و چهارم در این است که کلمات "اولیاء"، "بازیگرانش" و "مساحتش" در جملات آموزش حضور نداشته‌اند. این مشکل با انجام که پیش‌پردازش و ریشه‌یابی کلمات تا حدود قابل توجهی قابل حل می‌باشد. در این صورت به عنوان مثال کلمه‌ی "بازیگرانش" با کلمه‌ی "بازیگر" جایگزین می‌شود که در کلمات آموزش حضور داشته است.

هرچند هدف دسته‌بند رابطه در این سیستم تشخیص یک رابطه بوده و تک کلاسه می‌باشد و بدین صورت آموزش دیده است اما از دسته‌بند ماشین بردار پشتیبان می‌توان انتظار تشخیص دو یا چند رابطه‌ی مرتبط را داشت. به عنوان نمونه این دسته‌بند برای پرسش "زبان و پایتخت کشور ایران چیست" رابطه‌ی "پایتخت یک کشور" را با احتمال ۶۴٪ و رابطه‌ی "زبان رسمی یک کشور" را با احتمال ۳۱٪ تشخیص می‌دهد. در صورتی که دسته‌بند شبکه عصبی پیچشی برای همین پرسش رابطه‌ی "پایتخت یک کشور" را با احتمال ۹۹٪ تشخیص داده و برای بقیه‌ی روابط احتمال بسیار کمی تولید می‌کند. با جا به جا کردن کلمات همین پرسش و تغییر آن به جمله‌ی "پایتخت و زبان کشور ایران چیست" مشاهده می‌شود که نتیجه‌ی دسته‌بند شبکه عصبی پیچشی تغییر کرده و این بار رابطه‌ی "زبان رسمی

امتیاز $F_1$	فراخوانی (میانگین ماکرو)	دقت (میانگین ماکرو)	صحت	نوع دسته‌بند	روش آزمایش
۹۹.۵۲	۹۹.۴۹	۹۹.۵۵	۹۹.۷۲	CNN	۵ تایی
۹۹.۳۳	۹۹.۳۱	۹۹.۳۵	۹۹.۶۳	SVM	
۹۹.۴۰	۹۹.۳۷	۹۹.۴۴	۹۹.۶۶	CNN	مجموعه داده
۹۹.۰۲	۹۸.۸۹	۹۹.۱۶	۹۹.۴۶	SVM	
۹۶.۵۶	۹۶.۶۳	۹۶.۴۹	۹۸.۲۴	CNN	مجموعه داده الگو متمایز
۹۷.۳۶	۹۷.۱۹	۹۷.۵۴	۹۸.۵۷	SVM	
۹۹.۴۰	۹۹.۳۲	۹۹.۴۹	۹۹.۶۰	CNN	مجموعه داده یک جهته
۹۸.۲۳	۹۸.۱۰	۹۸.۳۷	۹۸.۹۵	SVM	
۹۳.۴۵	۹۳.۲۰	۹۳.۶۹	۹۶.۶۵	CNN	مجموعه داده یک جهته الگو متمایز
۹۲.۷۶	۹۲.۸۶	۹۲.۶۷	۹۵.۹۶	SVM	

جدول ۵-۶: ارزیابی دسته‌بند شبکه عصبی پیچشی و ماشین بردار پشتیبان که از روش تعبیه‌سازی word2vec استفاده کرده‌اند. این دو دسته‌بند به اختصار و به ترتیب با نام‌های CNN و SVM در جدول بیان شده‌اند.

یک کشور را با احتمال ۱۰۰٪ تشخیص می‌دهد. دلیل این عملکرد پیچیده‌تر بودن این شبکه نسبت به دسته‌بند ماشین بردار پشتیبان و آموزش آن برای هدف تک کلاسه می‌باشد.

## ۲-۳-۶ ارزیابی کتابخانه‌ی تشخیص‌دهنده موجودیت‌های نام‌دار

یکی از روش‌های گفته شده برای استخراج موجودیت‌های جملات در این سیستم استفاده از کتابخانه‌ی پیاده شده در آزمایشگاه پردازش زبان‌های طبیعی امیرکبیر می‌باشد. ارزیابی این قسمت در دو سطح

پرسش	رابطه‌ی هدف	رابطه‌ی تشخیص داده شده
اولیاء یاپ کوهن را نام ببرید	آثار یک شخص	فرزندان یک شخص
نفوس خون کاین چند نفر است	جمعیت	جمعیت یک کشور
سریال تلویزیونی معالجه با شوک (فیلم ۱۹۷۳) بازیگرانش کدامند	بازیگران یک سریال تلویزیونی	کارگردان یک سریال تلویزیونی
گمینا تژچیل مساحتش چقدر است	مساحت یک کشور	درآمد
چه کسی سریال تلویزیونی دکتر اورلوف افتتاح را کارگردانی می کند	کارگردان یک سریال تلویزیونی	کارگردان یک فیلم

جدول ۶-۶: نمونه‌های مشکل‌دار دسته‌بند شبکه عصبی پیچشی

توکن<sup>۱۷</sup> و عبارت<sup>۱۸</sup> انجام شده است. در سطح توکن، برچسب<sup>۱۹</sup>های پیش‌بینی شده توسط مدل با برچسب‌های صحیح آن به طور مستقل برای هر کلمه مقایسه می‌شود. این برچسب‌ها می‌توانند نشان‌دهنده‌ی موجودیت بودن یا نبودن آن کلمه باشند. اما در سطح عبارت در صورتی پاسخ مدل به عنوان پاسخ درست برداشت می‌شود که برچسب تمامی کلمات آن جمله مطابق با برچسب‌های صحیح باشد. در سطح عبارت روش دومی نیز برای ارزیابی پیاده شده که در آن اگر تعداد برچسب‌های تولیدی توسط کتابخانه کمتر از تعداد برچسب‌های مورد انتظار باشد، تعدادی برچسب 'o' (که نماینده‌ی موجودیت نبودن است) به انتهای برچسب‌ها اضافه می‌شود و سپس برچسب‌های دو عبارت مقایسه می‌شوند. دلیل این روش بر این اساس است که تشخیص ندادن برچسب برای یک کلمه و یا تشخیص موجودیت نبودن برای آن کلمه در عمل برای سیستم تفاوتی ندارد. نتایج این آزمایش را در جداول ۶-۷ و ۶-۸ مشاهده می‌نمایید. برای این آزمایش از مجموعه داده‌ی الگو متمایز که در بخش پیشین معرفی شد استفاده شده است.

صحت	دقت	فراخوانی	امتیاز $F_1$
۹۵.۱۳	۱۰۰	۸۰.۷۲	۸۹.۳۳

جدول ۶-۷: ارزیابی کتابخانه‌ی تشخیص‌دهنده موجودیت‌های نام‌دار در سطح توکن

نمونه‌هایی از داده‌هایی که این کتابخانه در تشخیص برچسب‌های آنان مشکل داشته را در جدول ۶-۹ مشاهده می‌کنید. برچسب 'o' نماینده‌ی موجودیت نبودن و برچسب 'i' نماینده‌ی موجودیت بودن

Token Level<sup>۱۷</sup>Phrase Level<sup>۱۸</sup>Tag<sup>۱۹</sup>



ورودی	صحت
برچسب تولید شده	۶۷.۳۰
برچسب تکمیل شده	۷۹.۴۲

جدول ۶-۸: ارزیابی کتابخانه‌ی تشخیص‌دهنده موجودیت‌های نام‌دار در سطح عبارت

است. یکی از ایرادات این کتابخانه که در موارد زیادی منجر به اشتباه شده، تشخیص اشتباه کلمات پایانی سوالات مانند "می‌باشد"، "است"، "چه کسی" و "چیست" به عنوان موجودیت است.

پرسش	برچسب هدف	برچسب تشخیص داده شده
چه آثاری مربوط به شخص جاد اپتاو می باشند	oooooiioo	ooooooooo
انتشار چه کتابی توسط نشر اکتیویژن انجام گرفته است	oooooiioo	oooooiiii
والدین جفری (۱۹۵۲-۲۰۱۴) را نام ببرید	oioooo	iiiiooo
ملیت شخص سالی فیلیپس چه نام دارد	ooioooo	ooioooo
ریاست اداره شهرداری سن-فردیناند، کبک در حال حاضر از ان چه شخصی است	ooooioooooooo	ooooiooooooii
سازنده آهنگ فیلم تروریسم (فیلم ۱۹۹۷) چه کسی است	ooioioooo	ooioiooooo
پلاک خودروهای شهر داننبرگ (الب) چند است	ooioioo	ooioioi
آثار حسین مسرور چه نام دارند	oioooo	oioooo
۴۵۷.۳۶ رنمینی میلیارد رنمینی تومان میزان متوسط درآمد سالیانه کدام شرکت است	iiiiooooooo	iiiiiiiiiii
مولفی که حکایت‌نامه (لطائف المتون) را نگارش کرده است کیست	ooiooooo	ooiooooo
نام نگارنده ای که اثر تاریخ حقوق ایران را نگارش کرده است چیست	ooooiooooo	ooooioooooi

جدول ۶-۹: نمونه‌های مشکل‌دار کتابخانه‌ی تشخیص‌دهنده موجودیت‌های نام‌دار (ستون‌های دوم و سوم چپ به راست می‌باشند)

### ۶-۳-۳ ارزیابی یکپارچه‌ی سیستم

برای ارزیابی یکپارچه‌ی سیستم ۱۰٪ از مجموعه داده‌ی اصلی (حدود ۶۵۰۰ داده) تحت عنوان مجموعه‌ی آزمایش جدا شده و بوسیله‌ی آنها ارزیابی انجام می‌شود. در این ارزیابی اولین رابطه‌ی تشخیص داده

شده توسط دسته‌بند و اولین موجودیت تشخیص داده شده در جمله استفاده شده و کوئری به پایگاه دانش ارسال می‌گردد. در صورتی آن داده‌ی آزمون موفقیت‌آمیز تلقی می‌شود که موجودیت پاسخ هدف در مجموعه موجودیت‌های دریافت شده از پایگاه دانش باشد. از آنجا که دسته‌بندها جداگانه آزمایش شده‌اند، در این ارزیابی دسته‌بندهای مورد استفاده روی کلیه‌ی داده‌ها آموزش دیده‌اند تا بهترین عملکرد خود را داشته باشند. برای تشخیص موجودیت‌ها نیز از هر دو روش توضیح داده شده در بخش تشخیص موجودیت‌ها در فصل چهارم استفاده شده است. در جدول ۶-۱۰ نتایج این ارزیابی را که طی دو آزمایش با دسته‌بندهای مختلف انجام شده‌اند مشاهده می‌نمایید.

نوع دسته‌بند	صحت
ماشین بردار پشتیبان (word2vec وزن‌دار)	۲۸.۴۷
شبکه عصبی پیچشی	۲۹.۰۷

جدول ۶-۱۰: ارزیابی یکپارچه‌ی سیستم

با بررسی نتایج این آزمایش مشخص می‌شود که در حدود ۸۰.۲۳٪ موارد دسته‌بند رابطه و کتابخانه‌ی تشخیص‌دهنده موجودیت‌های نام‌دار عملکرد صحیحی داشته‌اند. بنابراین در حدود ۵۱.۵۰٪ موارد با آنکه رابطه و برچسب‌های تشخیص داده شده صحیح بوده‌اند اما در نهایت پاسخ صحیح دریافت نشده است که نشان می‌دهد عمده‌ی مشکل در عملکرد پایگاه دانش فارس‌بیس می‌باشد. ناموفق بودن این موارد در علل زیر خلاصه می‌شوند:

- برخی از موجودیت‌ها دارای کلمات مشابه با موجودیت‌های دیگری هستند. در برخی از این موارد پایگاه دانش گره‌ی نادرستی از پایگاه را به کلمه‌ی مورد نظر نگاشت می‌کند. مانند:

○ قندهار پایتخت کدام کشور است

\* گره صحیح: قندهار

\* گره نگاشت شده: قندهار\_(ملکان)

○ نام بازیگران سریال تلویزیونی دایره راز ذکر کنید

\* گره صحیح: دایره\_راز

\* گره نگاشت شده: دو گره‌ی دایره و راز\_(طایفه) به ترتیب به کلمات مرتبط نگاشت شده‌اند.

- در برخی موارد با آنکه موجودیت مورد پرسش در پایگاه وجود دارد اما هیچ گره‌ای به آن نگاشت نمی‌شود. مانند:

○ مقدار پول کسب شده از فروش فیلم رسوایی ۲

✱ گره صحیح: [رسوایی\\_۲](#)

○ اسم فیلمی که Souheil Ben-Barka کارگردانش است

✱ گره صحیح: [Souheil\\_Ben-Barka](#)

- این پایگاه دانش ویژگی‌هایی که به صورت یک عدد (مانند موجودیت‌های مرتبط با سوالات درآمد، مساحت و جمعیت) یا یک آدرس اینترنتی هستند را به صورت یک گره نگه‌داری نمی‌کند و در تشخیص چنین موجودیت‌هایی ناتوان است. مانند:

○ مقدار ۷۰.۴۰ مربوط به مساحت کدام کشور است

○ سایت [www.kreis-reutlingen.de](http://www.kreis-reutlingen.de) متعلق به چه ارگانی است

با ارزیابی این نمونه‌های مشکل‌دار مشخص می‌شود که در حدود ۹۲٪ نمونه‌ها هیچ گره‌ای نگاشت نمی‌شود که مربوط به علل دوم و سوم است و فقط در ۸٪ موارد نگاشت اشتباه انجام گرفته که مربوط به علت اول در علل ذکر شده می‌باشد.

### ۴-۳-۶ ارزیابی مدل دنباله به دنباله

در فصل دوم با نوعی شبکه‌ی عصبی به نام شبکه‌ی عصبی دنباله به دنباله آشنا شدیم. در آزمایشگاه پردازش زبان طبیعی امیرکبیر سیستم پرسش و پاسخ دیگری وابسته به پایگاه دانش فارس‌بیس توسعه یافته است که از شبکه عصبی دنباله به دنباله به صورت یکپارچه استفاده می‌کند. ورودی این شبکه یک پرسش و خروجی آن کوئری مرتبط با آن پرسش است که با ارسال آن کوئری به پایگاه دانش فارس‌بیس موجودیت پاسخ مرتبط دریافت می‌گردد [۱۵]. در ادامه به ارزیابی مدل دنباله به دنباله و مقایسه‌ی آن با سیستم فعلی موجود در این پروژه می‌پردازیم. در تمامی ارزیابی‌های صورت‌گرفته روی این مدل از مجموعه داده‌ی الگو متمایز که در قسمت ارزیابی دسته‌بندهای رابطه معرفی شد استفاده شده است. در اولین ارزیابی به بررسی رابطه‌ی شناخته شده توسط این مدل می‌پردازیم. به این منظور کوئری تولید شده توسط این مدل دریافت و شناسه‌ی رابطه از کوئری استخراج شده و با رابطه‌ی هدف مقایسه

می‌شود. این ارزیابی صحت ۸۳.۵۸٪ را نتیجه می‌دهد که همانطور که در جدول ۱۱-۶ مشاهده می‌نمایید دسته‌بندهای استفاده شده در این پروژه عملکرد بهتری را در تشخیص رابطه نشان داده‌اند. در این جدول بهترین نتایج دو دسته‌بند در آزمایش با مجموعه داده‌ی الگو متمایز آورده شده‌اند.

مدل	صحت
دسته‌بند ماشین بردار پشتیبان	۹۸.۷۶
دسته‌بند شبکه عصبی پیچشی	۹۸.۲۴
مدل دنباله به دنباله	۸۳.۵۸

جدول ۱۱-۶: مقایسه‌ی مدل دنباله به دنباله و دسته‌بندهای سیستم در تشخیص رابطه

در ارزیابی دیگر، این مدل به صورت سرتاسری مورد آزمایش قرار می‌گیرد. در این ارزیابی کوئری تولید شده توسط مدل به پایگاه دانش ارسال می‌شود و پاسخ‌های دریافتی با پاسخ هدف مقایسه می‌شوند. از آنجا که کوئری‌های تولید شده توسط مدل دنباله به دنباله نتیجه‌ی ضعیفی را به دنبال داشتند، در چند مرحله مورد بهبود قرار گرفته و دوباره ارزیابی شدند. نقص عمده‌ی این مدل در تشخیص اشتباه شناسه‌ی موجودیت موجود در پرسش می‌باشد که از آنجا که این مدل برای تولید کوئری ارتباطی با پایگاه دانش برقرار نمی‌کند قابل انتظار است. مراحل بهبود این مدل به شرح زیر است:

- **تشخیص موجودیت پرسش:** ممکن است در کوئری تولید شده شناسه‌ی موجودیت مورد پرسش در پایگاه دانش فارس‌بیس به اشتباه تشخیص داده شده باشد. برای بهبود می‌توان از روش‌های یادشده در این پروژه برای یافتن موجودیت موجود پرسش استفاده کرد و شناسه‌ی آن موجودیت را بوسیله‌ی رابط برنامه‌نویسی کاربردی فارس‌بیس استخراج و در کوئری جایگزین نمود.
- **استفاده از شناسه‌ی موجودیت اصلی:** این مرحله جایگزین مرحله‌ی قبل می‌باشد. در این مرحله با استفاده از مجموعه داده‌ها شناسه‌ی موجودیت مورد پرسش مستقیماً استفاده شده و در کوئری قرار داده می‌شود. در صورت استفاده از این مرحله دیگر برای تشخیص شناسه‌ی موجودیت نیاز به رابط برنامه‌نویسی کاربردی فارس‌بیس نمی‌باشد. همانطور که مشخص است، این مرحله در سیستم عملی قابل استفاده نیست و صرفاً برای ارزیابی کوئری تولیدی استفاده شده است.
- **بررسی کوئری وارون:** ممکن است شناسه‌ی رابطه و موجودیت پرسش صحیح باشند اما جهت کوئری نادرست باشد. در این مرحله در صورت ناموفق بودن کوئری، جهت کوئری وارون شده و دوباره ارسال و بررسی می‌شود. این مرحله در ادامه‌ی هر دو مرحله‌ی قبلی استفاده می‌شود.

صحت	مرحله
۱.۳۳	کوئری اصلی
۲۵.۷۸	تشخیص موجودیت پرسش
۲۷.۹۱	تشخیص موجودیت پرسش + بررسی کوئری وارون
۳۸.۱۷	استفاده از شناسه‌ی موجودیت اصلی
۴۰.۶۷	استفاده از شناسه‌ی موجودیت اصلی + بررسی کوئری وارون

جدول ۶-۱۲: ارزیابی مدل دنباله به دنباله

نتایج این ارزیابی را در جدول ۶-۱۲ مشاهده می‌کنید. همانطور که از نتایج مشخص است در بهترین حالت قابل استفاده در سیستم عملی و با استفاده از روش‌های تشخیص موجودیت این پروژه صحت ۲۷.۹۱٪ بدست می‌آید که نتیجه‌ای مشابه با سیستم پیاده شده در این پروژه دارد. توجه شود که مرحله‌ی استفاده از شناسه‌ی موجودیت اصلی در سیستم عملی قابل استفاده نمی‌باشد. علت اصلی صحت پایین پاسخ‌ها حتی در صورت استفاده از شناسه‌ی موجودیت اصلی، تعداد زیاد نمونه‌هایی می‌باشد که موجودیت حاضر در سوالشان از نوع عدد یا آدرس اینترنتی بوده و به ازای آنها هیچ گره‌ای در پایگاه دانش وجود ندارد. به همین سبب تولید کوئری برای چنین سوالاتی مقدور نمی‌باشد. این نمونه‌ها علت اصلی صحت پایین در ارزیابی یکپارچه‌ی سیستم نیز بوده‌اند که در بخش قبل به طور مفصل شرح داده شدند.

## فصل هفتم

### جمع‌بندی و نتیجه‌گیری و پیشنهادات

## ۱-۷ جمع‌بندی و نتیجه‌گیری

در این پروژه به طراحی و پیاده‌سازی یک سیستم پرسش و پاسخ خودکار برای زبان فارسی پرداختیم. این سیستم بر اساس یک پایگاه دانش پیاده‌سازی شده و توانایی پاسخ دادن به سوالات ساده را دارا می‌باشد. هدف سیستم‌های پرسش و پاسخ وابسته به گراف دانش<sup>۱</sup> امکان دسترسی راحت‌تر کاربران به اطلاعات گراف دانش می‌باشد به طوری که کاربر نیاز به دانستن ساختار موجود در گراف را نداشته باشد [۱۲].

سیستم موجود در این پروژه را به سه زیرمسئله‌ی ۱. تشخیص رابطه، ۲. تشخیص موجودیت و ۳. تولید کوئری تقسیم کرده و هر کدام را تحلیل و پیاده‌سازی کردیم. در نهایت با ارزیابی هر کدام از قسمت‌های سیستم توانستیم آنرا با یک سیستم مشابه مقایسه کنیم. این پروژه توانست بهبود و عملکرد مناسبی را در چنین سیستم‌هایی بدست بیاورد و همانطور که در فصل قبل ملاحظه شد، این سیستم طی آزمایش‌های مختلف توانست صحت بالای ۹۵٪ و در بهترین حالت صحت ۹۹.۷۲٪ را در تشخیص روابط بدست بیاورد. همچنین این سیستم در تشخیص موجودیت‌ها عملکرد خوبی داشته و توانست امتیاز  $F_1$  ۸۹.۳۳٪ را بدست بیاورد. مشکل اصلی سیستم همانطور که در قسمت ارزیابی یکپارچه‌ی سیستم در فصل قبل بررسی شد، در نگاشت گره‌ی نادرست از گراف دانش به موجودیت‌های شناخته شده است که بهبود این قسمت وابسته به پایگاه دانش مورد استفاده می‌باشد.

## ۲-۷ کارهای آینده

از قدم‌های آتی برای بهبود و پیشرفت این سیستم می‌توان به موارد زیر اشاره کرد:

- ریشه‌یابی کلمات و استفاده از ریشه‌ی کلمات برای تشخیص رابطه و موجودیت. این پیش‌پردازش از بروز مشکلاتی نظیر آنچه در ارزیابی دسته‌بند شبکه عصبی پیچشی مشاهده شد جلوگیری می‌کند.
- در مرحله‌ی شناسایی رابطه، ترتیب کلمات موجود در پرسش اهمیت دارند. به عنوان مثال دو جمله‌ی "پایتخت کشور x کدام است" و "x پایتخت کدام کشور است" دارای کلمات یکسانی می‌باشند اما ترتیب آنها باعث می‌شود که جهت رابطه برعکس شود. با استفاده از یک شبکه‌ی

<sup>۱</sup> Question Answering over Knowledge Graph (QA-KG)

عصبی بازگشتی طی مرحله‌ی تشخیص رابطه، می‌توان ترتیب کلمات را در نظر گرفته و جهت روابط را نیز تشخیص داد.

- پیاده‌سازی دسته‌بند چندکلاسه برای تشخیص هم‌زمان چند رابطه از یک پرسش از نوع ساده. به عنوان مثال پرسش ”پایتخت و زبان رسمی کشور ایران چیست” در واقع دو سوال ساده در قالب یک سوال می‌باشد. این کار نیاز به جمع‌آوری حجم زیادی از داده برای آموزش دارد.
- بهبود عملکرد پایگاه دانش در تشخیص گره‌ی مربوطه به موجودیت‌ها.
- ایجاد رابط برنامه‌نویسی کاربردی جداگانه‌ای در پایگاه دانش برای سوالاتی که موجودیت سوال از نوع عدد یا آدرس اینترنتی می‌باشد.
- اضافه کردن قابلیت پاسخ‌دهی به سوالات پیچیده. در حال حاضر به این منظور روش‌هایی نظیر [۴] و [۱] ارائه شده‌اند.



## منابع و مراجع

- [1] C. Unger, L. Bühmann, J. Lehmann A. Ngonga Ngomo D. Gerber P. Cimiano. Template-based question answering over rdf data. In *Proceedings of the 21st international conference on World Wide Web*, pages 639–648, 2012.
- [2] D. Sorokin, I. Gurevych. End-to-end representation learning for question answering with weak supervision. In *Semantic Web Evaluation Challenge*, Springer, pages 72–78, 2017.
- [3] G. Seni, J. F. Elder. Ensemble methods in data mining: Improving accuracy through combining predictions. *Synthesis Lectures on Data Mining and Knowledge Discovery*, 2(1), 2010.
- [4] H. Zafar, G. Napolitano, J. Lehmann. Formal query generation for question answering over knowledge bases. In *European Semantic Web Conference*, Springer, 2018.
- [5] I. Sutskever, O. Vinyals, Q. V. Le. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems (NIPS) 27*, pages 3104–3112, 2014.
- [6] J. Yin, X. Jiang, Z. Lu L. Shang H. Li X. Li. Neural generative question answering. In *Proceedings of the Workshop on Human-Computer Question Answering*, pages 5046–5052. Association for Computational Linguistics, 2016.

- [7] Kim, Y. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751. Association for Computational Linguistics, 2014.
- [8] M. Asgari Bidhendi, B. Janfada, B. Minaei Bidgoli. Farsbase-kbp: A knowledge base population system for the persian knowledge graph, 2020.
- [9] Patel, K. Mnist handwritten digits classification using a convolutional neural network (cnn). <https://towardsdatascience.com>, 2019.
- [10] Pressman, R. S. *Software Engineering: A Practitioner's Approach*. McGraw-Hill Education, 9 edition, 2019.
- [11] S. Mohammed, P. Shi, J. Lin. Strong baselines for simple question answering over knowledge graphs with and without neural networks. In *Proceedings of the 2018 Conference of the North American*, pages 72–78, 2018.
- [12] X. Huang, J. Zhang, D. Li P. Li. Knowledge graph embedding based question answering. In *The Twelfth ACM International Conference*, pages 105–113, 2019.
- [13] Y. Lan, S. Wang, J. Jiang. Knowledge base question answering with topic units. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pages 5046–5052, 2019.
- [۱۴] سیدمحمدباقر سجادی، بهروز مینایی بیدگلی، مجید عسگری بیدهندی، علی هادیان. ایجاد گراف دانش فارسی چند دامنه ای. در پنجمین همایش مدیران فناوری اطلاعات. پژوهشگاه علوم و فناوری اطلاعات ایران و نهاد ریاست جمهوری، ۱۳۹۶.
- [۱۵] قنبری نسب، سجاد. طراحی و پیاده سازی سیستم پرسش و پاسخ سوالات چندگزینه ای فارسی. دانشگاه صنعتی امیرکبیر، دانشکده مهندسی کامپیوتر و فناوری اطلاعات، ۱۳۹۸.



Amirkabir University of Technology  
(Tehran Polytechnic)

Department of Computer Engineering

Bachelor Thesis

Design and Implementation of an Automatic Question  
Answering System for Answering Simple Persian  
Questions

By

Alireza Torabian

Supervisor

Dr. Saeedeh Momtazi

August 2020