

Projet de Web Sémantique : JO 2024

Par Marc Pinet, Arthur Rodriguez et Amine Haddou

Introduction et objectifs

Notre projet s'inscrit dans le cadre des Jeux Olympiques de Paris 2024, un événement majeur qui génère une quantité importante de données hétérogènes. L'objectif est de concevoir et mettre en œuvre une application permettant de gérer, d'intégrer et d'exploiter ces données de manière sémantique, en utilisant les technologies du Web des données.

Architecture Sémantique

Modélisation Ontologique

Notre modélisation ontologique, définie dans `og24_schema.ttl`, adopte une approche modulaire et hiérarchique qui couvre l'ensemble des aspects des Jeux Olympiques. L'ontologie se structure autour de plusieurs concepts clés :

La classe racine `jo:Entity` se ramifie en trois branches principales : `jo:Person`, `jo:SportingEvent`, et `jo:Location`. Cette organisation permet une classification claire des différents éléments du domaine. Nous avons soigné la modélisation des événements sportifs avec une hiérarchie détaillée :

- La classe `jo:SportingEvent` se spécialise en différentes sous-classes représentant les types d'épreuves (`jo:Event`), avec une distinction entre les épreuves qualificatives, finales et à médailles.
- Les sports sont catégorisés selon leur nature (individuel, équipe, mixte) via les classes `jo:TeamSport`, `jo:IndividualSport`, et `jo:MixedSport`.
- Les disciplines sont modélisées avec précision, notamment pour l'athlétisme (`jo:TrackEvent`, `jo:FieldEvent`) et les sports aquatiques (`jo:AquaticsEvent`).

Organisation des Connaissances avec SKOS

Le thésaurus SKOS (`og24_thesaurus.ttl`) complète l'ontologie en organisant hiérarchiquement les sports olympiques. Cette structuration permet une classification flexible et multilingue des disciplines. Par exemple :

```
# schéma de concepts principal
jo:SportsThesaurus rdf:type skos:ConceptScheme ;
    rdfs:label "Olympic Sports Classification"@en, "Classification des Sports Olympiques"@fr ;
    skos:hasTopConcept jo:TrackAndField, jo:Aquatics, jo:TeamSports, jo:CombatSports .
```

Cette approche facilite la navigation dans les différentes disciplines et permet d'établir des relations sémantiques entre les sports (notamment via `skos:related` et `skos:broader`).

Règles d'Inférence

Les règles SPARQL (`og24_rules.ttl`) enrichissent le modèle avec des capacités d'inférence automatique. Nous avons défini plusieurs types de règles :

1. Règles de qualification des athlètes
2. Règles de détection des records
3. Règles de gestion des équipes
4. Règles de comptabilisation des médailles
5. Règles de calcul de performances moyennes

Ces règles permettent d'automatiser de nombreux aspects de la gestion des données olympiques et d'enrichir le graphe de connaissances avec des informations dérivées.

Contraintes SHACL

Les contraintes SHACL (`og24_constraints.ttl`) garantissent l'intégrité et la qualité des données. Nous avons défini des contraintes pour :

- Les événements sportifs (dates, lieux, résultats)
- Les résultats (performances, classements)
- Les équipes (composition, représentation nationale)
- Les conditions météorologiques
- La qualification des athlètes

Ces contraintes assurent la cohérence du graphe de connaissances et facilitent la détection d'anomalies dans les données.

Cette architecture sémantique répond pleinement aux exigences sur lesquelles nous souhaitons mettre l'accent, c'est-à-dire :

- Proposant une modélisation ontologique originale et adaptée au domaine
- Intégrant des concepts SKOS pour la classification des sports
- Implémentant des règles d'inférence SPARQL pour l'enrichissement automatique des données
- Définissant des contraintes SHACL pour garantir la qualité des données

Enrichissement des Données

Pour commencer, nous avons récupéré nos données du projet précédent de Web of Linked Data (`og24_data.ttl`).

Enrichissement Initial (01_data_enricher.py)

Le processus d'enrichissement initial, implémenté dans la classe `OlympicsKnowledgeEnricher`, constitue la première étape de la transformation des données. Cette classe commence par traiter les données structurées provenant des fichiers CSV, notamment les informations sur les médailles olympiques (dataset mis à disposition par le gouvernement français, [disponible ici](#)). Le système génère des URIs uniques pour chaque entité (événements, athlètes, résultats) et prend soin de normaliser les données, particulièrement pour les performances sportives où les temps sont convertis en format décimal pour une manipulation plus simple.

Nous extrayons aussi des informations provenant de données non structurées, à savoir des articles que nous avons directement recolté et stocké dans un CSV avec les colonnes `date, source, title, content`). Le système exploite le LLM Français de spaCy (`fr_core_news_lg`) pour analyser les articles de presse. Cette analyse permet d'extraire automatiquement diverses informations contextuelles : les performances des athlètes, les records établis, les attributions de médailles, et les détails sur les sites olympiques. Le code implémente des expressions régulières (regex) pour capturer ces informations de manière plus précise.

Architecture du Micro-service SPARQL

Nous avons mis en place la classe `WeatherEnricher` qui s'interface avec un micro-service SPARQL dédié pour obtenir des données météorologiques en temps réel. Ces données sont ensuite associées aux événements via leurs lieux de déroulement, avec une conversion automatique des températures de Kelvin en Celsius pour une meilleure lisibilité.

⚠ Cependant, il est important de noter que due à l'incapacité du service docker de M. Michel à traiter correctement les flottants dans certains cas, l'API de météo qu'on utilise reçoit des points encodés en « %20 » ce qui rend impossible l'extraction via des coordonnées exactes. Pour contourner ce problème, nous avons simplement arrondi les coordonnées à l'unité, se basant ainsi uniquement sur des entiers (solution du professeur qui n'a malheureusement pas réussi à identifier le problème)

Le micro-service SPARQL pour la météo s'appuie sur l'API OpenWeatherMap et utilise une architecture définie dans les fichiers TTL de description de service. La validation des données est assurée par des contraintes SHACL, garantissant l'intégrité et la conformité des informations météorologiques. Un système de cache avec une durée de vie de 3600 secondes optimise les performances en limitant les appels API redondants.

La construction des graphes RDF est gérée par des requêtes CONSTRUCT. L'utilisation de requêtes SPARQL fédérées permet aussi une interrogation des données météorologiques en lien avec les autres informations du graphe.

Linkage des Données (02_data_linking.py)

Le processus de liaison des données, fait par la classe `KnowledgeGraphLinker`, établit des connexions avec des sources de données externes. Pour DBpedia, le système effectue une recherche des athlètes par leur nom et établit des correspondances pour les sites olympiques et les événements sportifs via

des tables de mapping préconfigurées. Ces liens sont enrichis avec des informations biographiques supplémentaires comme les dates de naissance et les nationalités.

La liaison avec Wikidata ajoute une dimension supplémentaire en recherchant les athlètes dans cette base de connaissances et en établissant des liens owl:sameAs. Cette approche double-source renforce la fiabilité des données et enrichit considérablement le graphe de connaissances, passant de 355 triplets à 15152 triplets après enrichissement et linkage.

Quelques visualisations...

Pour chacune des visualisations, vous pourrez retrouver la requête SPARQL associée dans le fichier : ./visualisations/app.py dans la variable « QUERIES ». D'autres requêtes seront disponibles dans le fichier og24_queries.sparql, comme l'exige le projet dans l'énoncé.

Athlètes par pays

<input type="button" value="France"/> ▼	<input type="button" value="Voir les athlètes"/>
 Bronze Medal	▲
Sofiane OUMIHA Boxing  Silver Medal	▲
Sylvain ANDRE Cycling BMX Racing  Silver Medal	▲
Teddy RINER Judo  Gold Medal	▼

Ici, nous avons mis en place un tableau interactif permettant de voir les athlètes avec les médailles qu'ils ont gagné et dans quelle(s) discipline(s).

Tableau des médailles

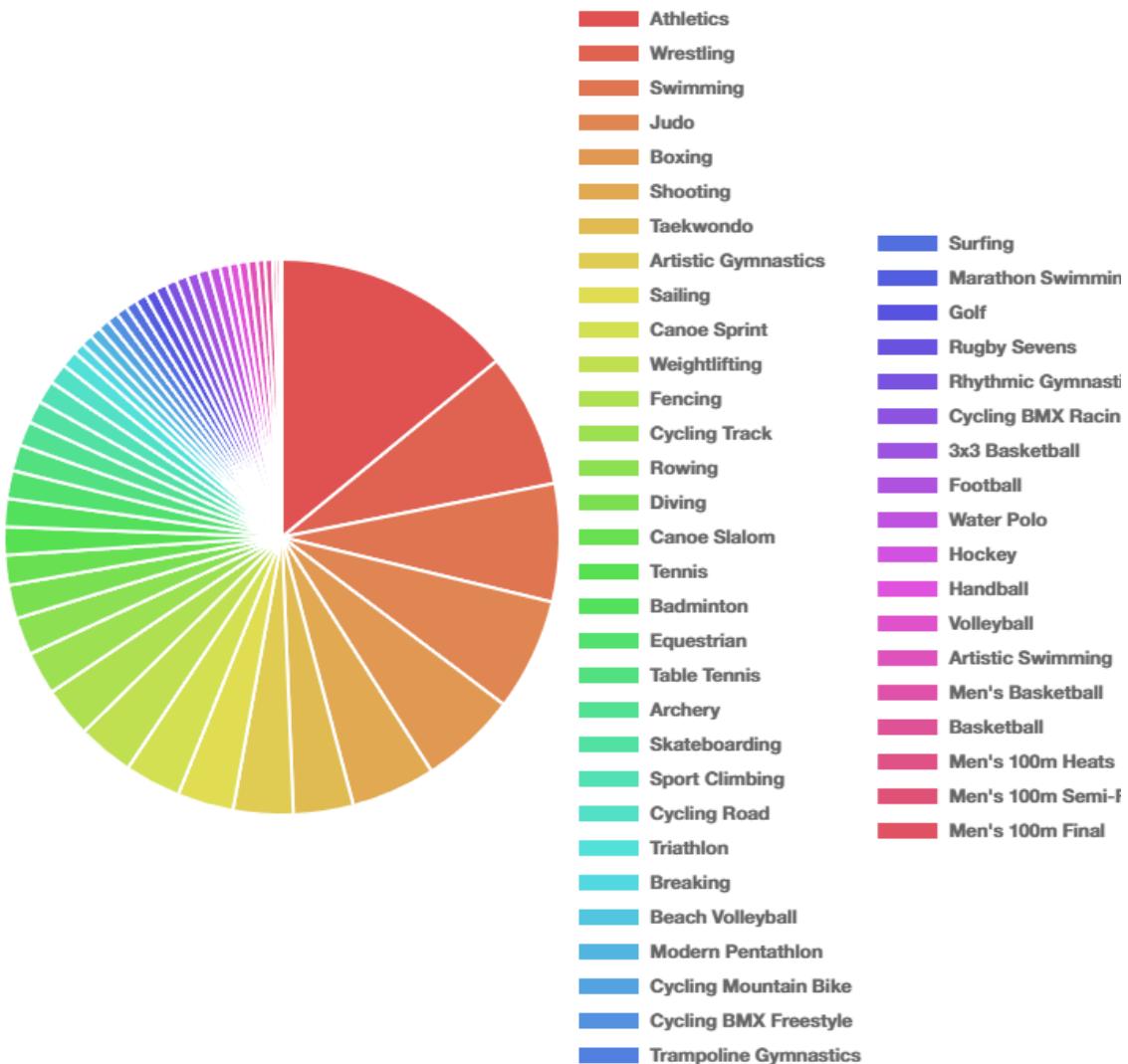
Actualiser

Pays	🥇	🥈	🥉	Total
China	40	27	24	91
United States	39	44	40	123
Japan	20	12	13	45
Australia	18	19	16	53
France	15	23	22	60
Great Britain	14	22	29	65
Netherlands	14	7	12	33
Korea	13	9	10	32
Italy	12	13	15	40
Germany	12	12	8	32
New Zealand	10	7	3	20
Canada	9	7	11	27
Uzbekistan	8	2	3	13
Hungary	6	7	6	19
Spain	5	4	9	18
Sweden	4	4	3	11

Ici, nous avons les différents types de médailles gagnées par chacun des pays, triés par celui qui en a gagné le + et par ordre de niveau de médaille.

Répartition par sport

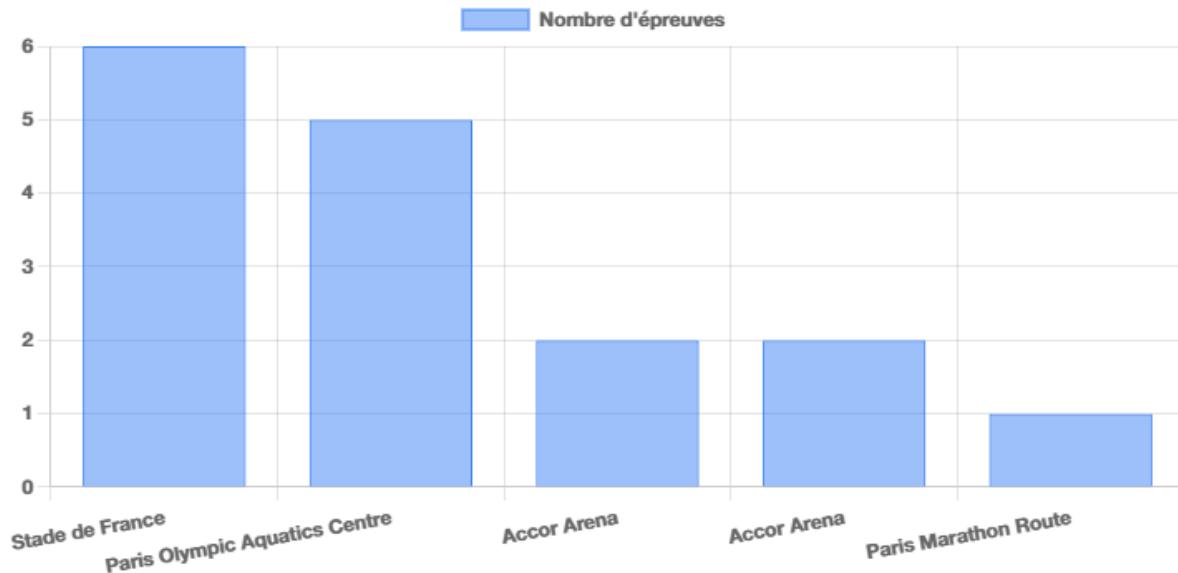
[Voir les statistiques](#)



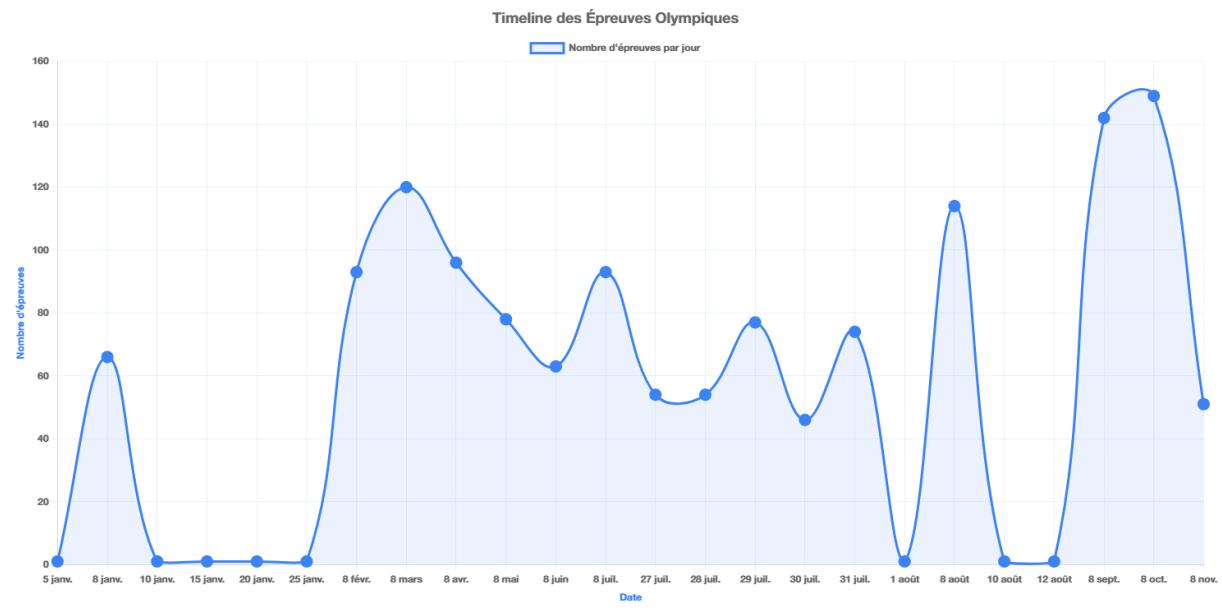
Voici un camembert qui permet de voir la distribution des participants, des événements, ou d'autres entités (comme des médailles ou des performances) par discipline sportive.

Répartition des épreuves par site

Voir les statistiques



Ici, on peut voir la répartition des épreuves par site (NOTE : malheureusement, en raison du manque de mentions dans le dataset des sites sur lesquels se sont déroulés les épreuves, nous n'avons eu que très peu de résultats, comparés à la quantité de données dont nous disposons.)



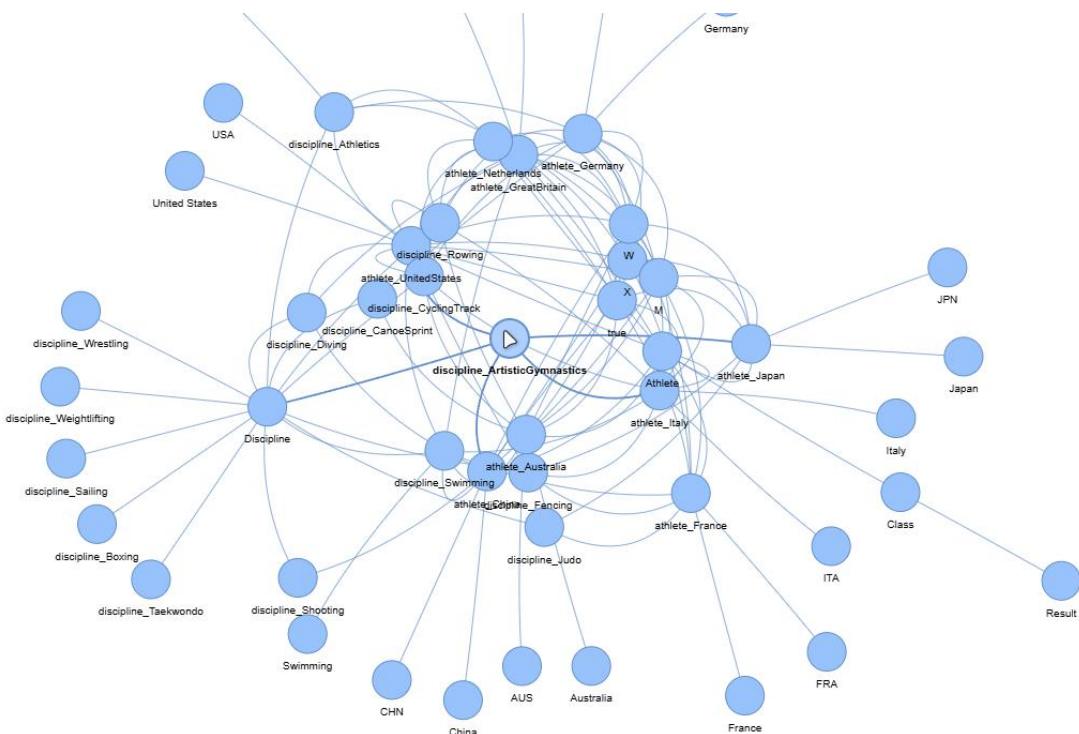
Nous avons aussi affiché le nombre d'épreuves par jour qui se sont déroulées durant toute la période.

Météo sur les sites

[Voir la météo](#)

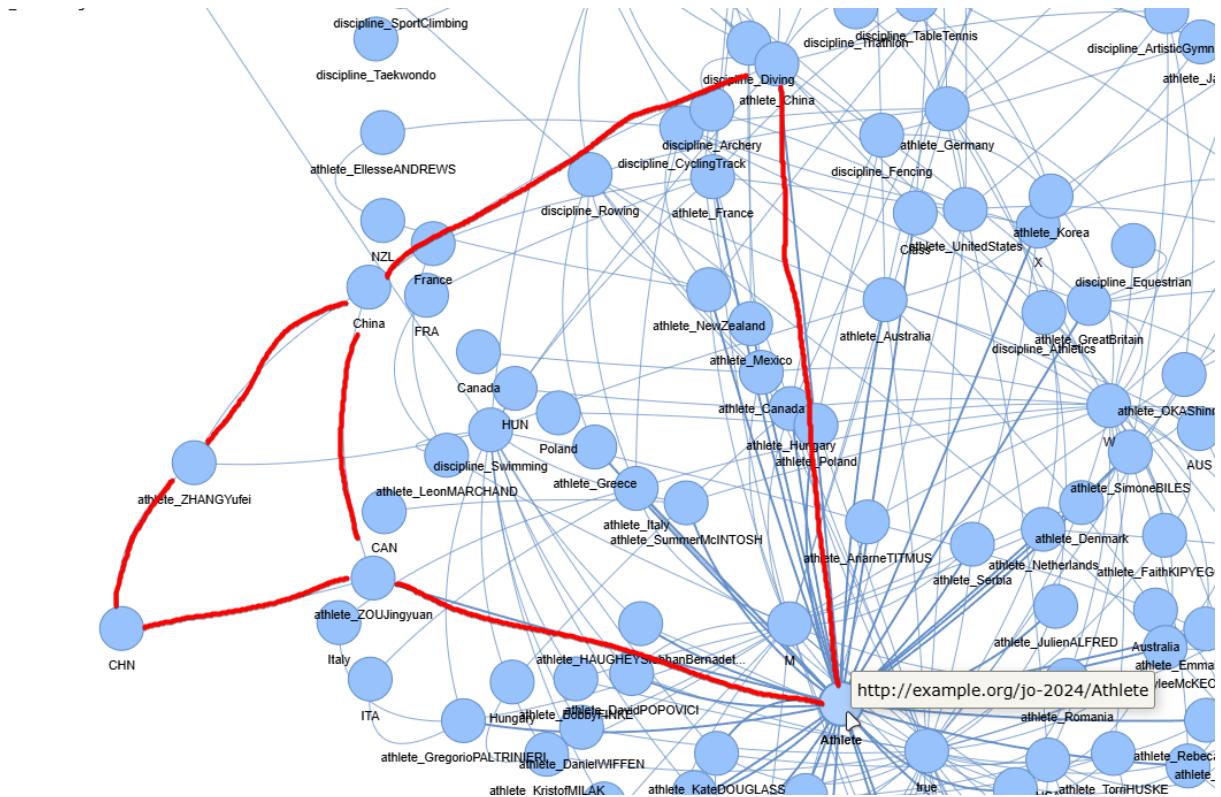
Site	Description	Température (°C)
Accor Arena	light rain	9.8
Accor Arena	light rain	9.8
Paris Marathon Route	light rain	9.8
Paris Olympic Aquatics Centre	light rain	9.8
Stade de France	light rain	9.8

Et enfin, notre visualisation « préférée », qui en raison du bug dans le docker de M. Michel quant au traitement des « . » et de l'encoding en « %20 », qui affiche la météo par lieu. Comme tous se situent à Paris et que l'arrondissement des coordonnées GPS s'est effectué à l'unité, ils ont évidemment tous les mêmes valeurs (mais le microservice fonctionne donc correctement !)



Et enfin, à l'aide du script `03_generate_graph.py`, nous avons réalisé l'affichage **PARTIEL** de notre graphe de connaissances. On peut voir que les liaisons font du sens et qu'elles respectent l'ontologie établie. Nous nous permettons d'insister sur « affichage partiel » car ici, nous n'avons affiché que 100

triplets sur les 15 000 générés après enrichissement, pour des raisons de lisibilité et de temps de génération conséquent. Autrement, voici une partie du graphe généré presque complètement :



Les liaisons sont fortes, les liens sont correctement établis (voir ligne rouge comme exemple).

Conclusion

Ce projet démontre la puissance des technologies du Web Sémantique pour la gestion et l'exploitation des données des Jeux Olympiques 2024. Notre architecture, combinant une ontologie personnalisée, des règles d'inférence SPARQL et des contraintes SHACL, a permis de créer un graphe de connaissances riche et cohérent. L'enrichissement des données à partir de sources hétérogènes (CSV, articles de presse, API météo) et leur liaison avec DBpedia et Wikidata ont considérablement augmenté la valeur informationnelle du graphe, passant de 355 à plus de 15 000 triplets.

Les visualisations développées démontrent le potentiel d'exploitation de ces données interconnectées, permettant des analyses variées allant de la distribution des médailles par pays à la répartition des épreuves dans le temps et l'espace. Bien que certaines limitations techniques aient été rencontrées, notamment dans le traitement des coordonnées GPS pour les données météorologiques, les solutions mises en place ont permis de maintenir la fonctionnalité du système.

Ce travail ouvre la voie à de nombreuses perspectives d'amélioration, comme l'intégration de nouvelles sources de données en temps réel pendant les Jeux, l'enrichissement des visualisations, ou encore l'exploitation plus poussée des capacités d'inférence pour des analyses prédictives.