Question Answering

Timothy Baldwin

THE UNIVERSITY OF MELBOURNE

Talk Outline

Overview

- **Definition:** question answering ("QA") is the task of automatically determining the answer (set) for a natural language question
- **Primary approaches:**
  - Natural Language Interface to Database ("NLIB") = automatically construct a query, and answer question relative to fixed KB
  - direct text matching = answer question via string/text passage in a document (collection)
- Strong focus on "factoid" QA, but some work on more open-ended task of multi-faceted QA
- A wonderful case study for tracing the history of NLP

Talk Outline

Knowledge-rich Restricted-domain QA

- The first phase of QA research focused on NL interface to knowledge base from a particular domain (e.g. blocks world, baseball statistics, or lunar rock samples) with idiosyncratic "data semantics"
- QA system = means of translating NL query into formal representation that can be used to query knowledge base directly

Example System: LUNAR

- LUNAR system [Woods, 1978]:
  - knowledge base = geological facts/figures relating to lunar rock samples from Apollo 11
    - *The application envisaged was a system that would be accessible to geologists anywhere in the country by teletype connections and would enable them to access the NASA data base without having to learn either the programming language in which the system was implemented or the formats and conventions of the data base representations.*
    - example question: *In which breccias is the average concentration of titanium greater than 6 percent?*
  - parser = "augmented transition grammar" (ATN), with 3500 lexical entries

Example System: LUNAR

- focus on "meaning representation language" output of the ATN, as means of capturing natural language semantics (with particular focus on "logical quantification")

```
(FOR EVERY XI / (SEQ SAMPLES):
    (CONTAIN X1 OVERALL SILICON) ;
    (PRINTOUT XI)).
```

- coverage = ???
- means of evaluation = ???
- accuracy = ???
- things that are noticeably lacking = parse ranking/disambiguation, anaphora resolution, robustness handling

Knowledge-rich Restricted-domain QA: Limitations

- Biggest issues:
  - specific to a particular domain, and doesn't generalise across domains
  - scalability — inherently limited in scale by the size of the knowledge base
- The NL parsers used for knowledge-rich restricted-domain QA tended to adopt a "depth-first" approach to resource development (= attempt to parse particular input types particularly well, at expense of generality/coverage)
- Note that these are limitations of these early approaches, and that there have been successful applications of general-purpose parsers to knowledge-rich restricted-domain QA [Mollá and Vicedo, 2007, Packard, 2014]

Talk Outline

QA as IR

- The focus of the IR community has always been the resolution of "user information needs" (what information [type] is the user after in issuing a given query?), e.g. Broder [2002]:
  - Informational: want to learn about something (≈40%)
    `Australian submarine contract`
  - Navigational: want to go to a particular page (≈25%)
    `Australian Taxation Office`
  - Transactional: want to do something (web-mediated) (≈35%)
    - Access a service (e.g. `Melbourne weather`)
    - Shop (e.g. `Samsung Galaxy S7`)
  - Mixed-mode information needs
    - Find a good hub site (e.g. `Tokyo hotels`)
    - Exploratory search
- As such, it is natural to treat QA as an IR problem

Example Task: TREC QA

- TREC (= the annual (US) forum for IR evaluations, across a range of different tasks) introduced QA as a "track" in 1999 [Voorhees, 1999]
- Document collection = newspaper data initially; now moved to include more varied web collections, social media data streams, etc.
- Queries = factoid-style questions (with guaranteed answer in collection)

```
Number: 10000
What date in 1989 did East Germany open the Berlin Wall?
LA012890-0072
LA122489-0101
Nov 9
```

since expanded to include definition- and list-based answers, and query sessions [Dang et al., 2007]

Example Task: TREC QA

```
<qa>
    <q id="222.1" type="FACTOID">
        When was 3M founded?
    </q>
</qa>
<qa>
    <q id="222.2" type="FACTOID">
        Where is the company based?
    </q>
</qa>
..
<qa>
    <q id="222.7" type="LIST">
        What brand name products does 3M manufacture?
    </q>
</qa>
<qa>
    <q id="222.8" type="OTHER">
    </q>
</qa>
```

Example Task: TREC QA

- Results = document ranking + 50/250-byte "snippet" in each document
- Evaluation = initially based on MRR, but since moved across to manual precision and recall evaluation based on the returned "nuggets"

Example Task: TREC QA

- Classical IR approach:
  - query the document collection with question q_i, and return document ranking $D_i = \langle d_{i,j} \rangle$
  - analyse q_i to determine the expected answer type $t(q_i)$
  - perform some shallow parsing/NER over (a snippet of) $d_{i,j}$ to find entities of type $t(q_i)$, and rerank D_i accordingly

(Pure IR) Example System: PRICS

- Combine "coordinate matching" (= TF) with features including [Kwok et al., 2006]:
  - synonym matching
  - document rank
  - IDF
  - exact match for "important" words
  - proximity
  - heading match score
  - phrasal match
- Ad hoc scoring of different features to generate final document–extent ranking

QA in Modern Search Engines

- There has long been commercial interest in QA, but early commercial-scale QA engines tended to involve a lot of (semi-automatic) data curation and database lookup rather than actual NLP
- Only recently are we seeing commercial search engines conservatively and strategically providing QA-style results to queries

Talk Outline

QA as IE

- The TREC QA tasks emerged out of the IR community, but were quickly picked up on as relevant by the information extraction (IE) community (at a time when IE was languishing slightly ...)
- IE components of a TREC-style system:
  - question classification
  - named entity recognition

Question Classification

- **Task** = predict the entity type of the answer based on the wording of the question
- Example question classification [Li and Roth, 2002]: hierarchy of 6 coarse-grained categories (`ABBREVIATION`, `ENTITY`, `DESCRIPTION`, `HUMAN`, `LOCATION` and `NUMERIC VALUE`), which decompose into 50 fine-grained classes (e.g. {`city`, `country`, `mountain`, `state`, `other` } $\in$ `LOCATION`)

Question Classification

- **Example approach 1:** feature engineering-based discriminative model [Blunsom et al., 2006], with features including
  - bag-of-words features (unigrams, bigrams, trigrams)
  - positional *n*-gram features
  - quoted *n*-grams
  - POS, chunk, CCG supertag and NE features
  - "target" word features, based on CCG parse tree (word, POS, chunk, NE, ...) + positional *n*-grams
- **Example approach 2:** deep learning model, whereby the features are automatically constructed as part of the training [Kalchbrenner et al., 2014]
  - deep convolutional neural network over word embeddings

QA as Semantic Parsing

- The primary issue with the early work on knowledge-rich restricted-domain QA was domain specificity (of the KB and the parser); people also focused (too?) heavily on formal semantic representation
- In the 00s, large-scale knowledge bases became available such as Freebase and YAGO, semi-automatically seeded from resources such as DBpedia, WordNet and GeoNames, and expanded through community contributions + "machine reading"
- Associated with these KBs were pre-defined general-purpose data schemas and query languages (e.g. SPARQL)
- This led to a revitalisation of NLIB research, in the form of knowledge-rich *open*-domain QA and under the title of "semantic parsing" / "machine reading"

QA as Semantic Parsing

- **Semantic parsing** = automatically translate a natural language text into a formal meaning representation [Wong and Mooney, 2007, Poon and Domingos, 2009]
- **Basic approach:** induce a parser model which produces a formal meaning representation based on supervised learning (over query–MR pairs), or based on alignment between some intermediate representation (e.g. dependency graphs) and the query language; use constraints of query language, and possibly analysis of query results, to prune/refine the parser output

Talk Outline

QA as Deep Learning

- In line with some of the more ambitious goals of deep learning, there has also been recent work which has attempted to perform string–string QA in an end-to-end deep learning architecture, e.g.
  - "quiz bowl" QA, mapping paragraph-length quiz-style questions to factoid answers [Iyyer et al., 2014]
  - "episodic" QA, where questions can be asked of the current state of different entities in a monologue [Weston et al., 2015, Kumar et al., 2015], e.g.:

```
I: Jane went to the hallway.
I: Mary walked to the bathroom.
I: Sandra went to the garden.
I: Daniel went back to the garden.
I: Sandra took the milk there.
Q: Where is the milk?
A: garden
```

QA as Deep Learning

- While deep learning can be highly effective at ranking answer sets or matching questions to answers, it is not currently scalable as a full-on IR solution, so most deep learning research focuses on answering questions relative to text passages which contain the answer, answer set ranking (assuming pre-retried answer set), etc.

Example Deep QA Dataset: SQuAD

- Randomly sample paragraphs from popular Wikipedia pages, and have crowdworkers: (a) create questions which are answerable from that paragraph; and (b) answer questions created by others based on that paragraph [Rajpurkar et al., 2016]
- Example:

> The economy of Victoria is highly diversified: service sectors including financial and property services, health, education, wholesale, retail, hospitality and manufacturing constitute the majority of employment. Victoria's total gross state product (GSP) is ranked second in Australia, although Victoria is ranked fourth in terms of GSP per capita because of its limited mining activity. ...

  - *What kind of economy does Victoria have?*
  - *Where according to gross state product does Victoria rank in Australia?*

A Related Challenge Dataset

- Levesque [2014] proposed a set of "Winograd Schema" as a means of evaluating general AI via QA pairs:
  - Two parties are mentioned in the question (both are males, females, objects, or groups)
  - A pronoun is used to refer to one of them (*he* ,*she*, *it*, etc.)
  - The question is always the same: what is the referent of the pronoun?
  - Behind the scenes, there are two special words for the schema. There is a slot in the schema that can be filled by either word. The correct answer depends on which special word is chosen.

  Example:

  > *Joan made sure to thank Susan for all the help she had given. Who had given the help?*

Talk Outline

Summary

- What is question answering?
- What are the different paradigms of QA?
- What are different approaches to QA evaluation?
- What are different sub-tasks associated with QA?

References

Phil Blunsom, Krystle Kocik, and James R. Curran. Question classification with log-linear models. In *Proceedings of 29th International ACM-SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2006)*, pages 615–616, 2006.

Andrei Broder. A taxonomy of web search. *SIGIR Forum*, 36(2):3–10, 2002.

Hoa Trang Dang, Diane Kelly, and Jimmy J. Lin. Overview of the TREC 2007 Question Answering Track. In *Proceedings of TREC 2007*, 2007.

Mohit Iyyer, Jordan Boyd-Graber, Leonardo Claudino, Richard Socher, and Hal Daumé III. A neural network for factoid question answering over paragraphs. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, pages 633–644, 2014.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*, 2014.

Ankit Kumar, Ozan Irsoy, Jonathan Su, James Bradbury, Robert English, Brian Pierce, Peter Ondruska, Ishaan Gulrajani, and Richard Socher. Ask me anything: Dynamic memory networks for natural language processing. *arXiv preprint arXiv:1506.07285*, 2015.

K.L. Kwok, L. Grunfeld, N. Dinstl, and M Chan. Trec-9 cross language, web and question-answering track experiments using PIRCS. Technical report, DTIC Document, 2006.

Hector J. Levesque. On our best behaviour. *Artificial Intelligence*, 212:27–35, 2014.

References

Xin Li and Dan Roth. Learning question classifiers. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING 2002)*, pages 556–562, Taipei, Taiwan, 2002.

Diego Mollá and José Luis Vicedo. Question answering in restricted domains: An overview. *Computational Linguistics*, 33(1):41–61, 2007.

Woodley Packard. UW-MRS: Leveraging a deep grammar for robotic spatial commands. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 812–816, Dublin, Ireland, 2014. URL `http://www.aclweb.org/anthology/S14-2144`.

Hoifung Poon and Pedro Domingos. Unsupervised semantic parsing. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP 2009)*, pages 1–10, 2009.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*, 2016.

Ellen M. Voorhees. The TREC-8 question answering track report. In *Proceedings of TREC-8*, volume 99, pages 77–82, 1999.

Jason Weston, Antoine Bordes, Sumit Chopra, and Tomas Mikolov. Towards AI-Complete question answering: A set of prerequisite toy tasks. *arXiv preprint arXiv:1502.05698*, 2015.

References

Yuk Wah Wong and Raymond J Mooney. Learning synchronous grammars for semantic parsing with lambda calculus. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL 2007)*, Prague, Czech Republic, 2007.

William A. Woods. Semantics and quantification in natural language question answering. *Advances in Computing*, 17, 1978.