دانشگاه صنعتی مالک اشتر

عنوان گزارش

بررسی و معرفی کتابخانه آنتواسپای

دانشجو

محمدمهدی حسین‌زاده نیگجه

استاد

دکتر سید غلامحسن طباطبایی

تیر ماه ۱۴۰۵

بسم الله الرحمن الرحیم

فهرست

فصل اول مقدمه ۵

فصل دوم معرفی کتابخانه آنتواسپای ۹

۲-۱) تعریف آنتواسپای ۱۰

۲-۲) تاریخچه و هدف از توسعه آنتواسپای ۱۱

۲-۳) توسعه‌دهنده و متن‌باز بودن آنتواسپای ۱۲

۲-۴) کاربرد کتابخانه آنتواسپای ۱۳

۲-۵) معماری و نحوه عملکرد آنتواسپای ۱۴

۲-۶) قابلیت‌ها و امکانات کتابخانه آنتواسپای ۱۷

۲-۶-۱) استخراج کلاس‌ها ۱۷

۲-۶-۲) استخراج ویژگی‌های شیء و داده ۱۷

۲-۶-۳) بازسازی و نمایش سلسله‌مراتب کلاس‌ها ۱۷

۲-۶-۴) استخراج نمونه‌ها ۱۸

۲-۶-۵) استخراج فراداده ۱۸

۲-۶-۶) تولید مستندات خودکار ۱۹

۲-۶-۷) نمایش روابط و گراف‌های معنایی ۱۹

۲-۶-۸) جستجو و پیمایش در هستی‌شناسی ۱۹

۲-۶-۹) قابلیت توسعه و یکپارچه‌سازی ۱۹

فصل سوم نصب و راه‌اندازی ۲۲

۳-۱) پیش‌نیاز های نصب و راه‌اندازی ۲۳

۳-۱-۱) نصب زبان برنامه‌نویسی Python ۲۳

۳-۱-۲) ابزار مدیریت بسته pip ۲۳

۳-۱-۳) استفاده از محیط مجازی ۲۴
۳-۱-۴) محیط توسعه..... ۲۵
۳-۱-۵) فایل هستی‌شناسی ۲۵
۳-۱-۶) اتصال اینترنت..... ۲۵
۳-۲) نصب کتابخانه آنتواسپای..... ۲۶
۳-۳) معرفی سناریوی عملی و فایل هستی‌شناسی ۲۷
۳-۳-۱) خروجی بارگذاری فایل هستی‌شناسی..... ۲۹
۳-۳-۲) دستورات و قابلیت‌های اصلی کتابخانه آنتواسپای ۳۰
فصل چهارم مزایا، محدودیت‌ها و مقایسه کتابخانه OntoSpy با ابزارهای مشابه ۳۳
۴-۱) مزایای کتابخانه آنتواسپای ۳۳
۴-۲) محدودیت‌های کتابخانه آنتواسپای ۳۴
۴-۳) مقایسه آنتواسپای با ابزارهای مشابه ۳۵
فصل پنجم نتیجه‌گیری ۳۷

فهرست جداول و نمودارها

نمودار ۲-۱: معماری آنتواسپای..... ۱۶
جدول ۲-۱: قابلیت‌های کاربردی آنتواسپای..... ۲۰
نمودار ۳-۱: نمودار هستی‌شناسی..... ۲۹
جدول ۳-۱: دستورات کاربردی کتابخانه آنتواسپای..... ۳۰
جدول ۱-۴: جدول مقایسه آنتواسپای و ابزارهای مشابه..... ۳۵

فصل اول

مقدمه

با رشد سریع فناوری اطلاعات و افزایش حجم داده‌های تولیدشده در فضای دیجیتال، دسترسی، مدیریت و پردازش مؤثر اطلاعات به یکی از چالش‌های اصلی دنیای امروز تبدیل شده است. در وب سنتی، اطلاعات عمدتاً به‌گونه‌ای طراحی شده‌اند که برای انسان قابل درک باشند، اما ماشین‌ها تنها قادر به پردازش ساختار ظاهری داده‌ها هستند و درک معنای واقعی اطلاعات برای آن‌ها دشوار است. این محدودیت، توسعه سامانه‌های هوشمند و یکپارچه را با چالش‌هایی مواجه کرده است.

وب معنایی[۱] به‌عنوان نسل جدید فناوری وب، با هدف افزودن معنا و مفهوم به داده‌های موجود در اینترنت معرفی شد. ایده اصلی وب معنایی آن است که اطلاعات علاوه بر قابل‌خواندن بودن برای انسان، برای ماشین‌ها نیز قابل فهم، تفسیر و پردازش باشند. در این رویکرد، داده‌ها به‌گونه‌ای مدل‌سازی می‌شوند که نرم‌افزارها بتوانند روابط میان مفاهیم را تشخیص داده، اطلاعات را به‌صورت هوشمند ترکیب کنند و در تصمیم‌گیری، بازیابی اطلاعات و استنتاج دانش از آن‌ها استفاده نمایند.

یکی از مهم‌ترین اجزای وب معنایی، هستی‌شناسی[۲] است. هستی‌شناسی یک مدل رسمی و صریح برای نمایش دانش یک حوزه خاص محسوب می‌شود که مفاهیم، ویژگی‌ها، روابط میان مفاهیم و قوانین حاکم بر آن حوزه را تعریف می‌کند. به بیان دیگر، هستی‌شناسی واژگان مشترکی را فراهم می‌آورد که از طریق

---

[۱] Semantic Web
[۲] Ontology

آن انسان‌ها و سامانه‌های نرم‌افزاری می‌توانند برداشت یکسانی از اطلاعات داشته باشند. این ویژگی باعث افزایش قابلیت تبادل داده، یکپارچه‌سازی اطلاعات و همکاری میان سامانه‌های مختلف می‌شود.

اهمیت هستی‌شناسی تنها به وب معنایی محدود نمی‌شود، بلکه در حوزه‌های مختلفی مانند هوش مصنوعی، مهندسی دانش، بازیابی اطلاعات، سیستم‌های توصیه‌گر، اینترنت اشیا، پزشکی، زیست‌فناوری و تجارت الکترونیکی نیز کاربرد گسترده‌ای دارد. استفاده از هستی‌شناسی موجب کاهش ابهام در داده‌ها، افزایش قابلیت استفاده مجدد از دانش، بهبود تعامل میان سامانه‌های ناهمگون و فراهم شدن امکان استنتاج خودکار دانش می‌شود.

هستی‌شناسی یکی از مفاهیم بنیادی در حوزه وب معنایی، مهندسی دانش و هوش مصنوعی است. این مفهوم در علوم رایانه به مدلی رسمی و صریح برای نمایش دانش یک حوزه خاص اشاره دارد که در آن مفاهیم، ویژگی‌ها، روابط میان مفاهیم و قوانین حاکم بر آن حوزه تعریف می‌شوند. هدف اصلی هستی‌شناسی، ایجاد درکی مشترک از دانش میان انسان‌ها و سامانه‌های نرم‌افزاری است تا امکان تبادل، اشتراک‌گذاری و استفاده مجدد از اطلاعات فراهم شود.

یکی از شناخته‌شده‌ترین تعاریف هستی‌شناسی توسط Thomas R. Gruber ارائه شده است. وی هستی‌شناسی را «یک مشخصه رسمی و صریح از یک مفهوم‌سازی مشترک» تعریف می‌کند. در این تعریف، «رسمی» به معنای قابل پردازش بودن توسط ماشین، «صریح» به معنای تعریف دقیق مفاهیم و روابط و «مشترک» به معنای توافق میان افراد یا سامانه‌هایی است که از آن استفاده می‌کنند. این تعریف مبنای بسیاری از پژوهش‌ها و استانداردهای امروزی در زمینه وب معنایی قرار گرفته است.

یک هستی‌شناسی معمولاً از اجزای مختلفی تشکیل شده است که هر یک نقش مشخصی در نمایش دانش دارند. مهم‌ترین اجزای یک هستی‌شناسی عبارت‌اند از:

کلاس‌ها[۳]: کلاس‌ها مفاهیم یا دسته‌های اصلی موجود در یک حوزه را نمایش می‌دهند. برای مثال، در یک هستی‌شناسی دانشگاهی، کلاس‌هایی مانند «دانشجو»، «استاد» و «درس» می‌توانند وجود داشته باشند.

نمونه‌ها[۴]: نمونه‌ها اعضای واقعی یک کلاس هستند. به‌عنوان نمونه، «علی» می‌تواند یک نمونه از کلاس «دانشجو» باشد.

[۳] Classes
[۴] Individuals or Instances

ویژگی‌ها[۵]: ویژگی‌ها روابط یا خصوصیات مرتبط با کلاس‌ها و نمونه‌ها را مشخص می‌کنند. ویژگی‌ها معمولاً به دو دسته تقسیم می‌شوند:

ویژگی‌های شیء[۶]: برای ایجاد ارتباط میان دو نمونه یا دو کلاس به‌کار می‌روند؛ مانند رابطه «انتخاب می‌کند» بین دانشجو و درس.

ویژگی‌های داده‌ای[۷]: برای اختصاص مقادیر داده‌ای مانند نام، سن، شناسه یا تاریخ به نمونه‌ها استفاده می‌شوند.

روابط[۸]: روابط نحوه ارتباط میان مفاهیم مختلف را مشخص می‌کنند و ساختار منطقی دانش را شکل می‌دهند.

قیود و قوانین[۹]: این بخش شامل قوانین منطقی، محدودیت‌ها و اصولی است که نحوه ارتباط مفاهیم را تعیین می‌کنند و امکان استنتاج دانش جدید را فراهم می‌سازند.

هستی‌شناسی‌ها بر اساس دامنه کاربرد و میزان عمومی بودن، به انواع مختلفی تقسیم می‌شوند که مهم‌ترین آن‌ها عبارت‌اند از:

هستی‌شناسی سطح بالا[۱۰]: شامل مفاهیم بسیار کلی و مستقل از حوزه است؛ مانند زمان، مکان، شیء و رویداد. این نوع هستی‌شناسی به‌عنوان پایه‌ای برای توسعه سایر هستی‌شناسی‌ها مورد استفاده قرار می‌گیرد.

هستی‌شناسی دامنه[۱۱]: دانش مربوط به یک حوزه تخصصی مانند پزشکی، آموزش، کشاورزی یا بانکداری را مدل‌سازی می‌کند. این نوع از رایج‌ترین انواع هستی‌شناسی در پروژه‌های کاربردی است.

هستی‌شناسی وظیفه[۱۲]: دانش مرتبط با انجام یک وظیفه یا فرایند خاص، مانند تشخیص بیماری، برنامه‌ریزی یا تصمیم‌گیری را توصیف می‌کند.

---

[۵] Properties
[۶] Object Properties
[۷] Data Properties
[۸] Relations
[۹] Constraints and Axioms
[۱۰] Upper Ontology
[۱۱] Domain Ontology
[۱۲] Task Ontology

هستی‌شناسی کاربردی[۱۳]: برای یک سامانه یا کاربرد مشخص طراحی می‌شود و معمولاً ترکیبی از هستی‌شناسی‌های دامنه و وظیفه است.

استفاده از هستی‌شناسی مزایای متعددی از جمله ایجاد زبان مشترک برای نمایش دانش، کاهش ابهام در داده‌ها، افزایش قابلیت استفاده مجدد از دانش، تسهیل یکپارچه‌سازی اطلاعات و فراهم کردن امکان استنتاج خودکار را به همراه دارد. به همین دلیل، هستی‌شناسی یکی از مهم‌ترین فناوری‌های مورد استفاده در سامانه‌های مبتنی بر وب معنایی، هوش مصنوعی، سیستم‌های خبره، موتورهای جستجوی معنایی و مدیریت دانش به شمار می‌رود.

از آنجا که هستی‌شناسی‌ها ممکن است دارای ساختارهای پیچیده و تعداد زیادی کلاس، ویژگی و رابطه باشند، بررسی و تحلیل آن‌ها بدون استفاده از ابزارهای تخصصی دشوار است. ازاین‌رو، ابزارهایی مانند آنتواسپای[۱۴] توسعه یافته‌اند تا امکان تحلیل ساختار، استخراج اجزای هستی‌شناسی، تولید مستندات و نمایش روابط میان مفاهیم را به‌صورت خودکار فراهم کنند. در بخش‌های بعدی، این کتابخانه و قابلیت‌های آن به‌طور کامل بررسی خواهد شد.

با گسترش استفاده از هستی‌شناسی‌ها و افزایش پیچیدگی آن‌ها، نیاز به ابزارهایی برای ایجاد، تحلیل، اعتبارسنجی، مستندسازی و تجسم ساختار این مدل‌های دانشی بیش از پیش احساس می‌شود. این ابزارها به پژوهشگران و توسعه‌دهندگان کمک می‌کنند تا ساختار هستی‌شناسی‌ها را بهتر درک کرده، کیفیت آن‌ها را ارزیابی کنند و فرآیند توسعه و نگهداری آن‌ها را تسهیل نمایند. یکی از این ابزارهای متن‌باز و کاربردی، کتابخانه OntoSpy است که با ارائه امکانات متنوع برای تحلیل و مستندسازی هستی‌شناسی‌ها، نقش مهمی در توسعه پروژه‌های مبتنی بر وب معنایی ایفا می‌کند.

[۱۳] Application Ontology
[۱۴] OntoSpy

فصل دوم
معرفی کتابخانه آنتواسپای

هستی‌شناسی‌ها معمولاً شامل تعداد زیادی کلاس، ویژگی، روابط و نمونه هستند و بررسی ساختار آن‌ها به‌صورت دستی علاوه بر زمان‌بر بودن، احتمال بروز خطا را نیز افزایش می‌دهد. ازاین‌رو، استفاده از ابزارهای نرم‌افزاری که بتوانند ساختار هستی‌شناسی را به‌صورت خودکار تحلیل و نمایش دهند، نقش مهمی در فرآیند توسعه، نگهداری و ارزیابی این مدل‌ها ایفا می‌کند.

در میان ابزارهای موجود، آنتواسپای یکی از کتابخانه‌های متن‌باز و پرکاربرد در زبان برنامه‌نویسی پایتون است که با هدف تسهیل تحلیل و مستندسازی هستی‌شناسی‌های مبتنی بر استانداردهای وب معنایی توسعه یافته است. این کتابخانه امکان بارگذاری فایل‌های هستی‌شناسی، استخراج اطلاعات ساختاری، نمایش سلسله‌مراتب کلاس‌ها، شناسایی ویژگی‌ها و روابط، و تولید مستندات قابل‌فهم از ساختار دانش را فراهم می‌کند. همچنین، با بهره‌گیری از کتابخانه‌های استاندارد پایتون در حوزه پردازش داده‌های معنایی، آنتواسپای بستری مناسب برای پژوهشگران، توسعه‌دهندگان و مهندسان دانش فراهم کرده است تا بتوانند هستی‌شناسی‌های خود را با دقت بیشتری بررسی و تحلیل کنند.

در این فصل، ابتدا کتابخانه آنتواسپای معرفی شده و اهداف طراحی، ویژگی‌های کلی و ساختار آن مورد بررسی قرار می‌گیرد. سپس نحوه عملکرد، قابلیت‌های اصلی، مزایا، محدودیت‌ها و کاربردهای این کتابخانه تشریح می‌شود تا درک جامعی از امکانات آن به‌دست آید. این مباحث زمینه لازم را برای بررسی عملی و ارزیابی عملکرد آنتواسپای در فصل‌های بعد فراهم می‌کنند.

۲-۱) تعریف آنتواسپای

آنتواسپای یک کتابخانه متن‌باز[۱۵] مبتنی بر زبان برنامه‌نویسی پایتون است که برای تحلیل، کاوش، مستندسازی و تجسم هستی‌شناسی‌های مبتنی بر فناوری‌های وب معنایی توسعه یافته است. این کتابخانه به کاربران امکان می‌دهد تا فایل‌های هستی‌شناسی را در قالب‌های استاندارد وب معنایی بارگذاری کرده و ساختار آن‌ها را به‌صورت خودکار بررسی کنند. هدف اصلی آنتواسپای، ساده‌سازی فرآیند درک و تحلیل ساختار هستی‌شناسی‌ها و ارائه اطلاعاتی جامع درباره اجزای تشکیل‌دهنده آن‌ها است.

آنتواسپای با استفاده از قابلیت‌های پردازش داده‌های معنایی، اطلاعات موجود در یک هستی‌شناسی را استخراج کرده و آن‌ها را به شکلی سازمان‌یافته نمایش می‌دهد. این اطلاعات شامل کلاس‌ها، ویژگی‌های شیء[۱۶]، ویژگی‌های داده‌ای[۱۷]، نمونه‌ها، روابط میان مفاهیم و سایر عناصر تعریف‌شده در هستی‌شناسی است. علاوه بر این، کتابخانه قادر است سلسله‌مراتب کلاس‌ها و روابط میان آن‌ها را استخراج کرده و نمایی ساخت‌یافته از مدل دانشی ارائه دهد که درک و تحلیل آن را برای کاربران آسان‌تر می‌کند.

یکی از ویژگی‌های مهم این کتابخانه، پشتیبانی از استانداردهای رایج وب معنایی از جمله RDF ، RDFS و OWL است. این قابلیت باعث می‌شود که این کتابخانه بتواند با طیف گسترده‌ای از هستی‌شناسی‌های موجود سازگار باشد و در پروژه‌های مختلف پژوهشی و صنعتی مورد استفاده قرار گیرد. همچنین، آنتواسپای برای پردازش فایل‌های هستی‌شناسی از کتابخانه‌های معتبر پایتون، به‌ویژه RDFLib ، بهره می‌گیرد که امکان خواندن و مدیریت گراف‌های معنایی را فراهم می‌کند.

از دیگر قابلیت‌های مهم این کتابخانه می‌توان به تولید مستندات خودکار از ساختار هستی‌شناسی اشاره کرد. آنتواسپای می‌تواند اطلاعات استخراج‌شده را در قالب گزارش‌های ساخت‌یافته و صفحات HTML ارائه دهد و در برخی موارد، نمایش گرافیکی روابط میان مفاهیم را نیز فراهم کند. این قابلیت‌ها به پژوهشگران، توسعه‌دهندگان و مهندسان دانش کمک می‌کند تا بدون نیاز به بررسی دستی فایل‌های پیچیده OWL یا RDF ، ساختار هستی‌شناسی را به‌سرعت درک کرده و مستندسازی کنند.

به دلیل سادگی استفاده، متن‌باز بودن، قابلیت توسعه و پشتیبانی از استانداردهای وب معنایی، آنتواسپای به یکی از ابزارهای پرکاربرد در حوزه مهندسی دانش، وب معنایی، مدیریت دانش و پژوهش‌های مرتبط با هستی‌شناسی تبدیل شده است. این کتابخانه علاوه بر استفاده در پروژه‌های تحقیقاتی، در آموزش مفاهیم وب معنایی و تحلیل

---

[۱۵] Open Source
[۱۶] Object Properties
[۱۷] Data Properties

ساختار هستی‌شناسی‌ها نیز کاربرد گسترده‌ای دارد و بستری مناسب برای توسعه ابزارها و سامانه‌های مبتنی بر دانش فراهم می‌کند.

۲-۲) تاریخچه و هدف از توسعه آنتواسپای

با توسعه فناوری‌های وب معنایی و افزایش استفاده از زبان‌های استانداردی مانند RDF[۱۸] ، RDFS[۱۹] و WOL[۲۰] ، حجم و پیچیدگی هستی‌شناسی‌های تولیدشده در حوزه‌های مختلف به‌طور قابل‌توجهی افزایش یافت. این موضوع باعث شد که تحلیل ساختار، بررسی روابط میان مفاهیم، مستندسازی و ارزیابی کیفیت هستی‌شناسی‌ها به یکی از چالش‌های اصلی در مهندسی دانش تبدیل شود. در حالی که ابزارهایی مانند Protégé عمدتاً بر طراحی و ویرایش هستی‌شناسی تمرکز دارند، نیاز به ابزاری سبک، قابل برنامه‌نویسی و مناسب برای تحلیل خودکار ساختار هستی‌شناسی‌ها در محیط‌های پژوهشی و توسعه نرم‌افزار احساس می‌شد.

در پاسخ به این نیاز، کتابخانه آنتواسپای به‌عنوان یک پروژه متن‌باز مبتنی بر زبان برنامه‌نویسی پایتون توسعه یافت. هدف اصلی از طراحی این کتابخانه، فراهم کردن بستری برای کاوش[۲۱]، تحلیل ساختاری[۲۲]، استخراج فراداده[۲۳] و تولید مستندات خودکار[۲۴] از هستی‌شناسی‌های مبتنی بر استانداردهای وب معنایی بود. این کتابخانه به گونه‌ای طراحی شده است که بدون نیاز به پیاده‌سازی الگوریتم‌های پیچیده پردازش RDF ، اطلاعات ساختاری موجود در یک هستی‌شناسی را استخراج کرده و در قالبی قابل فهم در اختیار پژوهشگران و توسعه‌دهندگان قرار دهد.

معماری این کتابخانه بر پایه پردازش گراف‌های RDF بنا شده است. این کتابخانه با بهره‌گیری از قابلیت‌های کتابخانه RDFLib ، فایل‌های هستی‌شناسی را به یک گراف معنایی تبدیل کرده و سپس با پیمایش این گراف، مؤلفه‌های اصلی هستی‌شناسی شامل کلاس‌ها، زیرکلاس‌ها، ویژگی‌های شیء، ویژگی‌های داده‌ای، نمونه‌ها، محدودیت‌ها[۲۵]، دامنه[۲۶]، برد[۲۷] و روابط سلسله‌مراتبی را استخراج می‌کند. در ادامه، اطلاعات استخراج‌شده در قالب یک مدل شیءگرا سازمان‌دهی می‌شوند که امکان دسترسی برنامه‌نویسی به عناصر مختلف هستی‌شناسی را از طریق API کتابخانه فراهم می‌سازد.

---

[۱۸] Resource Description Framework
[۱۹] RDF Schema
[۲۰] Web Ontology Language
[۲۱] Ontology Exploration
[۲۲] Structural Analysis
[۲۳] Metadata Extraction
[۲۴] Automatic Documentation Generation
[۲۵] Restrictions
[۲۶] Domain
[۲۷] Range

یکی از اهداف کلیدی در توسعه ، ایجاد ابزاری مستقل از دامنه[۲۸] بوده است؛ به این معنا که کتابخانه بتواند بدون وابستگی به یک حوزه تخصصی خاص، انواع مختلف هستی‌شناسی‌های مبتنی بر RDF و OWL را پردازش کند. به همین دلیل، این کتابخانه از استانداردهای توصیه‌شده توسط کنسرسیوم وب جهان‌گستر (C۳W) پیروی کرده و با اغلب هستی‌شناسی‌های تولیدشده در حوزه‌هایی مانند پزشکی، زیست‌اطلاع‌رسانی، اینترنت اشیا، دولت الکترونیک، آموزش، تجارت الکترونیکی و مدیریت دانش سازگار است.

از منظر طراحی نرم‌افزار، تنها یک ابزار نمایش اطلاعات نیست، بلکه به‌عنوان یک لایه انتزاعی[۲۹] بر روی داده‌های RDFعمل می‌کند. این لایه، پیچیدگی‌های مربوط به پردازش گراف‌های معنایی را از دید توسعه‌دهنده پنهان کرده و امکان دسترسی به ساختار هستی‌شناسی از طریق مجموعه‌ای از کلاس‌ها و متدهای سطح بالا را فراهم می‌آورد. این ویژگی موجب شده است که آنتواسپای علاوه بر استفاده در مستندسازی، در توسعه سامانه‌های مبتنی بر دانش، ابزارهای تحلیل هستی‌شناسی، سامانه‌های ارزیابی کیفیت آنتواسپای و پروژه‌های پژوهشی مرتبط با مهندسی دانش نیز به‌کار گرفته شود.

در نسخه‌های مختلف این کتابخانه، تمرکز توسعه‌دهندگان بر افزایش سازگاری با استانداردهای جدید وب معنایی، بهبود عملکرد در پردازش هستی‌شناسی‌های بزرگ، توسعه ابزارهای مستندسازی HTML و فراهم کردن امکان یکپارچه‌سازی با سایر کتابخانه‌های اکوسیستم پایتون بوده است. این رویکرد سبب شده است که به یکی از ابزارهای شناخته‌شده برای تحلیل ساختاری و مستندسازی خودکار هستی‌شناسی‌ها در پروژه‌های تحقیقاتی و آموزشی تبدیل شود.

۲-۳) توسعه‌دهنده و متن‌باز بودن آنتواسپای

کتابخانه آنتواسپای توسط Michele Pasin توسعه داده شده است. این پروژه با هدف فراهم کردن ابزاری سبک، منعطف و قابل برنامه‌نویسی برای تحلیل، کاوش و مستندسازی هستی‌شناسی‌های مبتنی بر استانداردهای وب معنایی ایجاد شد. طراحی آن بر پایه زبان برنامه‌نویسی Python انجام شده است و از ابتدا با رویکرد توسعه متن‌باز در اختیار جامعه توسعه‌دهندگان و پژوهشگران قرار گرفته است.

متن‌باز بودن این کتابخانه این امکان را فراهم می‌کند که کد منبع آن برای همه کاربران در دسترس باشد و پژوهشگران بتوانند نحوه پیاده‌سازی الگوریتم‌ها، ساختار نرم‌افزار و روش‌های پردازش داده‌های معنایی را بررسی کنند. علاوه بر این، توسعه‌دهندگان قادر هستند قابلیت‌های جدیدی به کتابخانه اضافه کرده، خطاهای احتمالی را

[۲۸] Domain-Independent
[۲۹] Abstraction Layer

برطرف کنند و آن را متناسب با نیازهای پروژه‌های خود سفارشی‌سازی نمایند. این ویژگی، آنتواسپای را به ابزاری مناسب برای پروژه‌های تحقیقاتی و توسعه‌ای تبدیل کرده است.

یکی دیگر از مزایای متن‌باز بودن این کتابخانه، بهره‌مندی از مشارکت جامعه کاربری است. توسعه مستمر، گزارش خطاها، ارائه پیشنهادهای بهبود و انتشار نسخه‌های جدید با همکاری کاربران و توسعه‌دهندگان انجام می‌شود که این امر موجب افزایش پایداری، بهبود عملکرد و سازگاری بیشتر کتابخانه با استانداردهای جدید وب معنایی شده است.

از منظر فنی، کد منبع آنتواسپای بر روی مخزن GitHub[۳۰] منتشر شده و فرآیند توسعه آن مطابق با الگوی رایج پروژه‌های متن‌باز انجام می‌شود. این موضوع علاوه بر شفافیت در توسعه نرم‌افزار، امکان بررسی تغییرات، مشارکت در توسعه، ثبت مشکلات[۳۱] و مدیریت نسخه‌ها را برای کاربران فراهم می‌کند. همچنین انتشار این کتابخانه از طریق مخزن PyPI[۳۲] ، نصب و به‌روزرسانی آن را با استفاده از ابزار pip برای کاربران پایتون ساده کرده است.

متن‌باز بودن آنتواسپای، در کنار استفاده از فناوری‌ها و کتابخانه‌های استاندارد اکوسیستم پایتون، موجب شده است که این ابزار به‌راحتی در پروژه‌های پژوهشی، سامانه‌های مبتنی بر وب معنایی و ابزارهای مهندسی دانش ادغام شود. این ویژگی‌ها، آنتواسپای را به یکی از گزینه‌های مناسب برای تحلیل و مستندسازی خودکار هستی‌شناسی‌ها در محیط‌های دانشگاهی و صنعتی تبدیل کرده است.

۲-۴) کاربرد کتابخانه آنتواسپای

یکی از مهم‌ترین کاربردهای آنتواسپای، تحلیل ساختار داخلی هستی‌شناسی است. این کتابخانه با پیمایش گراف دانش، کلاس‌ها، زیرکلاس‌ها، ویژگی‌های شیء، ویژگی‌های داده‌ای، افراد، دامنه، برد، قیود و روابط معنایی را استخراج کرده و آن‌ها را در قالب ساختارهای سلسله‌مراتبی و قابل پردازش ارائه می‌دهد. این قابلیت، امکان بررسی معماری مفهومی یک هستی‌شناسی و شناسایی وابستگی‌های میان اجزای آن را برای پژوهشگران فراهم می‌کند.

از دیگر کاربردهای مهم آنتواسپای، مستندسازی خودکار است. در پروژه‌های بزرگ، بررسی مستقیم فایل‌های OWL یا RDF به دلیل حجم زیاد سه‌تایی‌های معنایی[۳۳] و پیچیدگی روابط، فرآیندی زمان‌بر و مستعد خطا است. آنتواسپای با استخراج فراداده‌ها و تولید مستندات ساخت‌یافته، اطلاعات مورد نیاز را به شکلی سازمان‌یافته در اختیار توسعه‌دهندگان قرار می‌دهد. این مستندات معمولاً شامل سلسله‌مراتب کلاس‌ها، مشخصات ویژگی‌ها، روابط

[۳۰] https://github.com/lambdamusic/Ontospy
[۳۱] Issue Tracking
[۳۲] Python Package Index
[۳۳] RDF Triples

میان موجودیت‌ها، تعاریف واژگان و اطلاعات توصیفی هر عنصر هستند و نقش مهمی در نگهداری، توسعه و اشتراک‌گذاری هستی‌شناسی ایفا می‌کنند.

این کتابخانه همچنین در مهندسی معکوس هستی‌شناسی[۳۴] کاربرد دارد. در بسیاری از پروژه‌ها، مستندات اولیه یک هستی‌شناسی در دسترس نیست یا در طول زمان به‌روزرسانی نشده است. آنتواسپای با استخراج ساختار مفهومی از فایل‌های موجود، امکان بازسازی نمای منطقی هستی‌شناسی و درک معماری آن را بدون نیاز به بررسی مستقیم کدهای RDF/XML یا Turtle فراهم می‌سازد.

از منظر توسعه نرم‌افزار، آنتواسپای به‌عنوان یک رابط برنامه‌نویسی کاربردی (API) برای تعامل با داده‌های معنایی نیز مورد استفاده قرار می‌گیرد. توسعه‌دهندگان می‌توانند از کلاس‌ها و متدهای این کتابخانه برای استخراج اطلاعات، انجام پیمایش روی گراف دانش، تولید گزارش‌های تحلیلی، پیاده‌سازی ابزارهای ارزیابی کیفیت هستی‌شناسی و توسعه سامانه‌های مبتنی بر دانش استفاده کنند. به دلیل پیاده‌سازی این کتابخانه بر پایه RDFLib، امکان یکپارچه‌سازی آن با سایر ابزارهای پردازش RDF در اکوسیستم پایتون نیز وجود دارد.

علاوه بر کاربردهای پژوهشی، در آموزش مفاهیم وب معنایی نیز نقش مؤثری ایفا می‌کند. نمایش ساخت‌یافته اجزای یک هستی‌شناسی، تولید نمودارهای سلسله‌مراتبی و ارائه مستندات خوانا، درک مفاهیمی مانند کلاس‌ها، ویژگی‌ها، روابط و ساختار گراف دانش را برای دانشجویان و پژوهشگران تسهیل می‌کند. از این‌رو، این کتابخانه نه‌تنها ابزاری برای تحلیل فنی هستی‌شناسی‌ها، بلکه بستری برای مطالعه، آموزش، ارزیابی و توسعه سامانه‌های مبتنی بر دانش در حوزه‌های مختلف از جمله هوش مصنوعی، مدیریت دانش، زیست‌اطلاع‌رسانی، اینترنت اشیا و یکپارچه‌سازی داده‌های معنایی به شمار می‌آید.

۲-۵) معماری و نحوه عملکرد آنتواسپای

معماری کتابخانه آنتو اسپای بر پایه مدل گرافی استاندارد RDF طراحی شده است و از رویکردی چندلایه برای پردازش، تحلیل و مستندسازی هستی‌شناسی‌ها استفاده می‌کند. این کتابخانه با بهره‌گیری از قابلیت‌های کتابخانه RDFLib، داده‌های معنایی موجود در فایل‌هایRDF ، RDFSو OWL را دریافت کرده، آن‌ها را به یک گراف دانش[۳۵] تبدیل می‌کند و سپس با پیمایش این گراف، ساختار مفهومی هستی‌شناسی را استخراج و مدل‌سازی می‌نماید.

فرآیند عملکرد آنتواسپای از مرحله بارگذاری فایل هستی‌شناسی آغاز می‌شود. در این مرحله، فایل‌های ذخیره‌شده در قالب‌هایی مانندRDF/XML ، Turtle (TTL)، N-Triples، JSON-LD و سایر قالب‌های استاندارد توسط

[۳۴] Ontology Reverse Engineering
[۳۵] Knowledge Graph

RDFLibتجزیه[۳۶] شده و به یک مجموعه از سه‌تایی‌های RDF شاملSubject ، Predicateو Object تبدیل می‌شوند. این سه‌تایی‌ها نمایش پایه‌ای دانش در وب معنایی هستند و روابط میان موجودیت‌ها را در قالب یک گراف جهت‌دار توصیف می‌کنند.

پس از تشکیل گراف، عملیات تحلیل معنایی را آغاز می‌کند. در این مرحله، کتابخانه با استفاده از واژگان استاندارد OWLوRDFS ، عناصر اصلی هستی‌شناسی را شناسایی و دسته‌بندی می‌کند. این عناصر شامل کلاس‌ها، زیرکلاس‌ها، ویژگی‌های شیء، ویژگی‌های داده‌ای، افراد، دامنه، برد، محدودیت‌های منطقی و سایر روابط معنایی هستند. اطلاعات استخراج‌شده در ساختارهای شیءگرای داخلی سازمان‌دهی می‌شوند تا دسترسی به آن‌ها از طریق رابط برنامه‌نویسی کتابخانه (API) امکان‌پذیر باشد.

یکی از مهم‌ترین ویژگی‌های معماری آنتواسپای، بازسازی سلسله‌مراتب مفاهیم[۳۷] است. این کتابخانه با تحلیل روابطی مانند rdfs:subClassOf وrdfs:subPropertyOf ، ساختار درختی کلاس‌ها و ویژگی‌ها را استخراج کرده و ارتباط میان مفاهیم را در قالب سلسله‌مراتب منطقی نمایش می‌دهد. این قابلیت به پژوهشگران امکان می‌دهد تا سازمان مفهومی دانش را بدون نیاز به بررسی مستقیم فایل‌های RDF یا OWL مشاهده و تحلیل کنند.

علاوه بر استخراج عناصر اصلی، اطلاعات توصیفی موجود در هستی‌شناسی را نیز بازیابی می‌کند. این اطلاعات شامل برچسب‌ها[۳۸]، توضیحات[۳۹]، شناسه‌ها[۴۰]، فضای نام[۴۱]، نسخه هستی‌شناسی، واردسازی‌ها[۴۲] و سایر فراداده‌های تعریف‌شده در واژگان استاندارد وب معنایی است. وجود این اطلاعات نقش مهمی در مستندسازی و تفسیر صحیح اجزای هستی‌شناسی دارد.

در لایه بالاتر، امکاناتی برای تولید مستندات خودکار فراهم می‌کند. داده‌های استخراج‌شده از گراف دانش در قالب صفحاتHTML ، گزارش‌های ساخت‌یافته و نمودارهای سلسله‌مراتبی سازمان‌دهی می‌شوند. این مستندات به کاربران اجازه می‌دهند بدون نیاز به مطالعه مستقیم ساختارRDF ، به اطلاعاتی مانند روابط میان کلاس‌ها، ویژگی‌ها، نمونه‌ها و مشخصات هر موجودیت دسترسی داشته باشند. در پروژه‌های بزرگ، این قابلیت نقش مهمی در نگهداری، توسعه و مستندسازی هستی‌شناسی ایفا می‌کند.

از منظر معماری نرم‌افزار، آنتواسپای را می‌توان به چهار لایه اصلی تقسیم کرد:

---

[۳۶] Parsing
[۳۷] Ontology Hierarchy Reconstruction
[۳۸] Labels
[۳۹] Comments
[۴۰] Identifiers
[۴۱] Namespaces
[۴۲] Imports

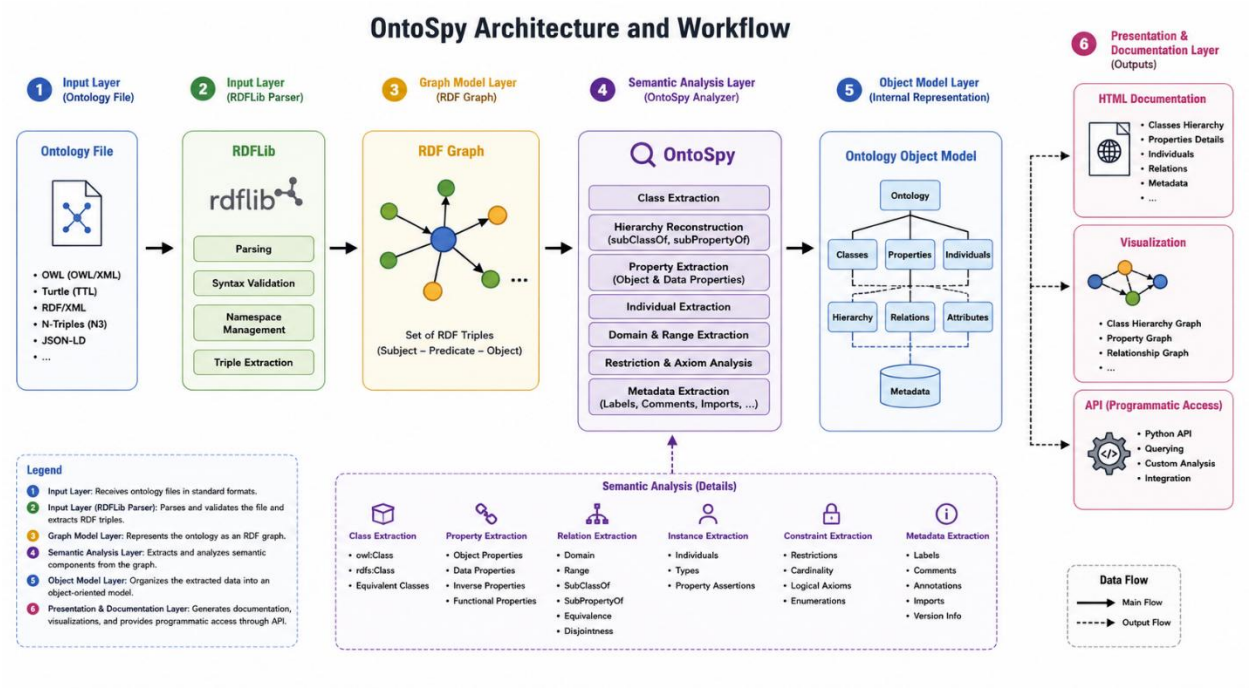
لایه دریافت داده[۴۳]: مسئول دریافت و تجزیه فایل‌هایRDF ، OWL، Turtleو سایر قالب‌های استاندارد با استفاده از RDFLib است.

لایه مدل‌سازی گراف[۴۴]: مسئول ایجاد گراف RDF و مدیریت ساختار سه‌تایی‌های معنایی است.

لایه تحلیل معنایی[۴۵]: وظیفه استخراج کلاس‌ها، ویژگی‌ها، روابط، نمونه‌ها، قیود و فراداده‌ها را بر عهده دارد.

لایه ارائه و مستندسازی[۴۶]: نتایج تحلیل را در قالبAPI ، گزارش‌های متنی، مستندات HTML و نمایش‌های ساختاری در اختیار کاربر قرار می‌دهد.

نمودار ۲-۱: معماری آنتواسپای



این معماری لایه‌ای باعث شده است که آنتواسپای علاوه بر عملکرد مناسب در تحلیل و مستندسازی هستی‌شناسی‌ها، قابلیت توسعه و یکپارچه‌سازی با سایر ابزارهای پردازش داده‌های معنایی را نیز داشته باشد. همچنین جداسازی وظایف در لایه‌های مختلف، امکان استفاده از این کتابخانه را در پروژه‌های پژوهشی، سامانه‌های

[۴۳] Input Layer
[۴۴] Graph Model Layer
[۴۵] Semantic Analysis Layer
[۴۶] Presentation & Documentation Layer

مدیریت دانش و ابزارهای تحلیل هستی‌شناسی فراهم کرده و توسعه‌دهندگان می‌توانند با استفاده از API آن، قابلیت‌های سفارشی مورد نیاز خود را بر روی ساختارهای استخراج‌شده پیاده‌سازی کنند.

۲-۶) قابلیت‌ها و امکانات کتابخانه آنتواسپای

کتابخانه آنتواسپای مجموعه‌ای از قابلیت‌های تخصصی را برای تحلیل، کاوش و مستندسازی هستی‌شناسی‌های مبتنی بر استانداردهای وب معنایی ارائه می‌دهد. این قابلیت‌ها با هدف ساده‌سازی فرآیند بررسی ساختار دانش، استخراج اطلاعات معنایی و تولید مستندات قابل فهم طراحی شده‌اند. مهم‌ترین امکانات این کتابخانه عبارت‌اند از:

۲-۶-۱) استخراج کلاس‌ها

یکی از قابلیت‌های اصلی، استخراج تمامی کلاس‌های تعریف‌شده در یک هستی‌شناسی است. این کتابخانه با تحلیل گراف RDF و شناسایی موجودیت‌هایی که به‌عنوان owl:Class یا rdfs:Class تعریف شده‌اند، ساختار مفهومی دامنه را استخراج می‌کند. علاوه بر شناسایی کلاس‌ها، اطلاعاتی نظیر نام کلاس، شناسه(IRI) ، توضیحات، برچسب‌ها و روابط آن با سایر کلاس‌ها نیز استخراج می‌شود. این قابلیت امکان تحلیل ساختار مفهومی و بررسی سازمان‌دهی دانش را برای پژوهشگران فراهم می‌کند.

۲-۶-۲) استخراج ویژگی‌های شیء و داده

آنتواسپای قادر است تمامی ویژگی‌های تعریف‌شده در هستی‌شناسی را استخراج و دسته‌بندی کند. ویژگی‌های شیء روابط میان دو موجودیت را توصیف می‌کنند، در حالی که ویژگی‌های داده‌ای برای انتساب مقادیر داده‌ای مانند متن، عدد یا تاریخ به موجودیت‌ها مورد استفاده قرار می‌گیرند.

در فرآیند استخراج، کتابخانه اطلاعاتی مانند دامنه، برد، ویژگی‌های معکوس، ویژگی‌های تابعی و سایر قیود منطقی مرتبط با هر Property را نیز بازیابی می‌کند. این اطلاعات برای تحلیل مدل مفهومی و ارزیابی کیفیت طراحی هستی‌شناسی اهمیت زیادی دارند.

۲-۶-۳) بازسازی و نمایش سلسله‌مراتب کلاس‌ها

یکی از قابلیت‌های مهم آنتواسپای، استخراج روابط سلسله‌مراتبی میان کلاس‌ها است. این کتابخانه با تحلیل روابط rdfs:subClassOf، ساختار درختی کلاس‌ها را بازسازی کرده و ارتباط میان کلاس‌های والد و فرزند را مشخص می‌کند.

بازسازی سلسله‌مراتب کلاس‌ها به کاربران کمک می‌کند تا نحوه سازمان‌دهی مفاهیم، میزان انتزاع هر کلاس و وابستگی‌های موجود در مدل دانشی را به‌خوبی درک کنند. این قابلیت به‌ویژه در تحلیل هستی‌شناسی‌های بزرگ که دارای صدها یا هزاران کلاس هستند، اهمیت ویژه‌ای دارد.

۲-۶-۴) استخراج نمونه‌ها

آنتواسپای علاوه بر استخراج ساختار مفهومی، نمونه‌های واقعی هر کلاس را نیز شناسایی می‌کند. نمونه‌ها یا Individualsنمایش‌دهنده موجودیت‌های واقعی موجود در دامنه دانش هستند که به کلاس‌های مختلف تعلق دارند.

این کتابخانه ارتباط هر نمونه با کلاس مربوطه، ویژگی‌های تعریف‌شده برای آن و روابط میان نمونه‌ها را استخراج می‌کند. وجود این قابلیت امکان تحلیل داده‌های موجود در گراف دانش و بررسی نحوه استفاده عملی از مفاهیم تعریف‌شده در هستی‌شناسی را فراهم می‌کند.

۲-۶-۵) استخراج فراداده

یکی دیگر از امکانات مهم آنتواسپای، استخراج فراداده‌های موجود در هستی‌شناسی است. این فراداده‌ها شامل اطلاعاتی مانند:

- عنوان
- برچسب‌ها
- توضیحات
- نسخه هستی‌شناسی
- نویسنده یا ایجادکننده
- تاریخ ایجاد یا ویرایش
- فضای نام
- Importهای تعریف‌شده
- Annotation Propertyها

استخراج این اطلاعات نقش مهمی در مستندسازی، نگهداری و مدیریت نسخه‌های مختلف هستی‌شناسی ایفا می‌کند.

۲-۶-۶) تولید مستندات خودکار

یکی از برجسته‌ترین قابلیت‌های آنتواسپای، تولید مستندات خودکار از ساختار هستی‌شناسی است. این کتابخانه پس از تحلیل فایل‌های RDF یاOWL ، اطلاعات استخراج‌شده را در قالب صفحات HTML سازمان‌دهی می‌کند.

این مستندات معمولاً شامل معرفی کلاس‌ها، ویژگی‌ها، نمونه‌ها، سلسله‌مراتب مفاهیم، مشخصات هرProperty ، روابط میان موجودیت‌ها و سایر اطلاعات ساختاری هستند. مستندات تولیدشده بدون نیاز به بررسی مستقیم فایل‌های RDF/XML یاTurtle ، نمایی خوانا و قابل فهم از هستی‌شناسی ارائه می‌کنند و برای مستندسازی پروژه‌های بزرگ بسیار کاربردی هستند.

۲-۶-۷) نمایش روابط و گراف‌های معنایی

آنتواسپای قابلیت نمایش ساختار گرافی هستی‌شناسی را نیز فراهم می‌کند. این کتابخانه با استخراج روابط موجود میان کلاس‌ها، ویژگی‌ها و نمونه‌ها، امکان تولید نمودارهای مفهومی و گراف‌های ارتباطی را فراهم می‌سازد.

نمایش گرافی، درک ساختار کلی هستی‌شناسی، مسیرهای ارتباطی میان مفاهیم و میزان وابستگی عناصر مختلف را آسان‌تر می‌کند. این قابلیت به‌ویژه در تحلیل معماری دانش و شناسایی الگوهای ارتباطی موجود در گراف‌های بزرگ کاربرد فراوانی دارد.

۲-۶-۸) جستجو و پیمایش در هستی‌شناسی[۴۷]

آنتواسپای امکانات متعددی برای پیمایش ساختار هستی‌شناسی در اختیار کاربران قرار می‌دهد. توسعه‌دهندگان می‌توانند از طریق رابط برنامه‌نویسی(API) ، کلاس‌ها، ویژگی‌ها، نمونه‌ها و روابط مختلف را جستجو کرده و اطلاعات مورد نیاز خود را استخراج کنند.

این قابلیت امکان دسترسی برنامه‌نویسی به تمام عناصر موجود در گراف دانش را فراهم کرده و توسعه ابزارهای تحلیل، سامانه‌های مدیریت دانش، موتورهای جستجوی معنایی و نرم‌افزارهای مبتنی بر وب معنایی را تسهیل می‌کند.

۲-۶-۹) قابلیت توسعه و یکپارچه‌سازی

به دلیل پیاده‌سازی آنتواسپای بر پایه زبان Python و استفاده از کتابخانهRDFLib ، این ابزار قابلیت یکپارچه‌سازی با سایر کتابخانه‌های پردازش داده‌های معنایی، سامانه‌های مدیریت دانش و پروژه‌های مبتنی بر هوش مصنوعی را دارد. توسعه‌دهندگان می‌توانند با استفاده ازAPI های موجود، قابلیت‌های جدیدی به کتابخانه اضافه کرده یا از آن به‌عنوان بخشی از سامانه‌های بزرگ‌تر استفاده کنند.

[۴۷] Ontology Navigation

در مجموع، قابلیت‌های آنتواسپای فراتر از یک ابزار نمایش اطلاعات است. این کتابخانه با فراهم کردن امکاناتی برای استخراج ساختار، تحلیل روابط، مستندسازی خودکار، تولید نمایش‌های گرافی و دسترسی برنامه‌نویسی به اجزای هستی‌شناسی، به یکی از ابزارهای مهم در حوزه مهندسی دانش، وب معنایی و تحلیل گراف‌های دانش تبدیل شده است.

جدول ۲-۱: قابلیت‌های کاربردی آنتواسپای

ردیف	قابلیت	توضیحات فنی	کاربرد
۱	بارگذاری هستی‌شناسی[۴۸]	خواندن و تجزیه فایل‌هایRDF ، RDFS، OWL، Turtle، N-Triplesو سایر قالب‌های استاندارد با استفاده ازRDFLib	آماده‌سازی داده‌ها برای تحلیل
۲	استخراج کلاس‌ها	شناسایی کلاس‌های تعریف‌شده (owl:Class و rdfs:Class)و استخراج ویژگی‌های مرتبط با آن‌ها	تحلیل ساختار مفهومی دامنه
۳	استخراج ویژگی‌های شیء	شناسایی روابط میان کلاس‌ها و موجودیت‌ها، همراه باDomain ، Rangeو Inverse Property	تحلیل روابط معنایی
۴	استخراج ویژگی‌های داده‌ای	استخراج ویژگی‌هایی که مقادیر داده‌ای مانند متن، عدد و تاریخ را به موجودیت‌ها اختصاص می‌دهند	بررسی ویژگی‌های توصیفی موجودیت‌ها
۵	استخراج نمونه‌ها	شناسایی نمونه‌های هر کلاس و روابط آن‌ها با سایر عناصر هستی‌شناسی	تحلیل داده‌های نمونه
۶	بازسازی سلسله‌مراتب کلاس‌ها	استخراج روابط rdfs:subClassOf و ایجاد ساختار درختی کلاس‌ها	نمایش سازمان مفهومی دانش
۷	تحلیل سلسله‌مراتب ویژگی‌ها	استخراج روابط rdfs:subPropertyOf و نمایش وابستگی میان ویژگی‌ها	تحلیل ساختار ویژگی‌ها
۸	استخراج فراداده	استخراجLabel ، Comment، Namespace، Version، Importsو Annotation ها	مستندسازی و مدیریت هستی‌شناسی
۹	استخراج قیود منطقی[۴۹]	تحلیلRestriction ها، Cardinality، Domain، Rangeو سایر قیودOWL	ارزیابی مدل منطقی هستی‌شناسی

[۴۸] Ontology Loading
[۴۹] Restrictions

۱۰	تولید مستندات HTML	ایجاد صفحات HTML شامل کلاس‌ها، ویژگی‌ها، روابط و نمودارهای ساختاری	مستندسازی خودکار پروژه
۱۱	نمایش ساختار گرافی	ایجاد نمایش گرافی از روابط میان کلاس‌ها و ویژگی‌ها	درک بهتر ساختار دانش
۱۲	رابط برنامه‌نویسی (Python API)	دسترسی مستقیم به عناصر هستی‌شناسی از طریق کلاس‌ها و متدهای کتابخانه	توسعه نرم‌افزارهای مبتنی بر وب معنایی
۱۳	پیمایش گراف دانش	جستجو و پیمایش کلاس‌ها، ویژگی‌ها، نمونه‌ها و روابط	تحلیل و پردازش داده‌های معنایی
۱۴	پشتیبانی از استانداردهای W۳ C	سازگاری باRDF ، RDFS، OWLو استانداردهای مرتبط	افزایش قابلیت همکاری میان سامانه‌ها
۱۵	یکپارچه‌سازی با اکوسیستم Python	امکان استفاده در کنار RDFLib، Graphviz، Jupyter Notebook و سایر ابزارهای تحلیل داده	توسعه سامانه‌های هوشمند و پژوهشی

فصل سوم
نصب و راه‌اندازی

پس از معرفی مفاهیم پایه وب معنایی، هستی‌شناسی و بررسی ساختار و قابلیت‌های کتابخانه آنتواسپای در فصل‌های پیشین، در این فصل جنبه عملی استفاده از این کتابخانه مورد بررسی قرار می‌گیرد. آشنایی با قابلیت‌های نظری یک ابزار، اگرچه برای درک عملکرد آن ضروری است، اما ارزیابی کارایی و نحوه استفاده از آن در یک محیط واقعی، مستلزم پیاده‌سازی و اجرای عملی آن است. ازاین‌رو، در این فصل فرآیند نصب، راه‌اندازی و استفاده از این کتابخانه به‌صورت گام‌به‌گام تشریح می‌شود.

در ابتدا، پیش‌نیازهای لازم برای اجرای کتابخانه معرفی شده و مراحل نصب و پیکربندی محیط توسعه ارائه می‌شود. سپس نحوه نصب آنتواسپای و بررسی صحت عملکرد آن در محیط پایتون توضیح داده خواهد شد. آماده‌سازی صحیح محیط اجرا، پیش‌نیاز استفاده از قابلیت‌های مختلف این کتابخانه برای تحلیل و مستندسازی هستی‌شناسی‌ها است.

پس از راه‌اندازی محیط، یک مطالعه موردی[۵۰] مبتنی بر یک هستی‌شناسی نمونه ارائه می‌شود. در این مطالعه، قابلیت‌های اصلی کتابخانه از جمله بارگذاری فایل هستی‌شناسی، استخراج کلاس‌ها، ویژگی‌های شیء و داده، نمونه‌ها، روابط معنایی، سلسله‌مراتب مفاهیم و تولید مستندات خودکار مورد ارزیابی قرار می‌گیرند. همچنین خروجی‌های تولیدشده توسط آنتواسپای تحلیل شده و میزان کارایی کتابخانه در استخراج و نمایش ساختار دانش بررسی می‌شود.

[۵۰] Case Study

هدف اصلی این فصل، ارائه یک ارزیابی عملی از عملکرد کتابخانه OntoSpy و نشان دادن نحوه به‌کارگیری آن در تحلیل هستی‌شناسی‌های مبتنی بر استانداردهای RDF و OWL است. نتایج حاصل از این بررسی، دیدگاه مناسبی نسبت به قابلیت‌ها، مزایا و محدودیت‌های این ابزار در اختیار پژوهشگران و توسعه‌دهندگان سامانه‌های مبتنی بر وب معنایی قرار خواهد داد و زمینه را برای جمع‌بندی نهایی و ارزیابی کلی کتابخانه در فصل پایانی فراهم می‌کند.

۳-۱) پیش‌نیاز های نصب و راه‌اندازی

پیش از نصب و استفاده از کتابخانه آنتواسپای، لازم است محیط اجرای مناسبی برای پردازش داده‌های معنایی و اجرای برنامه‌های مبتنی بر زبان Python فراهم شود. از آنجا که آنتواسپای به‌عنوان یک کتابخانه پایتون توسعه یافته است، عملکرد صحیح آن به نصب و پیکربندی برخی ابزارها و مؤلفه‌های نرم‌افزاری وابسته است. در این بخش، پیش‌نیازهای لازم برای نصب و راه‌اندازی این کتابخانه معرفی می‌شوند.

۳-۱-۱) نصب زبان برنامه‌نویسی *Python*

آنتواسپای بر پایه زبان برنامه‌نویسی Python توسعه یافته است؛ بنابراین، اولین پیش‌نیاز استفاده از این کتابخانه، نصب Python بر روی سیستم است. پیشنهاد می‌شود از نسخه‌های ۳.۸ یا بالاتر استفاده شود، زیرا نسخه‌های جدیدتر علاوه بر بهبود عملکرد، از بسیاری از کتابخانه‌های وابسته نیز به‌طور کامل پشتیبانی می‌کنند.

پس از نصبPython ، می‌توان با اجرای دستور زیر در خط فرمان از نصب صحیح آن اطمینان حاصل کرد:

```
python --version
```

در برخی سیستم‌عامل‌ها، به‌ویژه توزیع‌های لینوکس و نسخه‌های جدیدmacOS ، ممکن است لازم باشد از دستور زیر استفاده شود:

```
python3 --version
```

در صورت نصب صحیح، شماره نسخه Python در محیط خط فرمان نمایش داده خواهد شد.

۳-۱-۲) ابزار مدیریت بسته *pip*

کتابخانه آنتواسپای از طریقpip ، مدیر بسته استانداردPython ، نصب می‌شود. این ابزار امکان دانلود، نصب، به‌روزرسانی و مدیریت کتابخانه‌های موردنیاز را فراهم می‌کند و معمولاً همراه با Python نصب می‌شود.

برای بررسی نصب صحیح pip می‌توان دستور زیر را اجرا کرد:

```
pip --version
```

یا در برخی سیستم‌ها:

```
pip3 --version
```

نمایش نسخه pip نشان‌دهنده آماده بودن محیط برای نصب کتابخانه‌های پایتون است.

۳-۱-۳) استفاده از محیط مجازی[۵۱]

اگرچه استفاده از محیط مجازی الزامی نیست، اما برای پروژه‌های پژوهشی و توسعه نرم‌افزار به‌شدت توصیه می‌شود. محیط مجازی امکان ایجاد یک فضای مستقل برای هر پروژه را فراهم می‌کند؛ به‌گونه‌ای که کتابخانه‌های نصب‌شده در یک پروژه، بر سایر پروژه‌های موجود در سیستم تأثیر نگذارند.

مزایای استفاده از محیط مجازی عبارت‌اند از:

- جلوگیری از تداخل نسخه کتابخانه‌ها
- مدیریت آسان وابستگی‌های پروژه
- افزایش قابلیت بازتولید[۵۲] آزمایش‌ها
- سهولت انتقال پروژه به سایر سیستم‌ها

برای ایجاد یک محیط مجازی می‌توان از دستور زیر استفاده کرد:

```
python -m venv ontospy_env
```

پس از ایجاد محیط، لازم است آن فعال شود. در سیستم‌عامل‌های مختلف، نحوه فعال‌سازی متفاوت است. به طور مثال در ویندوز:

```
ontospy_env\Scripts\activate
```

در Linux وmacOS :

```
source ontospy_env/bin/activate
```

[۵۱] Virtual Environment
[۵۲] Reproducibility

۳-۱-۴) محیط توسعه[۵۳]

اگرچه اجرای آنتواسپای از طریق خط فرمان امکان‌پذیر است، استفاده از یک محیط توسعه یکپارچه (IDE) فرآیند برنامه‌نویسی، اشکال‌زدایی و مدیریت پروژه را ساده‌تر می‌کند. از جمله محیط‌های توسعه مناسب برای کار با آنتواسپای می‌توان به موارد زیر اشاره کرد:

- Visual Studio Code (VS Code)
- PyCharm
- Jupyter Notebook
- Spyder

در این پژوهش، از Visual Studio Code به‌عنوان محیط توسعه استفاده شده است؛ زیرا امکاناتی مانند مدیریت محیط‌های مجازی، تکمیل خودکار کد، اشکال‌زدایی و یکپارچگی مناسب با Python را فراهم می‌کند.

۳-۱-۵) فایل هستی‌شناسی

برای بررسی عملکرد آنتواسپای، وجود حداقل یک فایل هستی‌شناسی در یکی از قالب‌های استاندارد وب معنایی ضروری است. این کتابخانه از قالب‌هایی مانندRDF/XML ، OWL، Turtle (TTL)، N-Triplesو JSON-LD پشتیبانی می‌کند. فایل هستی‌شناسی می‌تواند توسط ابزارهایی مانند Protégé ایجاد شده یا از مخازن عمومی هستی‌شناسی تهیه شود.

در این پژوهش، از یک فایل هستی‌شناسی نمونه در قالب OWL برای بررسی قابلیت‌های مختلف کتابخانه آنتواسپای استفاده خواهد شد.

۳-۱-۶) اتصال اینترنت

در مرحله نصب اولیه، اتصال به اینترنت برای دریافت بسته آنتواسپای و وابستگی‌های آن از مخزن PyPI موردنیاز است. پس از نصب، بیشتر قابلیت‌های کتابخانه به‌صورت محلی قابل استفاده هستند و برای تحلیل فایل‌های هستی‌شناسی نیازی به اتصال دائمی به اینترنت وجود ندارد.

در مجموع، فراهم کردن پیش‌نیازهای فوق موجب می‌شود محیط اجرای مناسبی برای نصب و استفاده از آنتواسپای ایجاد شود. در بخش بعد، مراحل نصب کتابخانه و بررسی صحت عملکرد آن به‌صورت عملی تشریح خواهد شد.

[۵۳] IDE

۳-۲) نصب کتابخانه آنتواسپای

پس از آماده‌سازی محیط توسعه و نصب پیش‌نیازهای لازم، مرحله بعد نصب کتابخانه آنتواسپای است. این کتابخانه از طریق مدیر بسته pip ، که ابزار استاندارد مدیریت بسته‌های Python محسوب می‌شود، قابل نصب است. در فرآیند نصب، علاوه بر خود کتابخانه، مجموعه‌ای از وابستگی‌های موردنیاز نیز به‌صورت خودکار دریافت و نصب می‌شوند.

برای نصب آخرین نسخه پایدار آنتواسپای، کافی است دستور زیر در محیط خط فرمان یا ترمینال اجرا شود:

```
pip install ontospy
```

در صورتی که از سیستم‌هایی استفاده شود که Python ۳ با دستور python ۳ فراخوانی می‌شود، ممکن است لازم باشد از دستور زیر استفاده شود:

```
pip3 install ontospy
```

پس از اجرای دستور فوق، مدیر بسته pip ابتدا مخزن Python Package Index (PyPI) را بررسی کرده و آخرین نسخه سازگار از آنتواسپای را دریافت می‌کند. سپس وابستگی‌های موردنیاز کتابخانه، در صورت نصب نبودن، به‌طور خودکار دانلود و نصب می‌شوند. برخی از مهم‌ترین این وابستگی‌ها عبارت‌اند از:

- RDFLib برای پردازش و مدیریت گراف‌های RDF
- Jinja۲ برای تولید صفحات HTML و مستندات
- Click برای پیاده‌سازی رابط خط فرمان[۵۴]
- سایر کتابخانه‌های کمکی موردنیاز برای تحلیل و مستندسازی هستی‌شناسی‌ها

در طول فرآیند نصب، pip نسخه هر یک از بسته‌ها را بررسی کرده و در صورت وجود نسخه‌های ناسازگار، آن‌ها را به‌روزرسانی یا جایگزین می‌کند. در پایان، در صورت نصب موفق، پیامی مشابه عبارت زیر در ترمینال نمایش داده می‌شود:

```
Successfully installed ontospy
```

پس از اتمام نصب، لازم است صحت عملکرد کتابخانه بررسی شود. یکی از ساده‌ترین روش‌ها، مشاهده اطلاعات نسخه نصب‌شده است. برای این منظور می‌توان از دستور زیر استفاده کرد:

[۵۴] CLI

```
pip show ontospy
```

اجرای این دستور اطلاعاتی مانند نسخه نصب‌شده، محل نصب، وابستگی‌ها و مشخصات بسته را نمایش می‌دهد. همچنین می‌توان از محیط مفسر Python نیز برای اطمینان از نصب صحیح استفاده کرد:

```
import ontospy
print(ontospy.__version__)
```

در صورتی که کتابخانه بدون ایجاد خطا بارگذاری شود و شماره نسخه آن نمایش داده شود، می‌توان نتیجه گرفت که نصب با موفقیت انجام شده است.

پس از نصب و اطمینان از عملکرد صحیح کتابخانه، محیط آماده استفاده از قابلیت‌های آنتواسپای خواهد بود. در بخش بعد، با معرفی یک مطالعه موردی و بارگذاری یک فایل هستی‌شناسی نمونه، نحوه استفاده عملی از این کتابخانه و قابلیت‌های آن مورد بررسی قرار خواهد گرفت.

۳-۳) معرفی سناریوی عملی و فایل هستی‌شناسی

به‌منظور ارزیابی عملی قابلیت‌های کتابخانه آنتواسپای، در این پژوهش یک مطالعه موردی[۵۵] مبتنی بر یک هستی‌شناسی در حوزه مدیریت دانشگاه در نظر گرفته شده است. انتخاب این سناریو به دلیل برخورداری از ساختاری نسبتاً کامل، وجود مفاهیم متنوع و روابط متعدد میان موجودیت‌ها است که امکان بررسی بخش عمده‌ای از قابلیت‌های کتابخانه آنتواسپای را فراهم می‌کند.

در این سناریو، ساختار یک سامانه مدیریت دانشگاه با استفاده از زبان OWL مدل‌سازی شده است. این هستی‌شناسی، مفاهیم اصلی موجود در یک محیط دانشگاهی و ارتباط میان آن‌ها را به‌صورت رسمی نمایش می‌دهد. هدف از طراحی این مدل، ایجاد بستری مناسب برای بررسی نحوه استخراج و تحلیل اطلاعات توسط آنتواسپای است.

مهم‌ترین کلاس‌های تعریف‌شده در این هستی‌شناسی عبارت‌اند از:

- Student
- Professor
- Course
- Department

[۵۵] Case Study

- Faculty
- Classroom

هر یک از این کلاس‌ها دارای مجموعه‌ای از ویژگی‌ها و روابط معنایی هستند که تعامل میان موجودیت‌های مختلف سیستم را توصیف می‌کنند. برای مثال، یک دانشجو می‌تواند در چند درس ثبت‌نام کند، هر درس توسط یک استاد ارائه شود و هر استاد نیز به یک گروه آموزشی مشخص تعلق داشته باشد.

به‌منظور مدل‌سازی این روابط، مجموعه‌ای از Object Property ها در هستی‌شناسی تعریف شده‌اند که از جمله مهم‌ترین آن‌ها می‌توان به موارد زیر اشاره کرد:

- enrollsIn : ارتباط بین دانشجو و درس
- teaches : ارتباط بین استاد و درس
- belongsToDepartment : ارتباط استاد با گروه آموزشی
- offeredBy : ارتباط درس با گروه آموزشی
- locatedIn : ارتباط کلاس درس با دانشکده

علاوه بر روابط فوق، ویژگی‌های داده‌ای[۵۶] نیز برای ذخیره اطلاعات توصیفی موجودیت‌ها در نظر گرفته شده‌اند. برخی از این ویژگی‌ها عبارت‌اند از:

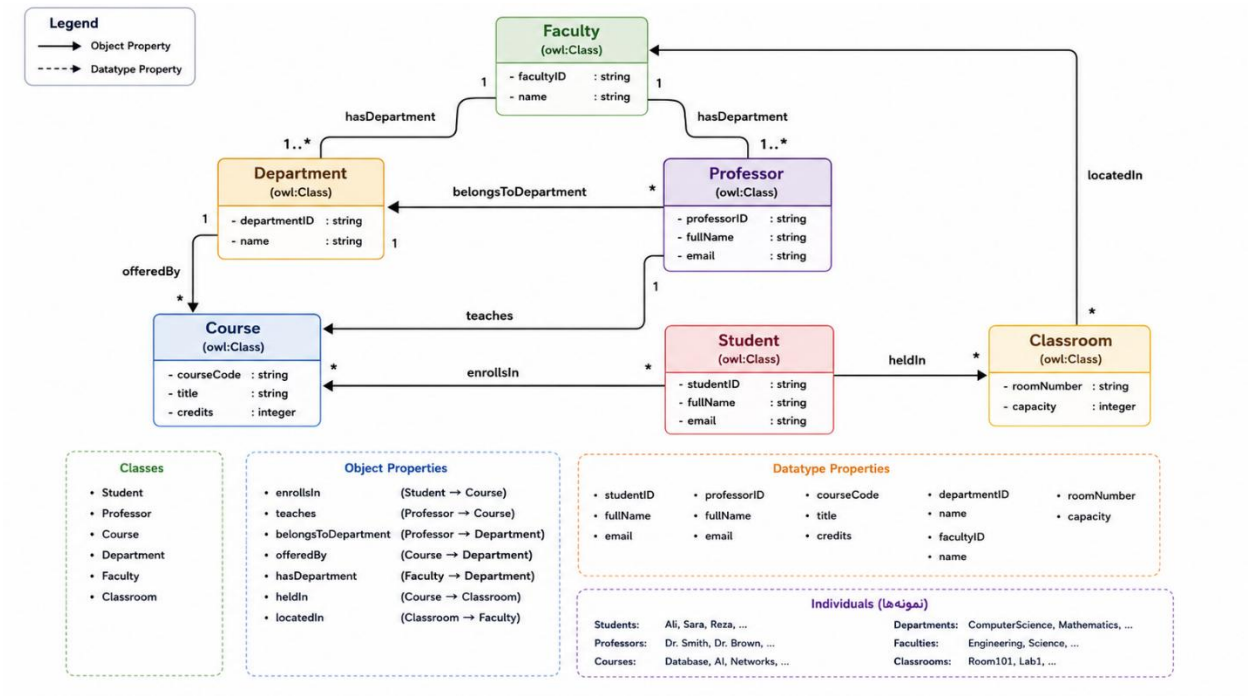
- studentID
- professorID
- courseCode
- credits
- fullName
- email

به‌منظور شبیه‌سازی یک محیط واقعی، برای هر یک از کلاس‌ها چند نمونه نیز تعریف شده است. برای مثال، نمونه‌هایی از دانشجویان، اساتید، دروس و گروه‌های آموزشی در فایل هستی‌شناسی ایجاد شده‌اند تا امکان بررسی قابلیت‌های استخراج نمونه‌ها و روابط میان آن‌ها فراهم شود.

[۵۶] Datatype Property

فایل مورد استفاده در این پژوهش با نام University.owl ذخیره شده و شامل تمامی کلاس‌ها، ویژگی‌ها، روابط و نمونه‌های موردنیاز برای ارزیابی عملکرد کتابخانه آنتواسپای است. این فایل در پوشه پروژه قرار گرفته و در ادامه فصل، به‌عنوان ورودی کتابخانه مورد استفاده قرار خواهد گرفت.

نمودار ۳-۱: نمودار هستی‌شناسی



۳-۳-۱) خروجی بارگذاری فایل هستی‌شناسی

پس از اجرای برنامه و بارگذاری فایل University.owl ، کتابخانه آنتواسپای فایل را با موفقیت پردازش کرده و اطلاعات کلی مربوط به هستی‌شناسی را استخراج می‌کند. این اطلاعات شامل نام هستی‌شناسی، تعداد کلاس‌ها، ویژگی‌ها، نمونه‌ها و فضای نام است.

نمونه‌ای از خروجی اجرای برنامه به‌صورت زیر است:

```
Ontology successfully loaded.
Ontology Name : University
Format : OWL/RDF
Classes : 6
Object Properties : 5
Datatype Properties : 6
Individuals : 18
Namespaces : 3
```

همچنین با چاپ شیء مربوط به هستی‌شناسی، اطلاعات پایه آن نیز قابل مشاهده است:

```
<Ontology: University.owl>
URI: http://www.example.org/university#
Classes: 6
Properties: 11
Individuals: 18
```

همان‌گونه که مشاهده می‌شود، فایل هستی‌شناسی بدون هیچ‌گونه خطایی توسط آنتواسپای بارگذاری شده است. کتابخانه پس از تجزیه فایلOWL ، گراف RDF مربوط به هستی‌شناسی را ایجاد کرده و تمام عناصر موجود در آن را برای انجام تحلیل‌های بعدی آماده می‌کند. در این مرحله، اطلاعات ساختاری هنوز به‌صورت جزئی نمایش داده نمی‌شوند، اما تمامی کلاس‌ها، ویژگی‌ها، نمونه‌ها و روابط موجود در حافظه بارگذاری شده‌اند و از طریق رابط برنامه‌نویسی (API) کتابخانه قابل دسترسی هستند.

۳-۳-۲) دستورات و قابلیت‌های اصلی کتابخانه آنتواسپای

کتابخانه آنتواسپای علاوه بر ارائه رابط برنامه‌نویسی(API) ، یک رابط خط فرمان[۵۷] نیز در اختیار کاربران قرار می‌دهد. این رابط امکان بارگذاری، تحلیل، مستندسازی و مدیریت هستی‌شناسی‌ها را بدون نیاز به توسعه برنامه‌های پایتونی فراهم می‌کند. مهم‌ترین دستورات این کتابخانه در جدول زیر معرفی شده‌اند.

جدول ۳-۱: دستورات کاربردی کتابخانه آنتواسپای

دستور	کاربرد
ontospy --help	نمایش راهنما و فهرست دستورات موجود
ontospy scan	اسکن و تحلیل یک فایل یا مجموعه‌ای از فایل‌های هستی‌شناسی
ontospy gendocs	تولید مستندات HTML از یک هستی‌شناسی
ontospy lib	مدیریت کتابخانه محلی آنتواسپای
ontospy ser	تبدیل قالب RDF
ontospy shell	ورود به محیط تعاملی آنتواسپای برای بررسی عناصر هستی‌شناسی
ontospy utils	ابزار کمکی آنتواسپای

برای مشاهده تمامی دستورات و گزینه‌های موجود در آنتواسپای، از دستور زیر استفاده می‌شود:

[۵۷] Command Line Interface - CLI

```
ontospy --help
```

این دستور فهرست کاملی از فرمان‌ها، پارامترها و گزینه‌های قابل استفاده را نمایش می‌دهد.

برای تحلیل یک فایل OWL یاRDF ، می‌توان از دستور زیر استفاده کرد:

```
ontospy scan University.owl
```

با اجرای این دستور، کتابخانه ساختار فایل را بررسی کرده و اطلاعاتی درباره کلاس‌ها، ویژگی‌ها، روابط و سایر اجزای هستی‌شناسی استخراج می‌کند.

یکی از مهم‌ترین قابلیت‌های آنتواسپای، تولید مستندات HTML است که با دستور زیر انجام می‌شود:

```
ontospy gendocs University.owl
```

در نتیجه اجرای این دستور، مجموعه‌ای از صفحات HTML ایجاد می‌شود که شامل اطلاعات ساختاری هستی‌شناسی، سلسله‌مراتب کلاس‌ها، ویژگی‌ها، روابط و نمونه‌ها است.

برای مدیریت کتابخانه محلی هستی‌شناسی ها می‌توان از دستور زیر استفاده کرد:

```
ontospy lib University.owl
```

نمونه های کاربردی به صورت زیر می‌باشد:

```
 ontospy lib --show
ontospy lib --show -i
ontospy lib --bootstrap
ontospy lib --cache
ontospy lib --reveal
ontospy lib --directory C:\OntospyLibrary
```

ذخیره فایل محلی یا آنلاین در کتابخانه با دستور save–، نمایش آنتولوژی های ذخیره شده با دستور show, –s-، افزودن آنتولوژی های معروف به کتابخانه با دستور bootstrap–، پاک‌سازی حافظه پنهان کتابخانه با دستور cache–، تغییر پوشه کتابخانه در سیستم عامل با دستور reveal–، نمایش نمونه ها هنگام بازبینی با دستور -i, --individuals، نمایش داده‌های ضمنی هنگام بازبینی با دستورات -x, --extra می‌باشد.

با استفاده از دستو ser می‌توان تولید خروجی Turtle، XML، N-Triples یا JSON-LD را انجام داد. به نمونه‌های زیر دقت کنید:

```
ontospy ser filename -f turtle
ontospy ser filename -f xml
ontospy ser filename -f pretty-xml
ontospy ser filename -f nt
ontospy ser filename -f json-ld
```

دستورات -f, --output_format جهت تعیین قالب خروجی و دستورات -v, --verbose جهت نمایش جزئیات اجرا مورد استفاده قرار می‌گیرد.

برای بررسی و پیمایش عناصر هستی‌شناسی به‌صورت تعاملی، می‌توان از دستور زیر استفاده کرد:

```
ontospy shell
```

این محیط امکان اجرای دستورات مختلف و دسترسی مستقیم به کلاس‌ها، ویژگی‌ها و نمونه‌های موجود در هستی‌شناسی را فراهم می‌کند.

آخرین دستوری که می‌خواهیم به شما معرفی کنیم دستو utils می‌باشد. از این دستور میتوان جهت جست و جوی هستی‌شناسی ها در مخزن آنلاین و آزمون فایل JSON-LD مانند زیر استفاده کرد:

```
ontospy utils –discover
ontospy utils --jsonld file.jsonld
```

جهت پیدا کردن آنتولوژی آنلاین از دستورات --discover, -d و جهت بررسی فایل JSON-LD با محیط آزمون آنلاین از دستورات --jsonld, -j می‌توان استفاده کرد.

به‌کارگیری این دستورات، فرآیند تحلیل و مستندسازی هستی‌شناسی‌ها را تا حد زیادی ساده کرده و کاربران را قادر می‌سازد بدون نیاز به توسعه برنامه‌های پیچیده، اطلاعات مورد نیاز خود را از فایل‌های RDF و OWL استخراج کنند.

فصل چهارم

مزایا، محدودیت‌ها و مقایسه کتابخانه OntoSpy با ابزارهای مشابه

پس از بررسی ساختار، قابلیت‌ها و اجرای عملی کتابخانه آنتواسپای، ارزیابی نقاط قوت و محدودیت‌های آن نقش مهمی در شناخت میزان کارآمدی این ابزار دارد. هرچند آنتواسپای یکی از کتابخانه‌های شناخته‌شده در حوزه تحلیل و مستندسازی هستی‌شناسی است، اما مانند سایر ابزارهای نرم‌افزاری، دارای مزایا و محدودیت‌هایی است که انتخاب آن را برای کاربردهای مختلف تحت تأثیر قرار می‌دهد. در این بخش، ابتدا مهم‌ترین مزایا و محدودیت‌های آنتواسپای بررسی شده و سپس این کتابخانه با چند ابزار رایج در حوزه مهندسی هستی‌شناسی مقایسه می‌شود.

۴-۱) مزایای کتابخانه آنتواسپای

یکی از مهم‌ترین مزایای آنتواسپای، سهولت استفاده از آن است. نصب کتابخانه از طریق مدیر بسته pip انجام می‌شود و رابط برنامه‌نویسی (API) آن به‌گونه‌ای طراحی شده است که توسعه‌دهندگان می‌توانند با چند خط کد، فایل‌های هستی‌شناسی را بارگذاری کرده و اطلاعات ساختاری آن‌ها را استخراج کنند. این ویژگی باعث می‌شود که زمان موردنیاز برای توسعه ابزارهای تحلیل هستی‌شناسی به میزان قابل‌توجهی کاهش یابد.

مزیت دیگر آنتواسپای، پشتیبانی کامل از استانداردهای وب معنایی است. این کتابخانه قادر به پردازش فایل‌هایRDF ، RDFSو OWL بوده و با بهره‌گیری ازRDFLib ، اطلاعات موجود در گراف‌های معنایی را

با دقت مناسبی استخراج می‌کند. این ویژگی امکان استفاده از آنتواسپای را در طیف گسترده‌ای از پروژه‌های مبتنی بر وب معنایی فراهم می‌سازد.

یکی از قابلیت‌های برجسته این کتابخانه، تولید خودکار مستندات HTML است. این قابلیت به پژوهشگران و توسعه‌دهندگان اجازه می‌دهد بدون نیاز به بررسی مستقیم فایل‌های RDF/XML یاOWL ، ساختار مفهومی، کلاس‌ها، ویژگی‌ها و روابط موجود در هستی‌شناسی را در قالب مستنداتی خوانا و ساختاریافته مشاهده کنند. این ویژگی در مستندسازی پروژه‌های بزرگ و نگهداری آن‌ها اهمیت ویژه‌ای دارد.

از دیگر مزایای آنتواسپای می‌توان به متن‌باز بودن، قابلیت توسعه، یکپارچه‌سازی آسان با سایر کتابخانه‌های اکوسیستم Python و مناسب بودن آن برای تحلیل خودکار هستی‌شناسی‌ها اشاره کرد. وجود API سطح بالا باعث شده است که این کتابخانه به‌راحتی در پروژه‌های پژوهشی، سامانه‌های مدیریت دانش و ابزارهای پردازش گراف‌های دانش مورد استفاده قرار گیرد.

۴-۲) محدودیت‌های کتابخانه آنتواسپای

با وجود قابلیت‌های متعدد، آنتواسپای دارای محدودیت‌هایی نیز است که باید در انتخاب آن برای پروژه‌های مختلف مورد توجه قرار گیرد.

یکی از مهم‌ترین محدودیت‌های این کتابخانه، نبود قابلیت ویرایش هستی‌شناسی است .آنتواسپای عمدتاً برای تحلیل، پیمایش و مستندسازی طراحی شده و امکاناتی برای ایجاد یا اصلاح ساختار آنتواسپای در اختیار کاربر قرار نمی‌دهد. بنابراین، برای طراحی یا ویرایش هستی‌شناسی معمولاً از ابزارهایی مانند Protégé استفاده می‌شود.

محدودیت دیگر، عدم پشتیبانی از استنتاج منطقی[۵۸] است. این کتابخانه قادر به اجرای موتورهای استنتاج مانند HermiTیا Pellet نیست و در نتیجه، کلاس‌های استنتاج‌شده، ناسازگاری‌های منطقی و نتایج حاصل از قواعد استنتاج را محاسبه نمی‌کند. در کاربردهایی که تحلیل معنایی پیشرفته موردنیاز است، باید از ابزارهای مکمل استفاده شود.

همچنین، قابلیت‌های تجسم در آنتواسپای نسبت به برخی ابزارهای تخصصی محدودتر است. اگرچه امکان تولید مستندات و نمایش ساختار سلسله‌مراتبی وجود دارد، اما امکانات تعاملی برای کاوش گراف‌های بزرگ یا ویرایش بصری روابط ارائه نمی‌شود.

[۵۸] Reasoning

از سوی دیگر، عملکرد آنتواسپای در پردازش هستی‌شناسی‌های بسیار بزرگ تا حد زیادی به کارایی RDFLib و منابع سخت‌افزاری سیستم وابسته است. در پروژه‌هایی که شامل میلیون‌ها سه‌تایی RDF هستند، ممکن است زمان بارگذاری و مصرف حافظه افزایش یابد.

۴-۳) مقایسه آنتواسپای با ابزارهای مشابه

برای ارزیابی بهتر جایگاه آنتواسپای، این کتابخانه با سه ابزار شناخته‌شده در حوزه مهندسی هستی‌شناسی، یعنی Protégé، OWLAPIو Apache Jena مقایسه شده است.

جدول ۱-۴: جدول مقایسه آنتواسپای و ابزارهای مشابه

ویژگی	OntoSpy	Protégé	OWLAPI	Apache Jena
زبان پیاده‌سازی	Python	Java	Java	Java
متن‌باز	✓	✓	✓	✓
تحلیل ساختار هستی‌شناسی	✓	✓	✓	✓
ویرایش هستی‌شناسی	✗	✓	محدود	محدود
تولید مستنداتHTML	✓	محدود	✗	✗
رابط برنامه‌نویسی(API)	✓	محدود	✓	✓
پشتیبانی ازReasoning	✗	✓	✓	✓
مناسب برای برنامه‌نویسی پایتون	✓	✗	✗	✗
رابط گرافیکی(GUI)	✗	✓	✗	✗
مناسب برای تحلیل خودکار	✓	متوسط	زیاد	زیاد

بر اساس مقایسه انجام‌شده، می‌توان نتیجه گرفت که آنتواسپای بیش از آنکه یک محیط توسعه یا ویرایشگر هستی‌شناسی باشد، ابزاری تخصصی برای تحلیل ساختاری، پیمایش و مستندسازی خودکار هستی‌شناسی‌ها است. در مقابل، Protégéبیشتر برای طراحی و توسعه هستی‌شناسی و OWLAPI و Apache Jena برای توسعه سامانه‌های مبتنی بر وب معنایی و پردازش پیشرفته داده‌های معنایی مورد استفاده قرار می‌گیرند.

در مجموع، انتخاب آنتواسپای زمانی مناسب است که هدف اصلی پروژه، تحلیل ساختار هستی‌شناسی، استخراج اطلاعات، تولید مستندات و یکپارچه‌سازی با برنامه‌های مبتنی بر Python باشد. در مقابل، اگر نیاز به طراحی

گرافیکی، ویرایش مدل یا اجرای استنتاج منطقی وجود داشته باشد، استفاده از ابزارهای مکمل یا جایگزین ضروری خواهد بود.

فصل پنجم

نتیجه‌گیری

با گسترش فناوری‌های وب معنایی و افزایش استفاده از هستی‌شناسی‌ها در حوزه‌هایی مانند مهندسی دانش، سیستم‌های هوشمند، داده‌های پیوندی و گراف‌های دانش، نیاز به ابزارهایی که بتوانند ساختار این هستی‌شناسی‌ها را به‌صورت خودکار تحلیل و مستندسازی کنند، بیش از پیش احساس می‌شود. در این پژوهش، کتابخانه آنتواسپای به‌عنوان یکی از ابزارهای متن‌باز و مبتنی بر زبان Python مورد بررسی قرار گرفت و قابلیت‌ها، معماری، نحوه عملکرد و کاربردهای آن در تحلیل هستی‌شناسی‌های مبتنی بر استانداردهای RDF و OWL ارزیابی شد.

در ابتدا، مفاهیم پایه مرتبط با وب معنایی، هستی‌شناسی و استانداردهای RDF ، RDFS و OWL معرفی شدند تا چارچوب نظری لازم برای شناخت عملکرد آنتواسپای فراهم شود. سپس ساختار داخلی کتابخانه، معماری لایه‌ای، فرآیند بارگذاری و تحلیل فایل‌های هستی‌شناسی و قابلیت‌های مختلف آن، از جمله استخراج کلاس‌ها، ویژگی‌های شیء، ویژگی‌های داده‌ای، نمونه‌ها، روابط معنایی و تولید مستندات HTML مورد بررسی قرار گرفت.

در بخش عملی پژوهش، با استفاده از یک سناریوی نمونه در حوزه مدیریت دانشگاه، فرآیند نصب، راه‌اندازی و استفاده از کتابخانه آنتواسپای تشریح شد. نتایج حاصل از اجرای آزمایش نشان داد که این کتابخانه قادر است ساختار یک هستی‌شناسی را با دقت مناسب استخراج کرده و اطلاعات مربوط به کلاس‌ها، ویژگی‌ها، نمونه‌ها و روابط میان آن‌ها را در قالبی منظم و قابل فهم ارائه کند. همچنین، قابلیت تولید خودکار مستندات

HTMLیکی از نقاط قوت مهم این ابزار به شمار می‌رود که فرآیند مستندسازی و بررسی ساختار هستی‌شناسی را برای پژوهشگران و توسعه‌دهندگان تسهیل می‌کند.

ارزیابی انجام‌شده نشان داد که آنتواسپای به دلیل سادگی استفاده، متن‌باز بودن، ارائه رابط برنامه‌نویسی مناسب و سازگاری با اکوسیستمPython ، گزینه‌ای مناسب برای تحلیل و مستندسازی هستی‌شناسی‌ها است. در مقابل، این کتابخانه محدودیت‌هایی نیز دارد؛ از جمله عدم پشتیبانی از ویرایش هستی‌شناسی، نبود قابلیت استنتاج منطقی و امکانات محدود در زمینه تجسم تعاملی ساختارهای معنایی. ازاین‌رو، در پروژه‌هایی که نیازمند طراحی، ویرایش یا استنتاج پیشرفته هستند، استفاده از آنتواسپای در کنار ابزارهایی مانندProtégé ، OWLAPIیا Apache Jenaمی‌تواند راهکار مناسب‌تری باشد.

به‌طور کلی، نتایج این پژوهش نشان می‌دهد که آنتواسپای ابزاری کارآمد برای تحلیل ساختاری، پیمایش و مستندسازی خودکار هستی‌شناسی‌ها است و می‌تواند در پروژه‌های مرتبط با وب معنایی، مدیریت دانش، داده‌های پیوندی و سامانه‌های مبتنی بر گراف دانش به‌عنوان یک ابزار مکمل مورد استفاده قرار گیرد. سادگی استفاده، قابلیت توسعه و امکان یکپارچه‌سازی با سایر کتابخانه‌هایPython ، از مهم‌ترین عواملی هستند که جایگاه این کتابخانه را در میان ابزارهای تحلیل هستی‌شناسی تقویت می‌کنند.