Neural Machine Translation for Query Construction and Composition

Tommaso Soru [1]   **Edgard Marx** [1]   **André Valdestilhas** [1]   **Diego Esteves** [2]   **Diego Moussallem** [1]   **Gustavo Publio** [1]

Abstract

Research on question answering with knowledge base has recently seen an increasing use of deep architectures. In this extended abstract, we study the application of the neural machine translation paradigm for question parsing. We employ a sequence-to-sequence model to learn graph patterns in the SPARQL graph query language and their compositions. Instead of inducing the programs through question-answer pairs, we expect a semi-supervised approach, where alignments between questions and queries are built through templates. We argue that the coverage of language utterances can be expanded using late notable works in natural language generation.

1. Introduction

Question Answering with Knowledge Base (KBQA) parses a natural-language question and returns an appropriate answer that can be found in a knowledge base. Today, one of the most exciting scenarios for question answering is the Web of Data, a fast-growing distributed graph of interlinked knowledge bases which comprises more than 100 billions of edges (McCrae et al., 2018). Question Answering over Linked Data (QALD) is a subfield of KBQA aimed at transforming utterances into SPARQL queries (Lopez et al., 2013). Being a W3C standard, SPARQL features a high expressivity (Prud'hommeaux et al., 2006) and is by far the most used query language for Linked Data.

Among traditional approaches to KBQA, Bao et al. (2014) proposed question decomposition and Statistical Machine Translation to translate sub-questions into triple patterns. The method however relies on entity detection and struggles in recognizing predicates by their contexts (e.g., *play* in a film or a football team). In the last years, several methods based on neural networks have been devised to
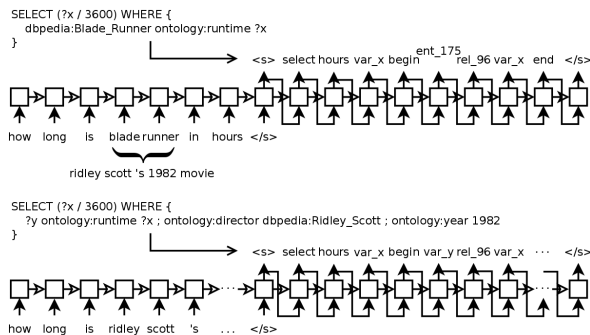

Figure 1. Utterances are translated into SPARQL queries encoded as sequences of tokens. Using complex surface forms leads to more graph patterns. We aim at learning these compositions.

tackle the KBQA problem (Liang et al., 2016; Hao et al., 2017; Lukovnikov et al., 2017; Sorokin & Gurevych, 2017). We study the application of the Neural Machine Translation paradigm for question parsing using a sequence-to-sequence model within an architecture dubbed Neural SPARQL Machine, previously introduced in Soru et al. (2017). Similarly to Liang et al. (2016), we employ a sequence-to-sequence model to learn query expressions and their compositions. Instead of inducing the programs through question-answer pairs, we expect a semi-supervised approach, where alignments between questions and queries are built through templates. Although query induction can save a considerable amount of supervision effort (Liang et al., 2016; Zhong et al., 2017), a pseudo-gold program is not guaranteed to be correct when the same answer can be found with more than one query (e.g., as the *capital* is often the *largest city* of a country, predicates might be confused). On the contrary, our proposed solution relies on manual annotation and a weakly-supervised expansion of question-query templates.

2. Neural SPARQL Machines

Inspired by the *Neural Programmer-Interpreter* pattern by (Reed & De Freitas, 2015), a Neural SPARQL Machine is composed by three modules: a *generator*, a *learner*, and an *interpreter* (Soru et al., 2017). We define a *query template* as an alignment between a natural language question and its respective SPARQL query, with entities replaced by placeholders (e.g., *"where is <A> located in?"*). The gen-

---

[1]AKSW, University of Leipzig, Leipzig, Germany [2]SDA, Bonn University, Bonn, Germany. Correspondence to: Tommaso Soru <tsoru@informatik.uni-leipzig.de>.

Table 1. Experiments on a DBpedia subset about movies with different SPARQL encodings and settings.

Encoding	Description	Test BLEU	Accuracy	Runtime	Convergence
v1	1:1 SPARQL encoding	80.89%	22.33%	1h02:01	13,000
v1.1	Improved consistency	80.61%	22.33%	1h21:21	17,000
v2	Added templates with > 1 placeholders	89.69%	91.04%	1h59:10	22,000
v2.1	Encoding fix (double spaces removed)	98.40%	91.05%	1h47:11	20,000
v3	Shortened SPARQL sequences	99.28%	94.82%	1h12:07	25,000
v4	Added direct entity translations	99.29%	93.69%	1h23:00	20,000

erator takes query templates as input and creates the training dataset, which is forwarded to the learner. The learner takes natural language as input and generates a sequence which encodes a SPARQL query. Here, a recurrent neural network based on (Bidirectional) Long Short-Term Memories (Hochreiter & Schmidhuber, 1997) is employed as a sequence-to-sequence translator (see example in Figure 1). The final structure is then reconstructed by the interpreter through rule-based heuristics. Note that a sequence can be represented by any LISP S-expression; therefore, alternatively, sentence dependency trees can be used to encode questions and ARQ algebra (Seaborne, 2010) can be used to encode SPARQL queries.

Neural SPARQL Machines do not rely on entity linking methods, since entities and relations are detected within the query construction phase. External pre-trained word embeddings help deal with vocabulary mismatch. Knowledge graph jointly embedded with SPARQL operators (Wang et al., 2014) can be utilized in the target space. A curriculum learning (Bengio et al., 2009) paradigm can learn graph pattern and SPARQL operator composition, in a similar fashion of Liang et al. (2016). We argue that the coverage of language utterances can be expanded using techniques such as Question (Abujabal et al., 2017; Elsahar et al., 2018; Abujabal et al., 2018) and Query Generation (Zafar et al., 2018) as well as Universal Sentence Encoders (Cer et al., 2018). Another problem is the disambiguation between entities having the same surface forms. Building on top of the DBtrends approach (Marx et al., 2016), we force the number of occurrences of a given entity in the training set to be inversely proportional to the entity ranking. Following this strategy, we expect the RNN to associate the word *Berlin* with the German capital and not with *Berlin, New Hampshire*.

3. Experiments and current progress

We selected the DBpedia Knowledge Base (Lehmann et al., 2015) as the dataset for our experiments, due to its central importance for the Web of Data. We built a dataset of 3,108 entities about movies from DBpedia and annotated 20 and 4 question-query templates with one and two placeholders, resp. Our preliminary results are given in Table 1.
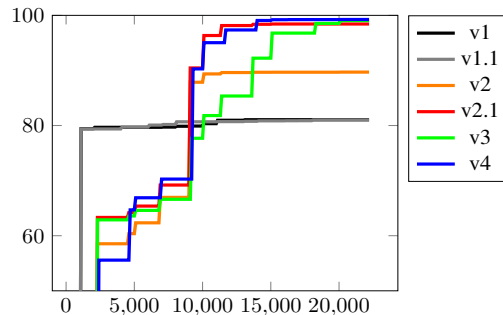

Figure 2. BLEU accuracy against training epochs.

We experimented with 6 different SPARQL encodings, i.e. ways to encode a SPARQL query into a sequence of tokens. At each row of the table, we provide the description of the corresponding changes, each of which persists in the next encodings. The experiments were carried out on a 64-CPU Ubuntu machine with 512 GB RAM.[1] We adopted the implementation of *seq2seq* in *TensorFlow* with internal embeddings of 128 dimensions, 2 hidden layers, and a dropout value of 0.2. All settings were tested on the same set of unseen questions after applying an 80-10-10% split.

The results confirmed that the SPARQL encoding highly influences the learning. Adding more complex templates (i.e., with more than one placeholder) to the generator input yielded a richer training set and more questions were parsed correctly. Merging tokens (see queries and their respective sequences in Figure 1) helped the machine translation, as the SPARQL sequences became shorter. Adding alignments of entities and their labels to the training set turned out to be beneficial for a faster convergence, as Figure 2 shows. The most frequent errors were due to entity name collisions and out-of-vocabulary words; both issues can be tackled with the strategies introduced in this work.

We plan to perform an evaluation on the WEBQUESTIONSSP (Yih et al., 2016) and QALD (Unger et al., 2014) benchmarks to compare with the state-of-the-art approaches for KBQA and QALD, respectively.

---

[1]Code available at https://github.com/AKSW/NSpM.

References

Abujabal, A., Yahya, M., Riedewald, M., and Weikum, G. Automated template generation for question answering over knowledge graphs. In *Proc. of the 26th Int. Conf. on World Wide Web*, pp. 1191–1200, 2017.

Abujabal, A., Saha Roy, R., Yahya, M., and Weikum, G. Never-ending learning for open-domain question answering over knowledge bases. In *Proc. of the 2018 The Web Conference*, pp. 1053–1062, 2018.

Bao, J., Duan, N., Zhou, M., and Zhao, T. Knowledge-based question answering as machine translation. In *Proc. of the 52nd Annual Meeting of the ACL*, 2014.

Bengio, Y., Louradour, J., Collobert, R., and Weston, J. Curriculum learning. In *Proc. of the 26th annual Int. Conf. on Machine Learning*, pp. 41–48. ACM, 2009.

Cer, D., Yang, Y., Kong, S.-y., Hua, N., Limtiaco, N., John, R. S., Constant, N., Guajardo-Cespedes, M., Yuan, S., Tar, C., et al. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*, 2018.

Elsahar, H., Gravier, C., and Laforest, F. Zero-shot question generation from knowledge graphs for unseen predicates and entity types. *arXiv preprint arXiv:1802.06842*, 2018.

Hao, Y., Zhang, Y., Liu, K., He, S., Liu, Z., Wu, H., and Zhao, J. An end-to-end model for question answering over knowledge base with cross-attention combining global knowledge. In *Proc. of the 55th Annual Meeting of the ACL (Vol. 1: Long Papers)*, pp. 221–231, 2017.

Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., Van Kleef, P., Auer, S., et al. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195, 2015.

Liang, C., Berant, J., Le, Q., Forbus, K. D., and Lao, N. Neural symbolic machines: Learning semantic parsers on freebase with weak supervision. *arXiv preprint arXiv:1611.00020*, 2016.

Lopez, V., Unger, C., Cimiano, P., and Motta, E. Evaluating question answering over linked data. *Journal of Web Semantics*, 21:3–13, 2013.

Lukovnikov, D., Fischer, A., Auer, S., and Lehmann, J. Neural network-based question answering over knowledge graphs on word and character level. In *Proc. of the 26th Int. Conf. on World Wide Web*, 2017.

Marx, E., Zaveri, A., Moussallem, D., and Rautenberg, S. Dbtrends: Exploring query logs for ranking rdf data. In *Proc. of the 12th International Conference on Semantic Systems*, pp. 9–16. ACM, 2016.

McCrae, J. P., Abele, A., Buitelaar, P., Cyganiak, R., Jentzsch, A., and Andryushechkin, V. The Linked Open Data Cloud, 2018. URL http://lod-cloud.net/.

Prud'hommeaux, E., Seaborne, A., et al. SPARQL query language for RDF. Technical report, World Wide Web Consortium, 2006. URL http://www.w3.org/TR/rdf-sparql-query/.

Reed, S. and De Freitas, N. Neural programmer-interpreters. *arXiv preprint arXiv:1511.06279*, 2015.

Seaborne, A. Arq-a sparql processor for jena. *Obtained Internet Httpjena Sourceforge NetARQ*, 2010.

Sorokin, D. and Gurevych, I. End-to-end representation learning for question answering with weak supervision. In *Semantic Web Evaluation Challenge*, pp. 70–83. Springer, 2017.

Soru, T., Marx, E., Moussallem, D., Publio, G., Valdestilhas, A., Esteves, D., and Neto, C. B. SPARQL as a foreign language. In *13th Int. Conf. on Semantic Systems (SEMANTiCS 2017) - Posters and Demos*, 2017.

Unger, C., Forascu, C., Lopez, V., Ngomo, A.-C. N., Cabrio, E., Cimiano, P., and Walter, S. Question answering over linked data (qald-4). In *Working Notes for CLEF 2014 Conf.*, 2014.

Wang, Z., Zhang, J., Feng, J., and Chen, Z. Knowledge graph and text jointly embedding. In *Proc. of the 2014 conf. on empirical methods in natural language processing (EMNLP)*, pp. 1591–1601, 2014.

Yih, W.-t., Richardson, M., Meek, C., Chang, M.-W., and Suh, J. The value of semantic parse labeling for knowledge base question answering. In *Proc. of the 54th Annual Meeting of the ACL (Vol. 2: Short Papers)*, pp. 201–206, 2016.

Zafar, H., Napolitano, G., and Lehmann, J. Formal query generation for question answering over knowledge bases. In *15th Extended Semantic Web Conference*, 2018.

Zhong, V., Xiong, C., and Socher, R. Seq2sql: Generating structured queries from natural language using reinforcement learning. *arXiv preprint arXiv:1709.00103*, 2017.