University of Ottawa/Université d'Ottawa

Faculty of Engineering School of Electrical Engineering and Computer Science

Assignment 2

Name : - *Aastha Aastha (300319310)*

Snehal Sudhir Bhole (300318105)

Course - *CSI5101 Knowledge Representation*

Semester - *Winter*

Instructor - *Professor Caroline Barrière*

Submission Date - *Feb 28, 2023*

Ontologies

Q1 – Ontology Design (12 points)

In Ontology Design, an important step is to provide "Competency Questions", so figuring out what is in-scope and out-of-scope for the ontology that we design. One other principle is to not reinvent the wheel and look for existing ontologies that could correspond to our need.

Assume you need to design a small ontology about the University environment which would show classes and properties, as well as a small KB (Knowledge Bases) which would instantiate some classes defined in the ontology.

Assume the following in-scope questions:
- **What is the list of departments at the University of Ottawa?**
- **What are the courses offered in the department of chemistry?**
- **What are the scholarships available for international students?**

Assume the following out-of-scope question:
- **What is the average age of professors in the EECS department?**

TO DO:

- **Add 2 more in-scope competency questions. Add 2 more out-of-scope competency questions.**

  Source

  In-scope competency questions
  - Are international students eligible for the Schulich Leader Scholarship?
  - If a student is enrolled in masters in computer science, which department should he head to for queries?

  Out-of-scope competency questions
  - What is the favourite food of the students studying in the department of Engineering?
  - What is the common hair colour among all the international students?

- **Explore Schema.org CollegeOrUniversity class (they call them types) https://schema.org/CollegeOrUniversity:**
  - **Does it contain sufficient properties for your needs?**

    Source

    In my overview, the above schema.org CollegeOrUniversity class gives a general overview of any educational organisation and describes its properties. It contains various properties which can define any college or university such as : -

Name (Text) - The name of the college or university
URL (URL) - The URL of the college or university website
Description (Text) - A description of college or university
Address (PostalAddress or Text) - A postal address describing the location of the college or university
Alumni (Person) - A person who has attended the college or university before
Department (Organisation) - Various departments present in the organisation
Member or Employee (Person) - Number of employees or professors working in the college or university
Number of Employees (Number) - Number of employees or professors working in the college or university

These properties are common to any educational organisation and provide sufficient high level perspective. Whereas, while looking at our specific requirements for the knowledge base based on competency questions, there are certain properties which are missing in the class. We require the data for the courses offered, number of students enrolled in each course, diversity of student enrolled(international students) and data for scholarship as well. Hence, the above CollegeOrUniversity class gives detailed description for the general scenario but for our needs it needs to be a bit more specific and elaborate to the requirements.

- **Notice that multiple inheritance is permitted, as CollegeOrUniversity is a subtype of EducationalOrganization which is a subtype of both CivicStructure and Organization. Why do you think that is appropriate (or not)?**

  Schema.org's trait of inheriting properties from other types is helpful because it enables more specialised types to inherit more general properties from their parent types. CollegeOrUniversity is a subtype of EducationalOrganization, which is a subtype of CivicStructure and Organization in the instance of CollegeOrUniversity. This makes sense given that colleges and universities are both groups that provide instruction and have physical structures and legal entities. This makes it possible for the CollegeOrUniversity type to acquire both educational properties as well as properties relating to organisational structure and geographic location.

- **What are some properties inherited from EducationalOrganization that seem strange to you to describe a University?**

  Some of the properties that a university inherits from an educational organisation, however, seem odd or not completely appropriate. The CollegeOrUniversity class has a “hasMerchantReturnPolicy” which seems a little strange to describe a college or a university as its more inclined towards organisational properties. A university, for instance, might not be able to use the feesAndCommissionsSpecification property because they don't usually charge fees or commissions the same way that businesses do. Similar to how accreditation is frequently a necessity for primary and secondary schools, but not always for universities, the accreditation property may not be as important for universities as it is for other types of educational organisations.

- **What about this other university ontology(link) ?**
  - **Does it contain what you need? Explore this ontology a bit.**

    This ontology describes the ontology language SHOE (Simple HTML Ontology Extensions) which is created specifically for the web. An extension of HTML called SHOE enables web designers to enrich online pages with semantic data. The Univ1.0 ontology is a representation of a university domain that includes details on studies, publications, and classes. Classes like "Course," "Professor," "Department," and "Publication" are included in the Univ1.0 taxonomy. More particular classes are subclasses of more general classes in the hierarchy that these classes are arranged in. For instance, a subcategory of "Course" is "Graduate Course."

    Additionally, the ontology has properties that outline the connections between various groups. For instance, the "Offers" property is used to connect a University to the courses it offers. Similar to this, the property "taughtBy" connects a course to the professor who instructs it.

    The Univ1.0 ontology's overall goal is to provide a rich representation of the academic domain that can be used to support a variety of applications, including academic search engines, study management, and course scheduling and it is more closer to our specific requirements. It specifies most of the properties and relations we need.

- **Building on classes from the 2 above-mentioned ontologies, define your own small university ontology and KB containing the classes and relations to answer your 5 competency questions.**
- **You can write out your ontology by providing for each class: name, superclass, properties, and by providing for each property: domain, range.**
- **You can populate your small KB with only a few examples that could answer the competency questions.**

Write your answer to contain all the information required and organized the way you want to best present your design and your exploration. Your answer can take a few pages if needed.

Sources

- https://www.researchgate.net/figure/An-example-part-of-University-Ontology_fig3_4001864
- https://www.ontotext.com/knowledgehub/fundamentals/what-is-a-knowledge-graph/

- https://protege.stanford.edu/publications/ontology_development/ontology101.pdf

We can create our own university ontology by defining the classes, relations and examples. For each class, we describe the name, its superclass, properties. For each property we will define the domain and range for that specific property.

UNIVERSITY ONTOLOGY

CLASSES

1. **University -** An institution for higher education
   Superclass - Thing
   Properties - hasDepartment, hasFaculty

2. **Department -** A unit within a university or college that imparts education in a specific domain
   Superclass - Organisation
   Properties - hasFaculty, hasStudent, hasCourse, hasProfessor

3. **Faculty -** A group of employees or staff members within a department
   Superclass - Person
   Properties - Qualification, coursesTaught, Department

4. **Student -** A person enrolled in an academic program at a university
   Superclass - Person
   Properties - coursesEnrolled, homeCountry, hasScholarship

5. **Course -** An academic program offered by various departments
   Superclass - CreativeWork
   Properties - teachingProfessor, Duration

6. **Professor -** A member of faculty who teaches a courses in a specific department
   Superclass - Person
   Properties - Publications, coursesOffered, Availability

7. **Scholarship -** An amount of money provided to the student studying in a specific course in a department
   Superclass - CreativeWork
   Properties - typeOfScholarship, Amount, Eligibility

PROPERTIES

1. **hasDepartment –**
   Domain: University,
   Range: Department
   Connects a university to its departments**.**

2. **hasStudent -**
   Domain: Department,
   Range: Student
   Relates a department to its students
3. **hasCourse -**
   Domain: Department,
   Range: Course
   Connects a department's classes.
4. **hasProfessor -**
   Domain: Department,
   Range: Professor
   Connects a school to its faculty.
5. **hasPublication -**
   Domain - Professor,
   Range - Publication
   hasPublication connects a professor's works with them.

RELATIONS

- **hasDepartment(University, domain) -** defines a relation between the number of departments a university has
- **hasFaculty(Department, professor) -** defines a relation between professor and the department they belong to
- **hasStudent(Department, student) -** relates the students to their specific department
- **offers(Department, Course) -** defines a relation between courses offered by each department
- **taughtBy(Course, Professor) -** defines a relation between courses taught by each professor
- **hasPublication(Professor, Document) -** provides a relation between professor and his/her publications

<u>UNIVERSITY KNOWLEDGE BASE</u>

University of Ottawa:

Department of Physics:
- **Courses:** “Magnetic Induction”, “Optical Fibre”
  - **Students:** “Zing Pao”, “Snehal Bhole”
    - **Scholarship:** “International Financial Aid”
- **Professors:** “Prof. Paola Flocchini”, “Prof. Shiva Nejati”
  - **Publication:** “Compressive Sensing using corona based architecture”

Department of Engineering:
- **Courses:** “Data Management”, “Ad hoc Networking”
  - **Duration:** “4 months”
- **Professors:** “Prof. Amiya Nayak”, “Prof. Caroline Barriere”

Relations:

- Department of physics <u>hasProfessor</u> Shiva Nejati
- Department of Engineering <u>hasCourse</u> Data Management
- Department of physics <u>hasStudent</u> Snehal Bhole
- Prof. Paola Flocchini <u>hasPublication</u> Compressive Sensing using corona based architecture
- Zing Pao <u>hasScholarship</u> International Financial Aid

Knowledge Base Ontology Diagram of University Knowledge Base: -

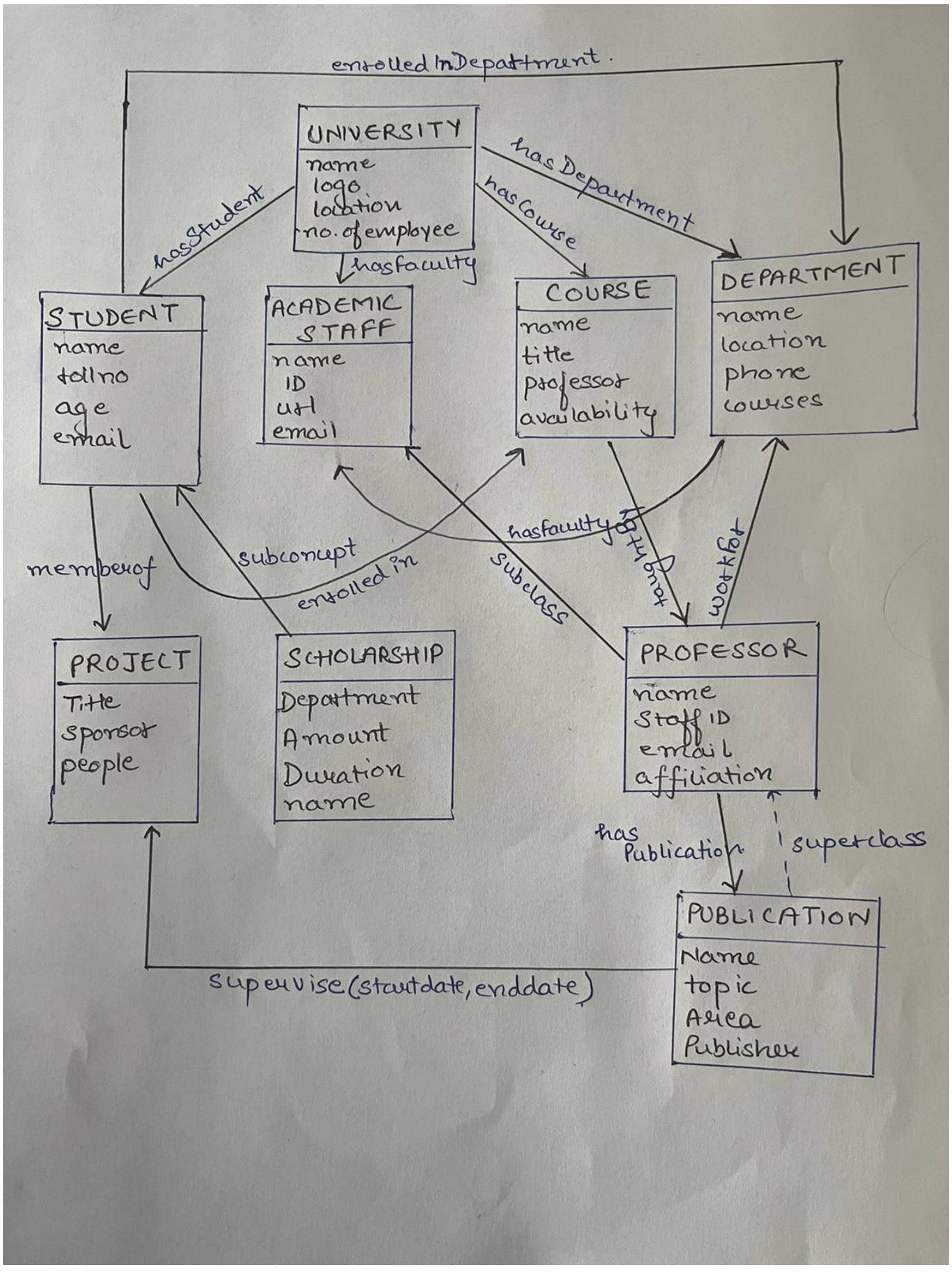

Figure 1: Knowledge Base Ontology Diagram of University Knowledge Base

Q2 – Linked Open Data (LOD) in Biomedical domain (5 points)

Some domains, like the biomedical domain, have developed and adopted various ontologies. Furthermore, the ontologies are linked to each other, as promoted by the Linked Open Data philosophy.

If you start here:https://guides.lib.umich.edu/ontology/ontologies, you'll get a list of different ontologies. We'll focus on Disease Ontology (https://disease-ontology.org/) .

Pick one disease (does not matter each one). You'll see links to ICD9, ICD10, MESH, SNOMEDCT, UMLS. These are all vocabularies or ontologies in the medical domain, and there are links from a particular disease to those other vocabularies.

TO DO:

- **Define your starting point (what did you choose as disease).**
- **Explore the other ontologies linked to (ICD9, ICD10, MESH, SNOMEDCT, UMLS). What are those? Are they open source? Are they published by authorities? Are they ontologies, or are they lists or vocabularies (without a hierarchical backbone and properties)?**
- **From the disease you chose, what does each ontology provide as additional information that the Disease Ontology does not have?**

Write your answer to contain all the information required and organized the way you want to best present your exploration. The purpose is to discover the various medical ontologies and reflect on why there are so many, and also on how they are linked to each other. Take the time to explore. If possible, show information about your starting point (disease) from each ontology (ICD9, ICD10, MESH, SNOMEDCT, UMLS). Some of them require a login, so it won't be possible.

[Parkinson's Disease: Causes, Symptoms, and Treatments]()

The disease chosen is Parkinson's Disease.Parkinson's disease is a brain disorder that causes unintended or uncontrollable movements, such as shaking, stiffness, and difficulty with balance and coordination. Symptoms usually begin gradually and worsen over time. As the disease progresses, people may have difficulty walking and talking.

Following Ontologies are linked with the disease:

Ontologies	Open Source	Publishing Authority	Properties
MESH:D010300	Yes	National Library of Medicine (NLM)	Ontology with hierarchical backbone
GARD:10251	Yes	National Library of Medicine (NLM)	Ontology with hierarchical backbone
OMIM:PS168600	No	Johns Hopkins University	Vocabulary without a hierarchical backbone
ICD9CM:332	No	World Health Organization (WHO)	Vocabulary with hierarchical backbone
ORDO:2828	Yes	Orphanet	Ontology with hierarchical backbone
NCI:C26845	Yes	National Cancer Institute (NCI)	Ontology with hierarchical backbone
UMLS_CUI:C0030567	No	National Library of Medicine (NLM)	Vocabulary with hierarchical backbone
ICD10CM:G20	Yes	World Health Organization (WHO)	Vocabulary with hierarchical backbone

Table 1: Ontologies Comparison Table

ICD9: International Classification of Diseases, 9th Revision was used in the United States till 2015. It provides standardised language for healthcare professionals to communicate information about patient conditions. It is an open source resource.

ICD10: International Classification of Diseases, 10th Revision is prominently used all over the world to classify and code diagnosis, symptoms and medical procedures. It is an open source resource.

MESH: Medical Subject Heading is thesuarus consisting of controlled vocabulary terms used to describe concepts and topics related to medicine and health care. It is an open source resource.

SNOMEDCT : (Systematized Nomenclature of Medicine--Clinical) Terms is a clinical terminology system used to describe and classify clinical data in electronic health records. It s multilingual terminology system that includes concepts in hierarchical for representing complex relationships between them. For the Parinson's disease, we could not find any link for SNOMEDCT .

UMLS : The Unified Medical Language System is a set of files and software that brings many biological vocabularies and standards together.It includes more than 2 million concepts and over 5 million terms from more than 200 different vocabularies and classifications.This service required Login and Login wasn't successful. Its a vocabulary with Hierarchical backbone and not available for all the users.

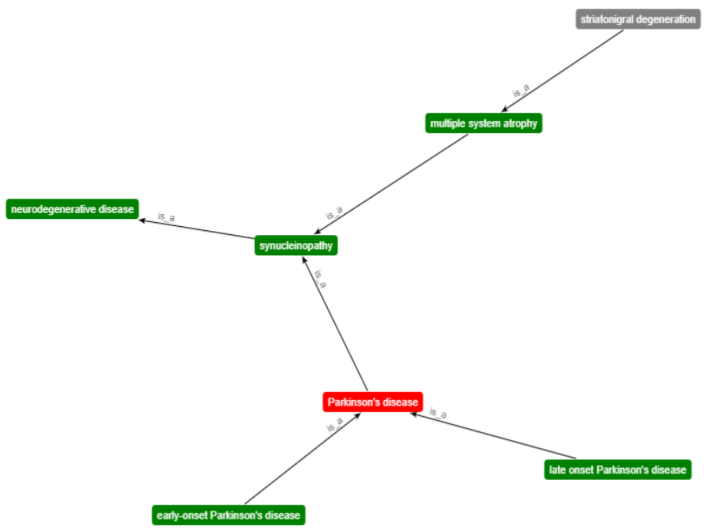

Figure 2: Hierarchical Visualisation for Parkinson's Disease from Disease Ontology

Figure 3: Hierarchical Visualisation for Parkinson's Disease from MESH

Overall, these ontologies provide additional information and context about Parkinson's Disease that is not available in the Disease Ontology, which allows for a more comprehensive and detailed understanding of the disease. Moreover, these Vocabularies and Ontologies are authorised by the reputed health organizations and used by Healthcare professionals as a knowledge base for medicine, medical procedures, etc.which allows for a more comprehensive and detailed understanding of the disease. Figure 2 and Figure 3 shows the differece in results generated by Disease Ontogy and MESH respectively, where it is evident that MESH has more details and comprehensive structure than disease ontology.

Sources:
MESH: Published by the National Library of Medicine (NLM).
Parkinson Disease
GARD: Published by the National Institutes of Health's (NIH) Office of Rare Diseases Research (ORDR)
OMIM: Published by Johns Hopkins University School of Medicine.
ICD9CM: Published by the World Health Organization (WHO).
http://icd9cm.chrisendres.com/index.php?action=child&recordid=2806
ORDO: Published by the European Bioinformatics Institute (EBI) and Orphanet.
NCI: Published by the National Cancer Institute (NCI).
UMLS_CUI: Published by the National Library of Medicine (NLM).
ICD10CM
https://www.icd10data.com/search?s=G20

Q3 – Knowledge Acquisition / Text Analytics (6 points)

This company NetOwl (https://www.netowl.com/) provides some services that could help the building of ontologies. I don't have a particular interest in that company, I'm just using it as an example mostly because it provides a good starting point to explore automatic knowledge acquisition (text analytics) and its relation to ontology building. Seeing that such companies exist is also a good indicator that the field is very much "alive" but that there is still lots to do.

TO DO:

- **Browse through this company's site, in particular go under "products" and then under Text Analytics. Describe each of these text analytics services (entity extraction, relationship extraction, event extraction, sentiment analysis, geotagging, categorization), and how they could be used (or not) in ontology building. How could each service be useful? Could it help to remove pressure from a domain expert, if such services were applied on domain-specific text? Could it completely replace a domain expert? Give your opinion and support your opinion with examples.**
- **For each service (entity extraction, relationship extraction, event extraction, sentiment analysis, geotagging, categorization), find another source (a blog, tutorial) which provides some information about the topic and helps understanding it. All these service fall under Text Analytics and are tasks of interest for Natural Language Processing researchers. It should be easy to find additional articles/blogs. Try to find some based on "good sources".**

Authoritativeness is a topic to be covered later in the semester… but you can start thinking about it.

- **The name NetOwl contains the word “net” and “owl”. Why does that seem like a good name for a company? Explain**

Write your answer to contain all the information required and organized the way you want to best present your exploration. The overall purpose of this question is to understand how some NLP tasks could be useful for ontology building. But automatic methods also have limitations, are error-prone and are dependent on the texts on which they are applied. Take the time to reflect about these issues.

NetOwl offers highly accurate, fast and scalable services in multiple language in AI-Based Natural Language Processing and Machine Learning Technologies.

- **Entity Extraction:**

Entity Extraction is a Text Analysis technique which uses NLP to extract data from unstructured data and classifies it according to categories. Entity Extraction enables Machines to identify or extract entities like product name,event, and location,etc. It is used to extract meaningful information in large amounts of unstructured data.

- **Relationship Extraction**

  Relationship extraction is a task where the relationship between two entities is found out by making use of linguistic features and context within a text. The task requires detection and classification of semantic relationships within a set of artifacts like text or xml documents. It is similar to Information Extraction with disambiguation of words and extraction of different relationships.

- **Event Extraction**

  Event extraction is a NLP technique that involves identifying nad extracting informatio about events from text data.The main aim is to automatically recognise and extract structured information. The extracted information generally includes types of event, time and location, participants involved and other relevant details. Event extraction typically uses various techniques like entity recognition, part-of-speech tagging and dependency parsing.

- **Sentiment Analysis**

  Sentiment Analysis is deriving insights from user experience and online reviews. Opinion Mining is another term used for sentiment analysis. It allows users to understand user reviews, opinions or thoughts and analyse them, and use the valuable insights and knowledge from large volumes of data for marketing research, customer feedback analysis, social media monitoring, etc.

- **Geotagging**
  Geotagging is a process of adding geographic coordinates or place names to media artifacts such as images, websites, text messages or videos based on the location where it is recorded. Geotagging provides helpful insights into customer activity. By using Geotags, its is easier to analyse consumer's buying patterns and customer's engagements with products.

- **Categorization**
  Categorization is the process of grouping objects or ideas into categories or classes based on their similarities and differences.In NLP, it is referred to automatically classifying text data into predefined categories or topics. Categories can be general news, domain specific or location based.

Relationship extraction can help in determining relationships between different entities extracted using entity extraction techniques. This is important for building the hierarchical structure of the ontology and identifying the connections. Information derived from event extraction assists in capturing the temporal aspects of ontology, and can represent the time-based relationships and events. Geotagging can help determine geographical aspects of ontologies, the geographical relationships between different concepts and identities. Categorization helps to classify different concepts and entities based on predefined classes and standards. All these techniques are important for building ontologies and it is essential in identifying categories, and structuring the information extracted from various sources of data, which can help represent the derived or acquired knowledge in the form of formal representation of data. Ontologies can be used for information retrieval, knowledge management and decision making.

Services provided by owlnet can be used by domain experts in ontology development and alignment, semantic annotations, querying and reasoning.
Domain experts can use resources to enhance their knowledge or extract some unprecedented data patterns or relationships. They can leverage tools and insight derived to create and refine ontologies accurately representing their expertise in the matter. These services can help integrate ontologies with other technologies, and thus can help domain experts to widen the scope of the domain.

These services cannot replace domain experts in spite of being accurate and efficient, these systems cannot fully automate certain tasks. It can't fully replace domain expertise,knowledge and cognitive skills acquired by domain experts. These tools can assist domain experts to work more productively and interpret based on their specialised knowledge. These systems still require human oversights for decision-making . Sometimes Entity extraction algorithms can struggle identifying context-specific entities or some events or categories may be misclassified, in such scenarios Domain Experts can review and annotate to ensure accuracy.Geotagging algorithms may have limited accuracy in specific regions, in areas with poor data coverage. Domain experts can identify and correct any errors or provide their additional inputs to provide additional context and insights in case of specific areas.

Additional Sources:

- **Introduction to Entity Extraction: What Is It And How It Works**

- [Relationship extraction]
- [Natural Language Processing — Event Extraction]
- [Sentiment Analysis: A Definitive Guide]
- [Geotagging]
- [Document Categorization & Text Classification]

The word “net” here represents a vast network of company’s data sources depicting a large semantic network that can be accessed by the company's services. The word “owl ”represents the company's ability to derive valuable insights and intelligence to its customers. The name NetOwl suggests a company that can navigate and extract details from a complex web of data. The name fits perfectly giving the expression that they are specialised in intelligent, intuitive and resourceful creatures which can quickly and accurately identify important information.

Q4 – Google Knowledge Graphs (GKG) and Search Engines (12 points)

As we discussed in class, although we do not yet have a Semantic Web as imagined by Tim BernersLee, a lot of the normalization (ontology development) is now used as annotation within web pages to help search engines locate particular structured information for which they provide a particular display. A lot of the normalized vocabulary is actually provided by Schema.org (https://schema.org/)

Here are two short articles about Schema.org and Google’s Knowledge Graphs (GKG) and their use in Search Engines (for SEO), both written by Michal Pecánek, 2020

- **Article about markup in web pages. What is Schema Markup? How to Use It for SEO?,**
- **Article about use of KG in Search Engines: Google’s Knowledge Graph Explained: How It Influences SEO,**

As a developer, you could also be interested by the Knowledge Graph API that Google provides, even if that is outside the scope of the current question.

Read the 2 articles suggested above, to be able to answer the following questions. The questions will also require that you navigate within Schema.org and explore classes and properties.

Sources

- https://kalicube.com/learning-spaces/faq/seo-glossary/knowledge-cards-in-seo-what-you-need-to-know/
- https://stackoverflow.com/questions/40805073/is-there-a-mapping-between-schema-org-datasets-and-wikidata
- https://www.ontotext.com/knowledgehub/fundamentals/linked-data-linked-open-data/
- https://www.w3resource.com/schema.org/CreativeWork.php

TO DO:

- **Define what Knowledge Panels/Cards are and why they are used?**

  Information boxes known as knowledge panels or cards can be found on the right side of Google search results for specific subjects or entities. They are made to give users fast access to pertinent and practical data related to the search query. Basic details like a brief description, images, related entities, and important facts are frequently included on knowledge panels and cards.

  For instance, when you look up a famous person or well-known company, you might see a knowledge panel or card with information about them, such as their picture, profession, birthplace, date, and noteworthy works and awards. Knowledge panels and cards show information that is automatically generated by Google algorithms and gathered from a variety of sources, including Wikipedia, Schema.org markup, and other reliable sources.

  Knowledge panels and cards have a variety of uses, such as:

  1. **Providing consumers easy access to pertinent information -** Users can rapidly understand the topic at hand without having to click on any links thanks to knowledge panels/cards, which offer a brief summary of information relevant to the search query.
  2. **Enhancing user experience:** Knowledge panels and cards make it easier for users to discover what they're looking for by summarising the information.
  3. **Website traffic growth:** If a knowledge panel or card contains a connection to a website, it may increase traffic there, which is good for website owners.
  4. **Establishing credibility and authority:** A website's inclusion in a knowledge panel or card can help to position it as a reliable source for information on the subject.

- **Describe the relation between GKGs and Schema.org? And between GKGs and Wikidata?**

  A database of interconnected entities and their attributes known as the Google Knowledge Graph (GKG) enables the company to give users more pertinent and contextual search results. A standardised vocabulary for structured data markup called Schema.org gives web developers a method to tell search engines what is on their web sites. Wikidata is a free, open-source knowledge base that offers structured data on people, places, things, and ideas. It is used by a number of programmes and services, including Google's Knowledge Graph.

  The connection between GKGs and Schema.org is that Google can receive organised data through Schema.org markup, which it can then use to improve its Knowledge Graph.Web designers can offer Google structured data about the entities and attributes on their web pages by using Schema.org markup. This will enable Google to better comprehend the content and give users more pertinent search results.

GKGs and Wikidata are related because Google's Knowledge Graph uses information from a range of sources, including Wikidata. Wikidata offers organised information on people, places, things, and ideas that can be used to improve the precision and thoroughness of Google's Knowledge Graph. Google can give users more pertinent and thorough search results and boost the precision of its machine learning algorithms by integrating data from Wikidata.

GKGs, Schema.org, and Wikidata all contribute to giving users more accurate and thorough search results by offering structured data about entities and their characteristics. Search engines like Google can better comprehend the content on web sites and give users more pertinent and accurate search results by utilising these resources.

- **In the second article, section 2 (Use schema markup on your site), look at the "sameAs" in the example. This "sameAs" link is fundamental in LOD. Why? What does it do?**

  The "sameAs" property is highlighted as a crucial component of schema markup in the article "Google's Knowledge Graph Explained: How It Influences SEO." The "sameAs" property is used to indicate that the current entity being described in the markup is the same as another entity that is identified by a URI or URL.

  The "sameAs" link is fundamental to Linked Open Data (LOD) because it makes it possible to join various data sources together. It is possible to build a web of connected data that can be used to give users more thorough and precise information by using the "sameAs" property to connect entities to other sources of information.

  The "sameAs" attribute is also crucial for SEO because it aids search engines in comprehending a web page's content and in providing users with more pertinent search results. Search engines can more accurately assess the relevance and importance of the content on a web page by using the "sameAs" property to connect entities on that page to other information sources. This can help to improve that page's ranking in search results.

  In conclusion, the "sameAs" attribute is critical for LOD because it establishes connections between various data sources. It is also crucial for SEO because it enables search engines to better comprehend the content of a web page and deliver more accurate search results to users.

- **In that same section 2, it mentions to use the Organization markup. Take a look at the Organization class in Schema.org:**
  - **The superclass of Organization is Thing, and some properties are inherited from Thing. What are some of these properties? Do they seem relevant to Organization?**

Any type of group, including businesses, nonprofits, and governmental institutions, can be represented by the Organization class in Schema.org. As stated, Thing is the superclass of Organization, from which it inherits some characteristics.

Some of the properties inherited by Organization from Thing includes: -

1. additionalType
2. alternateName
3. Description
4. Image
5. Name
6. Email
7. Url

Because they offer broad details about an organisation that may be valuable to search engines and other applications, these properties appear to be pertinent to Organization. Examples include using the "name" property to specify the name of the organisation and the "description" property to give a succinct explanation of what the organisation does. A connection to the company's website could be provided using the "url" property.

The properties that were passed down from Thing offer a foundational collection of data that can be used to characterise any kind of entity, including organisations. To provide more specific information about the organisation, such as its location, contacts, and the services it provides, additional properties unique to the Organization class can be added.

- **Organization can be the range for which properties?**

The following are a few examples of properties whose range is Organization:

Employee: A term used to describe a company employee.
Founder: A term used to describe the individual who founded an organisation.
Member: A term used to describe an individual or group that belongs to an organisation.
parentOrganization: Describes the larger organisation to which a particular group belongs.
subOrganization: a smaller organisation that is a component of a bigger organisation.
Alumni: A term used to describe an individual who has completed their education.
award Recipient: a term used to identify an individual or group that has been given an award by an institution.
sponsor: Used to indicate a person or organization that sponsors an event or activity organized by another organization.

These characteristics can be used to provide more detailed information about a company and its connections to other organisations. For example, The "parentOrganization" property, on the other hand, could be used to specify the parent company that an organisation is a subsidiary of, while the "employee" property could be used to list the workers of an organisation.

- **Why is a type like CreativeWork used so much as a possible type for various properties?**

  As it covers a wide variety of content, including books, articles, music, and videos, the type "CreativeWork" is frequently used as a potential type for different properties in Schema.org. This type belongs to a large superclass of Schema.org content-specific types, including Novel, Movie, and MusicComposition. Without having to define a different type for each type of content, it is possible to represent any type of content that is mainly textual by using CreativeWork as a type, such as a blog post or a news article. This makes the ontology more scalable and adaptable while also simplifying it. A generic type like CreativeWork also makes it simpler to map and integrate with other knowledge networks and data sources.

- **Why is Text like “Text or PostalAddress” often used as possibly type for various properties? What are the advantages and disadvantages of using Text as a property type?**

  Text provides a simple and flexible method to represent textual information, which is why it is frequently used as a potential type for different properties, such as "Text or PostalAddress." For instance, a broad variety of names may be used to represent the "name" property of an Organization using the Text type. Similar to this, the Text type can also be used to symbolise the "description" property of numerous Schema.org types.

  The versatility of Text as a property type is one of its primary benefits. It enables the representation of a broad variety of textual data without the need to designate separate classes for various text types. This can facilitate the reuse of existing properties and save time and effort when developing new ontologies.

  The semantic expressiveness of the data may be constrained if Text is used as a property class, though. For instance, a property represented by Text cannot be further defined using more specialised types like GeoCoordinates or PostalAddress. This might make it harder for search engines and other tools to comprehend and make use of the data. Additionally, using Text for some properties might necessitate further processing to obtain the necessary data, which can be tedious and error-prone.

- **Some classes don't have the same status in Schema.org, for example, look at VirtualLocation what's its status? What does that mean in the context of ontology development? Is Schema.org curated? If yes, by whom? Are they to be trusted?**

  The "unstable" label on the VirtualLocation class in Schema.org indicates that its description and properties could be altered in upcoming iterations of the schema. The class is still being developed, as evidenced by this status, and is not yet commonly used.

  The "unstable" state of VirtualLocation in the context of ontology development implies that the class may not yet be completely developed or well-defined. Because of this, using this class in ontology creation might need extra care and investigation to make sure it complies with industry best practises and new standards.

  Google, Microsoft, Yahoo, and Yandex worked together to maintain the Schema.org website. The schema is intended to be open and available to all, with community feedback and contributions playing a role in its development, even though the curation process is supervised by these significant tech companies.

  In general, Schema.org is regarded as a reputable and trustworthy source for organised data markup. Before using any classes or properties in a particular project, it is crucial to thoroughly assess their value and applicability, as with any resource.

- **A PostalAddress is a subclass of ContactPoint, why would an ontology want to make that distinction?**

  Due to the fact that they reflect various types of information, PostalAddress and ContactPoint must be distinguished. A physical location is represented by a PostalAddress, whereas a contact method, such as an email address, phone number, or social media handle, is represented by a ContactPoint.

  Search engines and other applications that need to comprehend and arrange data can benefit from an ontology's ability to make this distinction by being able to provide more precise and thorough information about an entity. Using the ContactPoint information, for instance, a search engine could provide contact details for a company or group while using the PostalAddress information to provide location-based search results.

- **Go in ReadAction. Look at the examples. They differentiate John read an article. John read a book. And John read a code. How is such differentiation done?**

  The object property in Schema.org is used to distinguish between various categories of reading content in the ReadAction class. The object property identifies the reading object, which may be one of many kinds, including an article, book, or code.

For example, the statement "John read an article" would be represented in Schema.org as follows:

```
{
        "@context": "http://schema.org",
         "@type": "ReadAction",
         "agent": {
                "@type": "Person",
                "name": "John"
         },
        "object": {
        "@type": "Article",
        "name": "Article Title"
        }
}
```

Similarly, the statement "John read a book" would be represented as:

```
{
        "@context": "http://schema.org",
        "@type": "ReadAction",
        "agent": {
                "@type": "Person",
                "name": "John"
        },
        "object": {
                "@type": "Book",
                "name": "Book Title"
        }
}
```

Schema.org enables differentiation between various kinds of reading materials within the ReadAction class by allowing for the specification of the type of object being read using the @type property within the object property.

- **Let say you wanted to express that Bora went to the National Art Gallery in Ottawa to admire the painting "Forest" by Paul Cézanne, what classes could be assigned to "John", "National Art Gallery", "Ottawa", "Forest", "Paul Cézanne". What are the properties between them? Inspired by the JSON-LD format shown in the ReadAction page examples, represent the sentence above using that format.**

  The following classes and properties can be added to the sentence "Bora went to the National Art Gallery in Ottawa to admire the painting 'Forest' by Paul Cézanne" to represent it in JSON-LD format:

- **Bora:** Person
- **National Art Gallery:** Museum
- **Ottawa:** City
- **Forest:** VisualArtwork
- **Paul Cézanne:** Person

The properties between them can be defined as:

- The reason for the visit was to admire the painting “Forest”.
- Bora performed an action of visiting the National Art Gallery in Ottawa.
- The painting was created by Paul Cézanne.

The JSON-LD format for this representation would be as follows:-

```
{
  "@context": "https://schema.org",
  "@type": "VisitAction",
  "agent": {
    "@type": "Person",
    "name": "Bora"
  },
  "actionStatus": "CompletedActionStatus",
  "location": {
    "@type": "Museum",
    "name": "National Art Gallery"
  },
  "toLocation": {
    "@type": "City",
    "name": "Ottawa"
  },
  "description": "To admire the painting 'Forest' by Paul Cézanne",
  "object": {
    "@type": "VisualArtwork",
    "name": "Forest",
    "creator": {
      "@type": "Person",
      "name": "Paul Cézanne"
    }
  }
}
```

Figure 3: JSON-LD format

Q5 - Propositional logic and proofs (4 points)

This book, A Concise Introduction to Logic, by Craig DeLancey is freely available online.

https://milnepublishing.geneseo.edu/concise-introduction-to-logic/

Chapter 4 talks about proofs and is an easy read.

https://milnepublishing.geneseo.edu/concise-introduction-to-logic/chapter/4-proofs/

In 4.5, there are some exercises suggested.

e. Premises: $(S \rightarrow \neg Q)$, $(P \rightarrow S)$, $\neg\neg P$. Show: $\neg Q$.
f. Premises: $(T \rightarrow P)$, $(Q \rightarrow S)$, $(S \rightarrow T)$, $\neg P$. Show: $\neg Q$.
g. Premises: R, P, $(P \rightarrow (R \rightarrow Q))$. Show: Q.
h. Premises: $((R \rightarrow S) \rightarrow Q)$, $\neg Q$, $(\neg(R \rightarrow S) \rightarrow V)$. Show: V.
i. Premises: $(P \rightarrow (Q \rightarrow R))$, $\neg(Q \rightarrow R)$. Show: $\neg P$.
j. Premises: $(\neg(Q \rightarrow R) \rightarrow P)$, $\neg P$, Q. Show: R.
k. Premises: P, $(P \rightarrow R)$, $(P \rightarrow (R \rightarrow Q))$. Show: Q.
l. Premises: $\neg R$, $(S \rightarrow R)$, P, $(P \rightarrow (T \rightarrow S))$. Show: $\neg T$.
m. Premises: P, $(P \rightarrow Q)$, $(P \rightarrow R)$, $(Q \rightarrow (R \rightarrow S))$. Show: S.
n. Premises: $(P \rightarrow (Q \rightarrow R))$, P, $((Q \rightarrow R) \rightarrow \neg S)$, $((T \rightarrow V) \rightarrow S)$. Show: $\neg(T \rightarrow V)$.

TO DO:

- **Choose 4 of these (between e and n) that you prove using rules of inferences.**
- **Present each proof as we saw in class (e.g. slide 28):**
    - **State the premises (assigning a number to each one).**
    - **For each intermediate step (until you arrive at the conclusion), state what is derived (either inferred or equivalent) and how (which equivalence rule or which inference rule is used).**

Answer 5 -

- Premises: (S→¬Q), (P→S), ¬¬P. Show: ¬Q.
  1. (S → ¬Q)
  2. (P → S)
  3. ¬¬P
  4. P (Double negation, premise 3)
  5. S (Modus ponens, premise 2 and 4)
  6. ¬Q (Modus tollens, premise 1 and 5)

- Premises: ¬Q, (¬Q→S). Show: S.
  1. ¬Q
  2. (¬Q → S)
  3. S (Modus ponens, premises 1 and 2)

- Premises: (T→P), (Q→S), (S→T), ¬P. Show: ¬Q
  1. (T → P)
  2. (Q → S)
  3. (S → T)
  4. ¬P
  5. ¬T (Modus tollens, premises 1 and 4)
  6. ¬S (Modus tollens, premises 3 and 5)
  7. ¬Q (Modus tollens, premises 2 and 6)

- Premises: P, (P→Q), (P→R), (Q→(R→S)). Show: S.
  1. P
  2. (P → Q)
  3. (P → R)
  4. (Q → (R → S))
  5. Q (Modus ponens, premises 1 and 2)
  6. (R → S) (Modus ponens, premises 4 and 5)
  7. R (Modus ponens, premises 1 and 3)
  8. S (Modus ponens, premises 6 and 7)

Q6 – Predicate logic (3 points)

Predicate logic has a certain expressive power. There are facts it can express and others it cannot. Its representation relies mostly on predicates (unary and binary) and on quantifiers (universal and existential).

TO DO:

Express the following statements in predicate logic. If you think the statement cannot be expressed in first-order logic, explain why.

- **All graduate students have an undergraduate degree.**

  Predicate: $G(x)$ means x is a graduate student.
  Predicate: $U(x)$ means x has an undergraduate degree.
  Statement: $\forall x(G(x) \rightarrow U(x))$

- **Anyone with an undergraduate degree studied in at least one university.**

  Predicate: $U(x)$ means x has an undergraduate degree.
  Predicate: $S(x, y)$ means x studied in y.
  Statement: $\forall x(U(x) \rightarrow \exists y(S(x, y)))$

- **A university, at any particular year, deliver courses to many students.**

  Predicate: $U(x)$ means x is a university.
  Predicate: $D(x, y)$ means x delivers courses to y.
  Predicate: $S(z, y)$ means z is a student of y.
  Statement: $\forall x \forall y(U(x) \land D(x, y) \rightarrow \exists z(S(z, y)))$

- **For a professor to teach a course, there must be a student attending it.**

  Predicate: $P(x)$ means x is a professor.
  Predicate: $C(y)$ means y is a course.
  Predicate: $T(x, y)$ means x teaches y.
  Predicate: $S(z, y)$ means z is a student attending y.
  Statement: $\forall x \forall y((P(x) \land C(y) \land T(x, y)) \rightarrow \exists z(S(z, y)))$

- **Classmates are students taking the same course.**

  Predicate: $S(x, y)$ means x is a student of y.
  Predicate: $C(x)$ means x is a course.
  Predicate: $Cl(x, y)$ means x and y are classmates.
  Statement: $\forall x \forall y((S(x, z) \land M(y, z) \land C(z)) \rightarrow Cl(x, y))$

- **A graduate student takes less courses than an undergraduate student.**

  Predicate: $G(x)$ means x is a graduate student.
  Predicate: $U(x)$ means x is an undergraduate student.
  Predicate: $T(x, y)$ means x takes fewer courses than y.
  Statement: $\forall x(G(x) \rightarrow \forall y((U(y) \land T(x, y)) \lor \neg U(y)))$

  The statement given here is ambiguous as it doesn't specify whats is "fewer courses". This statement cannot be expressed in first-order logic as it involves comparing the "number" of courses taken by two different individuals. First-order logic does not have a way to express this kind of comparison.

Q7. Fuzzy Logic(8 points)

You are asked to develop a fuzzy logic system for representing the impact of stress level and leisure time on quality of sleep. I know we are not expert in the field… but each of you can make up their own fuzzy sets and rules based on their experience. The purpose of the question is to practice developing a fuzzy logic system, not on its accuracy

TO DO:

- **Define 3 linguistic variables for sleep, leasure and stress. For each one, define its name, range (min/max values) and fuzzy subsets. For example, stress level can range from 1 to 15 and there could be 4 subsets (low, average, high, veryhigh). Each fuzzy subset should define the membership degrees associated with them. For doing so, draw a membership graph which provides the degree of membership of each value in your range to each subset defined.**
- **Define 3 fuzzy rules which use in different combinations the fuzzy sets defined (e.g. if stress level is average and leasure is minimal then sleep quality is poor). In those rules, stress and leasure are the antecedents and sleep is the consequence.**
- **Show an example of using fuzzy inference with your rules to establish a sleep quality (consequence) based on leasure time and stress level (antecedent). For doing so:**
    - **define an actual input (e.g. stress = 9 and leasure = 2 hours)**
    - **transform such input into membership values to your subsets**
    - **apply your 3 fuzzy rules to obtain membership values to the output set (e.g. sleep = poor)**
    - **transform the membership value of sleep to an actual number within the range that you decided.**

ATTENTION: This situation is totally hypothetical and you can make the linguistic variables and the rules as you like. Just try to make the system a bit realistic (e.g. if stress level if "very high", the sleep quality should not be "very good").

Linguistic Variables :

- Sleep Quality
  Range: 0-10 (where 0 represents very poor sleep quality, and 10 represents excellent sleep quality)
  Fuzzy subsets:
    - Very Poor: Range 0-2
    - Poor: Range 1-4
    - Average: Range 3-7
    - Good: Range 6-9
    - Excellent: Range 8-10

$$\text{Sleep Quality} = f(x) = \begin{cases} \text{Very Poor} & \text{if, } 0 < x < 2 \\ \text{Poor} & \text{if, } 1 < x < 4 \\ \text{Average} & \text{if, } 3 < x < 7 \\ \text{Good} & \text{if, } 6 < x < 9 \\ \text{Excellent} & \text{if, } 8 < x < 10 \end{cases}$$

Equation 1: Membership Function Defined for Sleep Quality

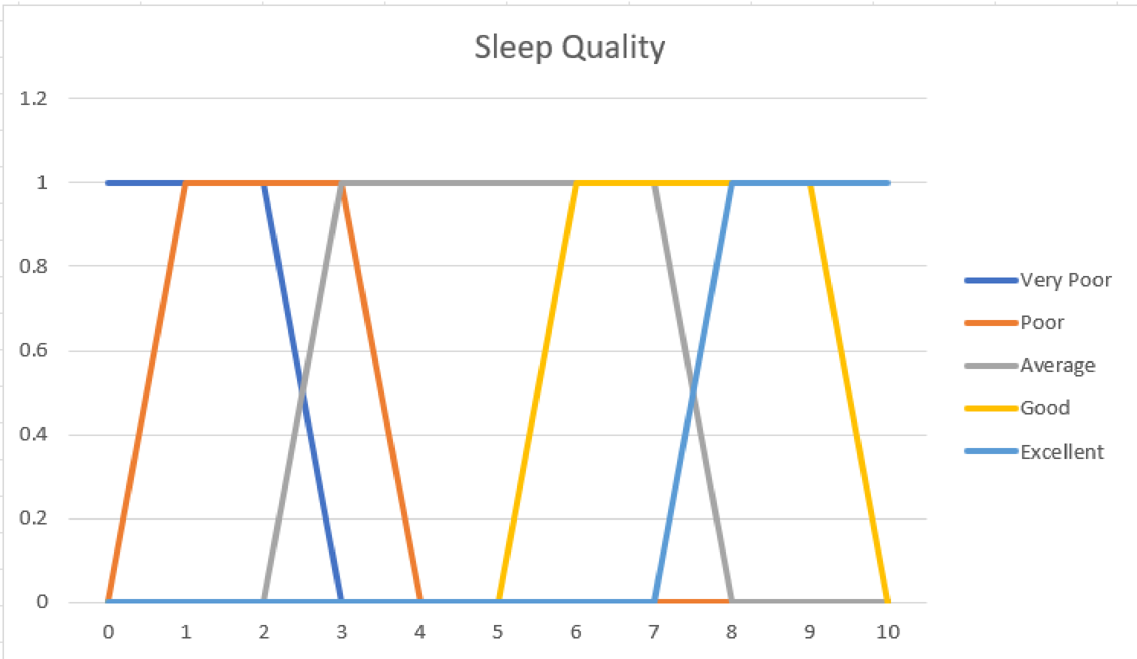

Figure 4: Fuzzy Membership function for Sleep Quality

- Leisure Time
  Range: 0-24 (where 0 represents no leisure time, and 24 represents a lot of leisure time)
  Fuzzy subsets:
  - None: Range 0-2
  - Little: Range 1-6
  - Some: Range 4-12
  - Moderate: Range 10-18
  - Lots: Range 16-24

$$\text{Leisure Time} = f(x) = \begin{cases} \text{None} & \text{if, } 0 < x < 2 \\ \text{Little} & \text{if, } 1 < x < 6 \\ \text{Moderate} & \text{if, } 4 < x < 12 \\ \text{Some} & \text{if, } 10 < x < 18 \\ \text{Lots} & \text{if, } 16 < x < 24 \end{cases}$$

Equation 2: Membership Function Defined for Leisure Time

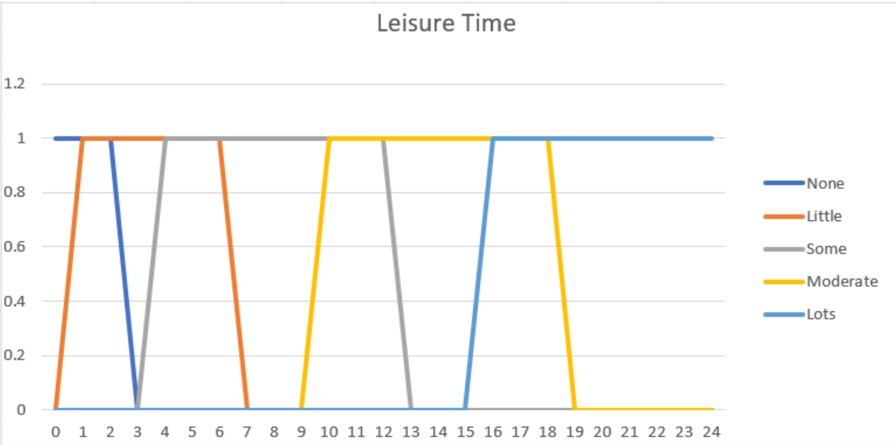

Figure 5: Fuzzy Membership function for Leisure Time

- Stress Level
  Range: 0-10 (where 0 represents no stress, and 10 represents very high stress)
  Fuzzy subsets:
    - None: Range 0-2
    - Low: Range 1-3
    - Moderate: Range 2-6
    - High: Range 4-8
    - Very High: Range 6-10

$$Stress = f(x) = \begin{matrix} None & \text{if, } 0 < x < 2 \\ Low & \text{if, } 1 < x < 3 \\ Moderate & \text{if, } 2 < x < 6 \\ High & \text{if, } 4 < x < 8 \\ Very\ High & \text{if, } 6 < x < 10 \end{matrix}$$

Equation 3: Membership Function Defined for Stress

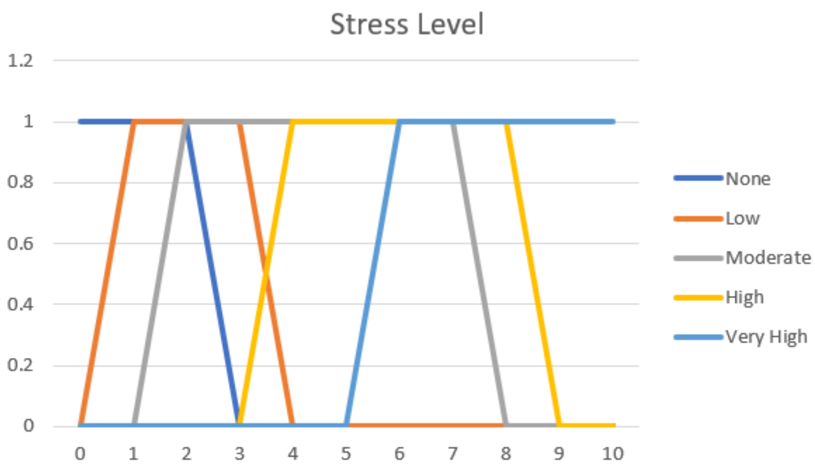

Figure 6: Fuzzy Membership function for Stress Level

The membership degrees associated with each fuzzy subset can be represented by the graphs in figure [1,2,3] and equations for the fuzzy subset are shown in equation 1,2 and 3.

Let the fuzzy rules for given linguistic variables be as follows:

- If leisure time is abundant and stress level is low or average, then sleep quality is good.
- If leisure time is low and stress level is high or very high, then sleep quality is poor.
- If leisure time is moderate and stress level is high, then sleep quality is moderate.

Example of using fuzzy inference:
Let the input values be as follows:
Leisure time: 4 hours
Stress level: 9

Transforming input to membership values:

- Leisure time: Little= 0.3, Some= 0.5
- Stress level: Very High= 0.9

Applying fuzzy rules:

1. Leisure time is Little(0.3) and stress level is Very High(0.9), then sleep quality is poor(0.4).
2. Leisure time is Some(0.5) and stress level is Very high (0.9), then sleep quality is Average(0.6).

Aggregating the results:

- Sleep quality:
    - Poor: 0.4
    - Average: 0.6

The resulting sleep quality is Average(0.6).