

Semantic Web 入門

<https://www.tohoho-web.com/ex/semantic-web.html>

https://www.soumu.go.jp/main_sosiki/joho_tsusin/policyreports/chousa/info_frontier/pdf/050412_2_4.pdf

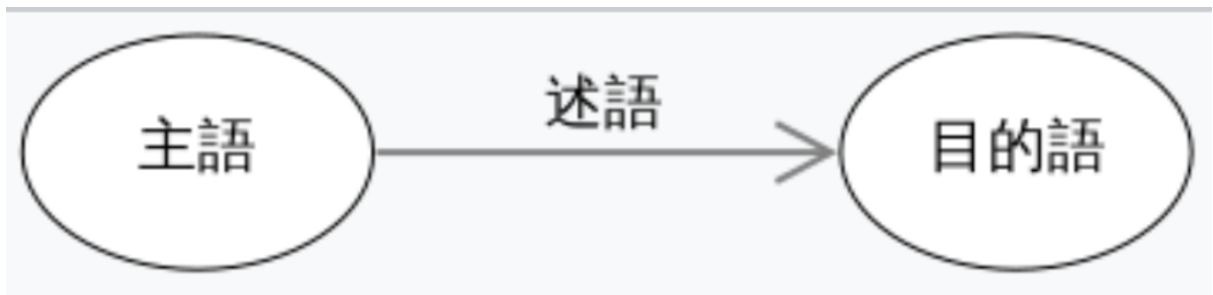
[Semantic Web](#)

Semantic Webとは？

- Semanticとは？
 - 意味論という意味
- Semantic Webとは？
 - 1998年に[Tim berners-lee](#)が提唱した
 - Webサイト上の情報などにMeta Dataを付与して意味(semantics)を持たせることで、コンピュータが、より豊かで正確な情報を分析・整理・再利用・提供できるようになることを目指した思想・構想・技術・プロジェクトを表す

Meta Dataとは？

- メタデータとは「データに付加するデータ」という意味で、例えば、`<div>山田太郎</div>` というデータに対して、`<div _property="creator">山田太郎</div>` というデータを付加することで、山田太郎が単なる文字列ではなく、作成者であることを明示します。
- creator などの **意味・語彙・スキーマ** を定義するものと、定義されたものをどの様に表現するかの **記法・文法・言語** に大別されます。
 - 記法としてはRDFが用いられる



RDFとは何か

- [RSS](#)やDBpediaで使用されている
- 主語(Subject),述語(Predicate),目的語(Object)の3要素をトリプルと呼び、すべてのデータをこの関係で記述する
 - 例えば [山田太郎は(主語)]-[名乗ります(述語)]-[山田太郎と(目的語)] など。
- 例えば [山田太郎は(主語)]-[住んでいます(述語)]-[東京都港区に(目的語)] など。
- 例えば [サザンオールスターズは(主語)]-[バンドメンバーを含んでいます(述語)]-[桑田佳祐を(目的語)] など。
- 主語、述語、目的語はすべて URI で表します。
- RDF 自体はフレームワークで、このフレームワークに従った具体的な表現形式として、[N-Triples](#), [Turtle](#), [TriG](#), [N-Quads](#), [RDFa](#), [RDF/XML](#), [JSON-LD](#) があります。

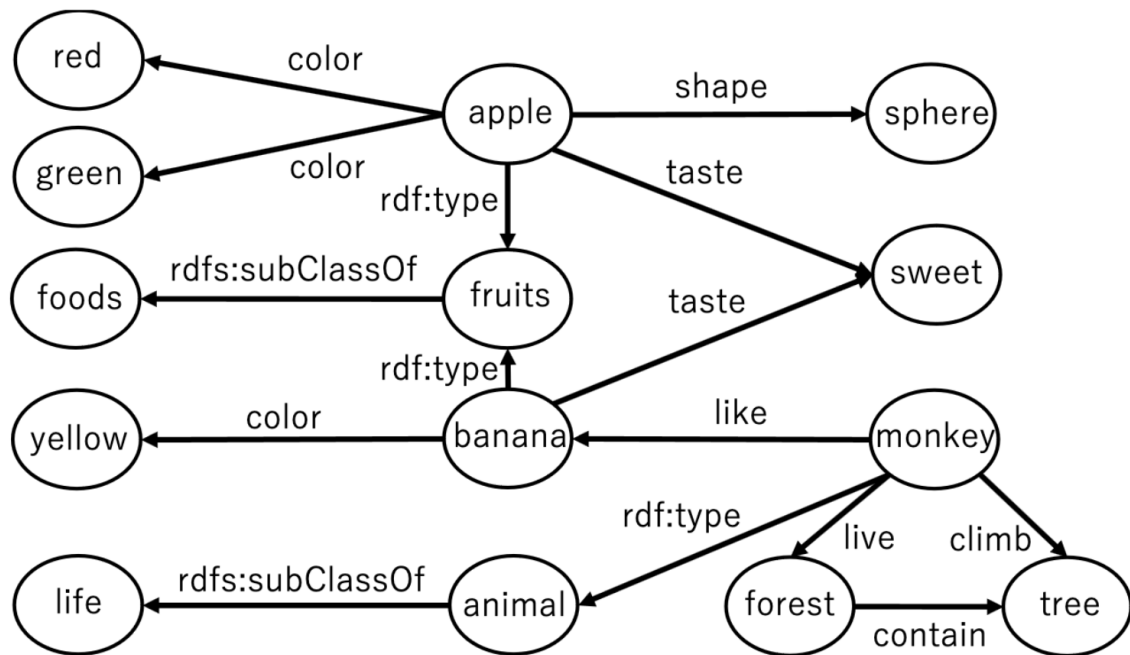


図 1: RDF グラフの例

Schema.org

- Google, Microsoft, Yahoo などの共同で設立された方と属性の集合を定義しているコミュニティ活動
- 例えば、Person という型(type)は、name(氏名)、address(住所)、gender(性別)、birthDate(誕生日)などの属性(property)を持ちます。
- name(氏名)は Text、address(住所) は Text または PostalAddress といった、属性もまた型を持ちます。
- 型や属性は <http://schema.org/Person> や <http://schema.org/name> などの URI で表されます。

Schema.orgの定義の例

Schema.orgの定義の例

Person (type):

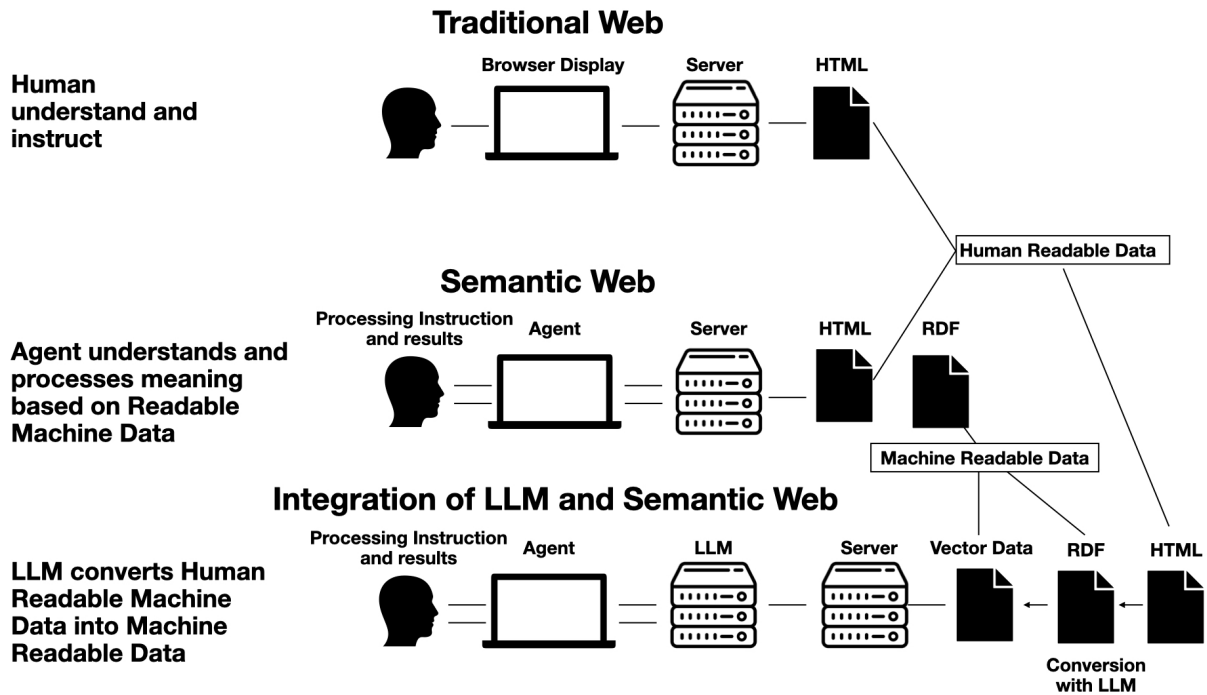
- name (Text)
- address (Text or PostalAddress)
- gender (Text or GenderType)
- birthDate (Date)

SPARQL

- SPARQL Protocol And RDF Query Language の略です。Linux 系で好まれる再帰的頭字語です。
- [DBpedia](#) などの RDFデータに対して、SQL ライクな文法でクエリをかけることができます。

LLMとRDFの組み合わせが良いのではないかと考えてみる

- LLMとの今後



- 人間がHTMLを読む時代->コンピュータがRDFを読む時代->LLMが自然言語をRDFに変換し、RDFをLLMが解釈する時代
- 自然言語からLLMがRDFを生成し、生成されたRDFを元にLLMがOutPutする未来
 - これまでのRDF財産を無駄にしない
 - より正確な情報を情報圧縮して渡せる
 - 個人的にはJsonでもいいのではと思う。RDFというフォーマットもベクトルデータに内包されると思う

新幹線の路線で試してみる

- 北陸新幹線のダイヤ

種別	東京駅 発車時刻	東京 駅	上野 駅	大宮 駅	熊谷 駅	本庄 早稲田 駅	高崎 駅	安中 榛名 駅	軽井 沢 駅	佐久 平 駅	上田 駅	長野 駅	飯山 駅	上越 妙高 駅	糸魚 川 駅	黒部 宇奈月温泉 駅	富山 駅	新高岡 駅	金沢 駅	終着
あさま	04分	●	●	●	●	●	○	●	●	●	◎									長野
はくたか	24分	●	●	●	→	→	●	→	△	△	△	◎	○	●	●	●	●	●	●	金沢
あさま◆	32分	●	●	●	→	→	●	→	●	●	●	●								長野
かがやき◆	52分	●	●	●	→	→	→	→	→	→	→	●	→	→	→	→	●	→	●	金沢
つぎ	(富山発)17分																●	●	●	金沢

- 質問：北陸新幹線の"かがやき"が停車する駅を全て教えてください。北陸新幹線の他の種別の停車駅は答えないでください。
- ChatGPT4の場合
 - <https://chat.openai.com/share/8422eafa-7b37-4373-b185-91c3ff2ef9b1>



You

「北陸新幹線の"かがやき"が停車する駅を全て教えてください。北陸新幹線の他の種別の停車駅は答えなくてください。」

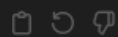


ChatGPT

北陸新幹線の「かがやき」が停車する駅は、以下の通りです。

- 東京駅
- 上野駅
- 大宮駅
- 熊谷駅
- 高崎駅
- 安中榛名駅
- 軽井沢駅
- 佐久平駅
- 上田駅
- 長野駅
- 飯山駅
- 上越妙高駅
- 糸魚川駅
- 黒部宇奈月温泉駅
- 富山駅
- 新高岡駅
- 金沢駅

これらは、北陸新幹線の「かがやき」が停車する主要な駅です。なお、時刻や運行日によっては、一部の駅を通過することもあるため、利用の際は最新の時刻表を確認してください。



- GPT 3.5の場合

- <https://chat.openai.com/share/9a23f759-54b3-4df8-9647-25b5a5356957>



You

北陸新幹線の「かがやき」が停車する駅を全て教えてください。北陸新幹線の他の種別の停車駅は答えしないでください。



ChatGPT

北陸新幹線の「かがやき」は、以下の駅に停車します。

1. 東京駅
2. 上野駅
3. 大宮駅
4. 本庄駅
5. 前橋駅
6. 軽井沢駅
7. 長野駅
8. 米原駅
9. 金沢駅
10. 敦賀駅
11. 新大阪駅

以上が北陸新幹線「かがやき」の停車駅です。





北陸新幹線の"かがやき"が停車する駅を全て教えてください。北陸新幹線の他の種別の停車駅は答えないでください。

回答案を表示



北陸新幹線「かがやき」の停車駅

北陸新幹線「かがやき」は、東京駅から金沢駅までを最速で結ぶ速達列車です。停車駅は以下の通りです。

全列車が停車する駅

- 東京駅
- 上野駅 (一部列車のみ)
- 大宮駅
- 長野駅
- 富山駅
- 金沢駅

一部列車のみ停車する駅

- 福井駅
- 加賀温泉駅
- 芦原温泉駅
- 小松駅
- 越前たけふ駅

詳細情報

- 上野駅は、早朝・深夜の一部列車のみ停車します。
- 福井駅、加賀温泉駅、芦原温泉駅、小松駅、越前たけふ駅は、下り列車の一部と上り列車の全てが停車します。
- 停車駅は、列車によって異なる場合がありますので、事前に確認することをおすすめします。

参考情報

- 北陸新幹線の時刻表: [無効な URL を削除しました]
- 北陸新幹線の停車駅一覧: [無効な URL を削除しました]





北陸新幹線の"かがやき"が停車する駅を全て教えてください。北陸新幹線の他の種別の停車駅は答えないでください。



MODIFY

北陸新幹線の「かがやき」が停車する駅は、東海新幹線と共同で利用できる「東海新幹線帯の駅」を以下に紹介します。

- 1. 東海新幹線帯
 - ・ 東海新幹線と共同で運行されており、かがやきも同様の路線で停車しています。
 - ・ 東海新幹線帯の駅は、東海新幹線と同様、京町地方鉄道と連絡可能です。

東海新幹線帯の駅 (かがやきが停車する駅):

- 1. 名古屋
- 2. Mikawa-Anjo
- 3. Okazaki
- 4. Gamagori
- 5. Toyohashi

上記の駅以外の北陸新幹線の駅では、かがやきは停車していません。ご参考まで。



End to End Time: 1.02s | Model: Mixtral 8x7B-32k

- 結果
 - 沿線に含まれない駅の場合は-10

停車駅	正解	GPT4	GPT3.5	Gemini	Groq
東京	O	O	O	O	
上野	O	O	O	O	
大宮	O	O	O	O	
熊谷		O			
本庄早稲田			X		
高崎		O			
安中榛名		O			
軽井沢		O	O		
佐久間平		O			
上田		O			
長野	O	O	O	O	
飯山		O			
上越妙高		O			
糸魚川		O			
黒部雨奈月温泉		O			
富山	O	O		O	
新高岡		O			
金沢	O	O	O	O	

停車駅	正解	GPT4	GPT3.5	Gemini	Groq
正解 - 誤答	6	6 - 11 = -5	6 - 1 -(5 * 10) = -45	6 - 0 = 6	6 - (5*10) = -50

- 北陸新幹線のRDF
 - <https://www.wikidata.org/wiki/Q1037409>

RDFデータをChatGPT 4に生成してもらう

- 北陸新幹線のwikiをChatGPT 4にRDFに変換してもらった。

以下は、提供された情報に基づいて作成した全ての停車駅に関するRDFデータです。このデータはTurtle形式で記述されており、各列車種別と停車駅、終着駅に関する情報を含んでいます。

```
@prefix ex: <http://example.org/railway/> .
@prefix st: <http://example.org/station/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

st:上野 a rdfs:Class ;
    rdfs:label "上野駅" .

st:富山 a rdfs:Class ;
    rdfs:label "富山駅" .

st:新高岡 a rdfs:Class ;
    rdfs:label "新高岡駅" .

st:佐久平 a rdfs:Class ;
    rdfs:label "佐久平駅" .

st:上越妙高 a rdfs:Class ;
    rdfs:label "上越妙高駅" .

st:上田 a rdfs:Class ;
    rdfs:label "上田駅" .

st:軽井沢 a rdfs:Class ;
    rdfs:label "軽井沢駅" .

st:東京 a rdfs:Class ;
    rdfs:label "東京駅" .

st:飯山 a rdfs:Class ;
    rdfs:label "飯山駅" .

st:熊谷 a rdfs:Class ;
    rdfs:label "熊谷駅" .

st:安中榛名 a rdfs:Class ;
    rdfs:label "安中榛名駅" .

st:高崎 a rdfs:Class ;
    rdfs:label "高崎駅" .

st:長野 a rdfs:Class ;
    rdfs:label "長野駅" .

st:黒部宇奈月温泉 a rdfs:Class ;
    rdfs:label "黒部宇奈月温泉駅" .

st:大宮 a rdfs:Class ;
    rdfs:label "大宮駅" .
```



```

st:本庄早稲田 a rdfs:Class ;
    rdfs:label "本庄早稲田駅" .

st:糸魚川 a rdfs:Class ;
    rdfs:label "糸魚川駅" .

st:金沢 a rdfs:Class ;
    rdfs:label "金沢駅" .

ex:あさま a rdfs:Class ;
    rdfs:label "あさま" ;
    ex:departureTime "08分"^^xsd:string ;
    ex:stopsAt st:東京, st:上野, st:大宮, st:熊谷, st:本庄早稲田, st:高崎, st:安中榛名, st:軽井沢, st:佐久
平, st:上田, st:長野 ;
    ex:terminatesAt st:長野 .

ex:かがやき a rdfs:Class ;
    rdfs:label "かがやき" ;
    ex:departureTime "24分"^^xsd:string ;
    ex:stopsAt st:東京, st:上野, st:大宮, st:金沢 ;
    ex:terminatesAt st:金沢 .

ex:はくたか a rdfs:Class ;
    rdfs:label "はくたか" ;
    ex:departureTime "56分"^^xsd:string ;
    ex:stopsAt st:東京, st:上野, st:大宮, st:高崎, st:軽井沢, st:佐久平, st:上田, st:長野, st:飯山, st:上
越妙高, st:

糸魚川, st:黒部宇奈月温泉, st:富山, st:新高岡, st:金沢 ;
    ex:terminatesAt st:金沢 .

ex:つるぎ a rdfs:Class ;
    rdfs:label "つるぎ" ;
    ex:departureTime "富山発15分"^^xsd:string ;
    ex:stopsAt st:富山, st:金沢 ;
    ex:terminatesAt st:金沢 .

```

このRDFデータは、各列車の種別（あさま、かがやき、はくたか、つるぎ）ごとに、発車時刻、停車駅、終着駅を記述しています。各駅もリソースとして定義され、ラベル（駅名）が付けられています。このデータを用いることで、列車と駅の関係を表現するセマンティックウェブアプリケーションや情報検索システムを構築できます。


生成されたRDFを元に、再度答えてもらう

- 生成してもらったRDF
 - Chat履歴:<https://chat.openai.com/share/76353c5b-01f1-4fa7-9ad5-437b6486f571>

You

以下の表から、各種別の停車駅のRDFファイルを生成してください。

種別	東京駅	上野駅	大宮駅	熊谷駅	本庄早稻田駅	高崎駅	安中榛名駅	軽井沢駅	佐久平駅	上田駅	長野駅	飯山駅	上越妙高駅	糸魚川駅	黒部宇奈月温泉駅	富山駅	新高岡駅	金沢駅	終着
あさま	08分	●	●	●	●	●	○	●	●	●	●								長野
かがやき◆	24分	●	●	●	→	→	→	→	→	→	●	→	→	→	→	●	→	●	金沢
あさま◆	40分	●	●	●	→	→	●	→	●	●	●	●							長野
はくたか	56分	●	●	●	→	→	●	→	△	△	△	●	○	●	●	●	●	●	金沢
つるぎ	(富山発)15分																		金沢

< 2 / 2 > 

```
@prefix ex: <http://example.org/railway/> .
@prefix st: <http://example.org/station/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

st:上野 a rdfs:Class ;
    rdfs:label "上野駅" .

st:富山 a rdfs:Class ;
    rdfs:label "富山駅" .

st:新高岡 a rdfs:Class ;
    rdfs:label "新高岡駅" .

st:佐久平 a rdfs:Class ;
    rdfs:label "佐久平駅" .

st:上越妙高 a rdfs:Class ;
    rdfs:label "上越妙高駅" .

st:上田 a rdfs:Class ;
    rdfs:label "上田駅" .

st:軽井沢 a rdfs:Class ;
    rdfs:label "軽井沢駅" .

st:東京 a rdfs:Class ;
    rdfs:label "東京駅" .

st:飯山 a rdfs:Class ;
    rdfs:label "飯山駅" .

st:熊谷 a rdfs:Class ;
    rdfs:label "熊谷駅" .

st:安中榛名 a rdfs:Class ;
    rdfs:label "安中榛名駅" .

st:高崎 a rdfs:Class ;
    rdfs:label "高崎駅" .

st:長野 a rdfs:Class ;
    rdfs:label "長野駅" .

st:黒部宇奈月温泉 a rdfs:Class ;
    rdfs:label "黒部宇奈月温泉駅" .

st:大宮 a rdfs:Class ;
    rdfs:label "大宮駅" .

st:本庄早稲田 a rdfs:Class ;
    rdfs:label "本庄早稲田駅" .

st:糸魚川 a rdfs:Class ;
    rdfs:label "糸魚川駅" .

st:金沢 a rdfs:Class ;
    rdfs:label "金沢駅" .

ex:あさま a rdfs:Class ;
    rdfs:label "あさま" ;
    ex:departureTime "08分"^^xsd:string ;
    ex:stopsAt st:東京, st:上野, st:大宮, st:熊谷, st:本庄早稲田, st:高崎, st:安中榛名, st:軽井沢,
    st:佐久平, st:上田, st:長野 ;
    ex:terminatesAt st:長野 .
```

```
ex:かがやき a rdfs:Class ;
  rdfs:label "かがやき" ;
  ex:departureTime "24分"^^xsd:string ;
  ex:stopsAt st:東京, st:上野, st:大宮, st:金沢 ;
  ex:terminatesAt st:金沢 .

ex:はくたか a rdfs:Class ;
  rdfs:label "はくたか" ;
  ex:departureTime "56分"^^xsd:string ;
  ex:stopsAt st:東京, st:上野, st:大宮, st:高崎, st:軽井沢, st:佐久平, st:上田, st:長野, st:飯山,
  st:上越妙高, st:糸魚川, st:黒部宇奈月温泉, st:富山, st:新高岡, st:金沢 ;
  ex:terminatesAt st:金沢 .

ex:つるぎ a rdfs:Class ;
  rdfs:label "つるぎ" ;
  ex:departureTime "富山発15分"^^xsd:string ;
  ex:stopsAt st:富山, st:金沢 ;
  ex:terminatesAt st:金沢 .
```

- RDFを読み込んでもらって、再度答えてもらう
 - **与えたPrompt:** 以下のRDFデータを元に、北陸新幹線の"かがやき"が停車する駅を全て教えてください。北陸新幹線の他の種類の停車駅は答えないでください。+上記RDF
- 結果
- ChatGPT4

北陸新幹線の"かがやき"が停車する駅は以下の通りです。

- 東京駅
- 上野駅
- 大宮駅
- 金沢駅

- ChatGPT3.5

北陸新幹線の"かがやき"が停車する駅は次の通りです：

- 東京駅（st:東京）
- 上野駅（st:上野）
- 大宮駅（st:大宮）
- 金沢駅（st:金沢）

- Gemini Pro 1.5

RDFデータに基づき、北陸新幹線「かがやき」が停車する駅は以下の通りです。

1. 東京駅
2. 上野駅
3. 大宮駅
4. 金沢駅

****その他の停車駅は含まれていません。****

****注意**:**

* 上記の情報は、提供されたRDFデータに基づいています。実際の運行状況は異なる場合がありますので、最新情報をご確認ください。

* 北陸新幹線には、「かがやき」以外にも「あさま」、「はくたか」、「つるぎ」などの種別があります。これらの停車駅は、「かがやき」とは異なります。

ご参考になれば幸いです。

- Groq

Sure! Here are the stations where the "かがやき" train stops, according to the provided RDF data:

* 東京 (Tokyo)
* 上野 (Ueno)
* 大宮 (Omiya)
* 金沢 (Kanazawa)

Note that the "かがやき" train does not stop at 新高岡 (Niigata) or 富山 (Toyama), even though they are listed as stops for other trains in the data.

結果

- (r = RDF)

停車駅	正解	rGPT4	rGPT3.5	rGemini	rGroq
東京	O	O	O	O	O
上野	O	O	O	O	O
大宮	O	O	O	O	O
熊谷					
本庄早稲田					
高崎					
安中榛名					
軽井沢					
佐久間平					
上田					
長野	O				
飯山					
上越妙高					
糸魚川					
黒部雨奈月温泉					
富山	O				
新高岡					
金沢	O	O	O	O	O
正解 - 誤答	6 - 0 = 6	6 - 2 = 4	6 - 2 = 4	6 - 2 = 4	6 - 2 = 4

わかったこと

- 適切なRDFを与えれば低いモデルでも正答率が高い。

- 適切なRDFをLLMで記述できる可能性は高い
 - 人間が一応見て確認するか,GPT5が誤りもなくRDFに転換してくれる未来を待つのもあり

考えられること

- RDFである必要はないかもしれない
 - JSONとかで圧縮してもいいかもしれない
- 低い情報量のLLMが高い情報を持つLLMに対し、その都度RDFを作成するRequestを行う。高い性能を持つLLMはその都度Wikidataなどを参照し、過去のRDFデータを参照しつつ、不足していればその都度作成する。最後に、RDFをデータベースに保存し、低い性能のLLMでも高精度な回答を行える未来があるかもしれない

普及例

- Google リッチスニペット

Google search results for "ディナー" (Dinner) in the area of 武蔵野市 (Musashino City). The results show three restaurant listings: La Prima Pagina, Restaurant COZY, and オステリア エジリオサーラ. Each listing includes a photo, name, rating, address, and a snippet of a review. A map on the right shows the location of these restaurants in the Musashino area.

- RSSによるヘッドラインの配信

```
<?xml version="1.0" encoding="UTF-8"?><rss version="2.0"
  xmlns:content="http://purl.org/rss/1.0/modules/content/"
  xmlns:fw="http://wellformedweb.org/CommentAPI/"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:atom="http://www.w3.org/2005/Atom"
  xmlns:sy="http://purl.org/rss/1.0/modules/syndication/"
  xmlns:slash="http://purl.org/rss/1.0/modules/slash/"
  >

<channel>
  <title>「一言」で理解するIT用語辞典</title>
  <atom:link href="https://it-dictionary.info/feed/" rel="self" type="application/rss+xml" />
  <link>https://it-dictionary.info</link>
  <description></description>
  <lastBuildDate>Thu, 01 Sep 2022 09:02:39 +0000</lastBuildDate>
  <language>ja</language>
  <sy:updatePeriod>
    hourly </sy:updatePeriod>
  <sy:updateFrequency>
    1 </sy:updateFrequency>
  <generator>https://wordpress.org/?v=6.0.3</generator>

  <image>
    <url>https://it-dictionary.info/wp-content/uploads/2022/10/c013b3821b45793ec7c990adba918f17.png</url>
    <title>「一言」で理解するIT用語辞典</title>
    <link>https://it-dictionary.info</link>
    <width>32</width>
    <height>32</height>
  </image>

  <item>
    <title>ITサービスマネジメント（ITSM）とは？</title>
    <link>https://it-dictionary.info/contents/itsm</link>
    <dc:creator><![CDATA[makisaka]]></dc:creator>
    <pubDate>Sat, 20 Aug 2022 00:23:14 +0000</pubDate>
    <category><![CDATA[マネジメント]]></category>
    <guid isPermalink="false">https://it-dictionary.info/?p=207</guid>
    <description><![CDATA[<p></p>ITサービスマネジメント（ITSM: IT Service Management）とは、ITシステムをユーザーに提供するために必要となる運用・改善・管理などを体系的に実施すること。ITサービスマネジメントの必要性 当然な [48230:]]></description>
```