

Chatty-Gen Supplementary Materials

1 Prompts

1.1 Choosing Entity Predicate (I_{EL})

```
1 # Instruction:
2 Given the specified node type and its associated
3 predicates, choose a suitable predicate to be used to
4 extract labels for this type.
5
6 # Input:
7 node type:
8 predicates:
9
10 # Response:
```

Listing 1: Choosing representative predicate for type prompt

1.2 Generating Self-Contained questions (I_{IQ})

```
1 # Instruction:
2 Generate a list of questions based on the given entity
3 and its subgraph. The subgraph is represented as a
4 list of triples. Each question must ask about a fact
5 from the triples in the subgraph and must fall into
6 one of the following categories: list, count, boolean,
7 wh (open-ended), or date-related questions. Each
8 question must include the entity. Each question must
9 be answerable solely from the information in the
10 provided subgraph without explicitly mentioning it.
11
12 # Input:
13 Entity:
14 Subgraph:
15 number of questions:
16
```

```
17 # Response:
```

Listing 2: Generating independent questions prompt

1.3 Generating Self-Contained Questions With Triples (I'_{IQ})

```
1 # Instruction:
2 {Question Generation Prompt}
3 For each question, choose the triples from the input
4 subgraph which was used to generate the question.
5 Return both the question and the exact triple from the
6 subgraph that it was based on.
7
8 # Input:
9 Entity:
10 Subgraph:
11 number of questions:
12
13 # Response:
```

Listing 3: Generating Independent Questions Prompt with Triples

1.4 Generating SPARQL Queries (I_{SQ})

```
1 # Instruction:
2 Given a question and set of triples used to generate
3 this question. Create the SPARQL query representing
4 the question. Do not include the answer in the query.
5
6 # Input:
7 Question:
8 Triples:
9
10 # Response:
```

Listing 4: Generating SPARQL query from Question and Triples

1.5 Generating Dialogue (I_{DG})

```
1 # Instruction:
2 Given an entity and a set of questions focused on this
3 entity, choose the appropriate pronoun that refers to
4 it. Replace the entity with its pronoun in the
5 questions and return the modified questions. Ensure
6 that the modified questions do not contain the
```

```

7 original entity and that the pronoun used in the
8 modified questions is contextually appropriate and
9 grammatically correct.
10
11 # Input:
12 Questions:
13 Seed Entity:
14
15 # Response:

```

Listing 5: Generating SPARQL query from Question and Triples

1.6 A Single Prompt Approach (I_S)

```

1 # Instruction:
2 Generate a set of questions, a dialogue and sparqls
3 based on the provided entity and its subgraph. The
4 subgraph is represented as a varied list of triples.
5 Each question should be a fact from the triples in the
6 subgraph and fall into one of the following categories
7 : list, count, boolean, wh (open-ended), or
8 date-related questions. Each question should have the
9 entity and be answerable solely from the information
10 in the provided subgraph without explicitly mentioning
11 it. For the generated questions, generate a
12 corresponding dialogue where the first is standalone
13 and subsequent questions with replaced entity with its
14 pronoun. And a list of SPARQL queries that retrieves
15 answers. Return the following: questions, dialogue,
16 and SPARQL queries.
17
18 # Input:
19 Entity:
20 Subgraph:
21 number of questions:
22
23 # Response:

```

Listing 6: Generating Benchmark (standalone questions-SPARQL queries-Dialogue questions) using a single prompt

2 Examples

Example YAGO:

Entity: http://yago-knowledge.org/resource/A._S._M._Hutchinson

1. Where was Arthur Stuart-Menteth Hutchinson born?
2. What alternate names are known for him?
3. What was his occupation?
4. When did he die?
5. Where did he pass away?

Example DBLP:

Entity: <https://dblp.org/rec/phd/Turing38>

1. What type of work is 'Systems of Logic Based on Ordinals.' classified as in Bibtex?
2. Who authored it?
3. What is the DOI of it?",
4. In which year was it published?
5. How many creators were involved in it?

Figure 1: Examples of generated questions by Chatty-Gen for KGs of different domains.