

Rapport Projet Web Sémantique

EL KATEB Sami, PAUL Thomas

[] La patient est lié inutilement à casereport et examination qui sont déjà liées [] Il faudrait avoir au moins un patient qui a plusieurs consultations et examens

Évolution de notre modélisation de rapports médicaux

Pour faire évoluer notre modélisation de rapports médicaux, nous avons commencé par intégrer les retours reçus sur notre premier projet. Nous avons donc remplacé la propriété *gender* par les classes Man et Woman.

Puis nous avons ajouté les numéros de sécurité sociale des patients et les numéros RPPS des médecins. Nous avons pu définir ces propriétés en tant que owl:InverseFunctionalProperty car deux entités ayant la même valeur pour l'une de ces deux propriétés implique que les entités sont identiques. Par conséquent, nous avons également pu les définir comme valeur de owl:hasKey pour les patients et les médecins respectivement.

Étant donné que nous avons débuté une ontologie owl, nous avons mis à jour la déclaration de notre ontologie en ajoutant la référence à la version précédente. Nous avons également corrigé les erreurs que nous avons faites concernant les propriétés que nous avons définies comme ObjectProperty alors que celles-ci étaient des DataProperty.

Nous avons ajouté à notre ontologie des liens de parenté tel que hasAncestor qui est une propriété asymétrique, irreflexive et transitive et sa sous-propriété hasParent qui n'est pas transitive. Nous avons également ajouté la propriété hasChild qui est l'inverse de hasParent et possède les mêmes types que celle-ci. De plus, nous avons ajouté la propriété hasBrother et hasSister toutes deux disjointes et irreflexives, sous-propriété de hasSibling.

Ces liens de parenté nous permettent de mettre en évidence les prédispositions génétiques. Nous avons ainsi pu créer la classe PersonWithGeneticDiabetesPre-disposition qui inclut les personnes ayant un ancêtre diabétique ou un frère/sœur diabétique. Nous avons choisi de ne pas exclure les personnes diagnostiquées diabétique de cet ensemble car nous considérons que celles-ci sont toujours des personnes prédisposées génétiquement au diabète.

Par la suite nous avons défini l'ensemble des maladies respiratoires (RespiratoryCondition) comme l'union des maladies ayant pour symptôme la toux,

l'essoufflement ou le mal de gorge. Nous avons également défini l'ensemble des maladies infectieuses et son complément, l'ensemble des maladies non infectieuses.

Finalement, nous avons défini une équipe (Team) puis une équipe médicale (MedicalTeam) comme étant une équipe composée d'au moins un médecin.

Nous avons également introduit 3 thésaurus SKOS dans nos données: un thésaurus de médicament, un thésaurus de symptômes et un thésaurus des maladies. Ceux-ci se prêtent bien à être organisés dans un thésaurus car ils peuvent facilement être classés de manières hiérarchiques (différentes familles de médicaments, différents types de symptômes et de maladies). De plus les thésaurus permettent d'établir des relations entre les différents termes. Enfin la création de thésaurus permet de normaliser la terminologie. Nous avons également créé une collection de maladies contagieuses qui permet de regrouper les maladies à travers leur contagiosité alors que le thésaurus est lui organisé par organe atteint.

OWL Entailment

Lors de la création de notre ontologie OWL nous avons défini les clés des patients comme étant leur numéro de sécurité sociale à l'aide de la propriété `hasKey`. Cette propriété exprime que deux instances ayant le même numéro de sécurité sociale sont la même instance. Nous pouvons vérifier l'effet de cette propriété en recherchant le poids du Patient7. En effet l'instance Patient7 ne possède pas de propriété `weight`, cependant le Patient11 possédant le même numéro de sécurité sociale possède cette propriété. Étant donné que les deux instances sont considérées comme la même instance, nous pouvons récupérer le poids à partir du Patient7 (`mcr_query_ entailment 1`). Le même effet est également atteint à en définissant la propriété `ssn` comme `InverseFunctionalProperty`.

Nous avons également défini les `PersonWithGeneticDiabetesPredisposition` comme étant les personnes ayant un ancêtre atteint du diabète ou un sibling (frère/sœur) atteint du diabète. Cependant les seules relations définies dans nos données sont les relations `hasChild` et `hasBrother`. La relation `hasParent` est inférée à partir de la relation `hasChild` dont elle est l'inverse. Celle-ci permet ensuite d'inférer la relation `hasAncestor` dont elle est la sous-classe. La relation `hasAncestor` est transitive et s'applique donc sur plusieurs générations. Elle permet d'inférer des prédispositions génétiques à partir des diagnostics effectués sur les grands-parents. Ainsi en incluant uniquement les relations `hasChild` dans nos données, le raisonneur OWL RL infère donc correctement que les patients 2, 8 et 9 qui ne sont pas diagnostiquée diabétique ont une prédisposition génétique au diabète (`mcr_query_ entailment 2`).

Nous retrouvons dans les maladies respiratoire ...