Aggregation-Based Answering for Broad Product Questions

Eilon Sheetrit[1,†], Yuval Nezri[2], Avihai Mejer[2] and David Carmel[2]

[1]Reichman University
[2]Amazon

Abstract

Product Question Answering (PQA) is a popular and important feature in e-commerce services that many customers use as part of their shopping journey. Most previous works on PQA focus on questions that were asked in the context of a specific product. In this work we address a different use case of answering product questions without the context of a specific product. We refer to these questions as Broad Product Questions (BPQ) as these questions often target a broad set of relevant products, such as *"Do jeans shrink after a wash?"*. We propose a new answering approach to address BPQs by aggregating information from a set of relevant products. We highlight the advantages of the aggregation-based answering approach in context of e-commerce, and we present an empirical evaluation of the utility users find in these answers compared to common web retrieval-based answers.

Keywords

question answering, broad product questions, aggregation-based answers

1. Introduction

Product Question Answering (PQA) is an important and helpful feature in e-commerce services that many customers use as part of their shopping journey [1, 2, 3]. The common use case for PQA are questions which are asked in the context of a specific product, for example, via a Q&A search bar on the product's details page. A reference to the specific product can be explicit, e.g. *"how should I clean **this humidifier**?"*, or implied, such as *"Can I turn off the display?"*, asked in the context of a specific humidifier item. Most of the existing answering strategies utilize semi-structured information provided on the product details page, namely product description and specifications (e.g., [4, 5, 6, 7]), community Q&As (e.g., [1, 8, 9]), and customer reviews (e.g., [1, 10, 11, 12, 13, 14]).

In this work we address a different use case of answering product questions that are not associated with a specific product, but rather address a broad set of relevant products, for example, *"how many watts an **air conditioner** takes?"*; we refer to these questions as Broad Product Questions (BPQs). Such questions are issued by customers on the search bar of e-commerce websites without the context of a specific product. These questions are commonly

eCom'23: ACM SIGIR Workshop on eCommerce, July 27, 2023, Taipei, Taiwan

[†]Portions of this work were done while working at Amazon.

eilon.sheetrit@post.runi.ac.il (E. Sheetrit); yuvnez@amazon.com (Y. Nezri); amejer@amazon.com (A. Mejer); dacarmel@amazon.com (D. Carmel)

Table 1
Examples of web-based answers from two popular web search engines (Jan' 4, 2023).

Question	Search Engine 1	Search Engine 2	Comment
how heavy is a pillow	Typically the queen pillows weigh around 3.5 to 4.0 pounds.	A flat standard pillow can weigh up to 8 oz., while a medium sized standard pillow weighs approximately 14 oz.	Good answers
how heavy is a **feather** pillow	No direct answer	No direct answer	Only "10 blue links" were provided
how many watts does a microwave use	Compact Microwave 600-800 watts [...] Standard Microwave (800-1000 watts)	Typically, microwaves use 500-800 watts for working.	Good answers
how many watts does a **toshiba** microwave use	0.9 Cubic feet *900 watt* Stainless Steel microwave [...]	The Toshiba EM131A5C SS microwave oven has a power output of *950 watts*.	Answers relate to one specific product

answered using a Web-Based Question Answering (Web-QA) approach [15], i.e., retrieving an answer based on relevant text snippets from the web.

While Web-QA has made much progress in recent years, it suffers from several limitations in answering product related questions. Due to the highly dynamic nature of product information (new versions, price changes) Web-based answers are not guaranteed to be up-to-date, and, in the context of an e-commerce service, to be grounded in active products, i.e. products that are still available and can be purchased. Another notable limitation is the lack of ability to answer a question at varying specificity levels. Several examples are shown in Table 1, where commercial search engines, who had recently expanded their question answering capabilities, retrieve a good answer to a broad product question, e.g., *"how heavy is a pillow"*, however when adding a qualifier to the question, such as *"how heavy is a **feather** pillow"*, the engines fail to provide a satisfying answer. In other cases, the web-based answer relates to only one specific product, rather than to the family of products that the question asks about (See the "Toshiba microwave" example in Table 1).

In this work we propose a novel answering framework for addressing BPQs, in the context of an e-commerce service, which allows mitigating the aforementioned limitations. The answering framework, illustrated in Figure 1, is based on aggregation of information from a set of relevant products and operates as follows: (1) the target item-name which represents the requested product set is identified in the question, (2) a representative set of relevant products are gathered, (3) the relevant information is extracted for each product in the set, and (4) an answer is generated from the aggregated information. Since the answer is algorithmically generated from the e-commerce catalog, the freshness and quality of the data is grounded, the set broadness of relevant products is under control, and the answer is associated only with concrete products that are currently available for purchase, while omitting ‘ghost’ products which have vanished from the market. Moreover, it is possible to refer to several concrete relevant products directly in the answer. For example, the question *“how heavy are thermal curtains?”* is answered by our method with *“Between 1.2 and 5.1 pounds, based on 42 thermal curtains products. Several alternatives include H.VERSAILTEX (1.9 lb), Deconovo (2.9 lb), and NICETOWN (5 lb).”*. We refer to these type of answers as Aggregation-based Answers. Several more PQA examples are presented in Table 2. The empirical study we present suggests that such answers can complement, and in some cases replace, the web-based answers.

The aggregation-based answering approach can gather product information from a variety of sources such as product-description, specifications, Q&As, and customer reviews. However, as a first step, we focus in this work on product specifications only and limit ourselves to product attribute questions. We leave for future work the utilization of additional sources for BPQ answering, and the handling of additional question types. In the rest of the paper we (i) describe in detail the BPQ answering framework, (ii) present empirical evaluation of the utility that users find in aggregation-based answers. We also report on user satisfaction results from an online experiment of presenting the new form of answers to customers of an e-commerce service.

2. Related Work

The line of work mostly related to ours is on answering product-related questions; e.g., [2, 16, 3, 14, 1, 8, 4, 7]. The retrieved answer can be extracted from the product related Q&As [4, 1, 2, 3, 8, 9], from customer reviews [13, 12, 4, 1, 16, 17, 14, 3], or by aggregating multiple sources of information [18, 10, 11]. For a comprehensive survey on PQA approaches we refer the reader to the work of Deng et al. [19]. All these works focus on questions asked in the context of a specific product.

Answering attribute questions is a sub-task of Question Answering [15, 20, 7], and is also closely related to the task of attribute value extraction [21, 22, 23]. Previous works focused on retrieving a single table entry or aggregating some entry values that correctly answer the question [24, 23, 25, 26, 27]. However, these works assume the table of data exists, while in our work we dynamically collect the relevant data. These methods can handle complex questions that require the consideration of table entry relations and the whole table structure. Integrating inter-relations within attribute values into our method is an interesting direction for future research.

3. Aggregation Based Answering Framework

The following section describes the aggregation-based framework for answering BPQs. We focus on answering attribute questions from product specifications (BPQ-PS). Figure 1 describes the BPQ-PS framework, its components are further discussed below.

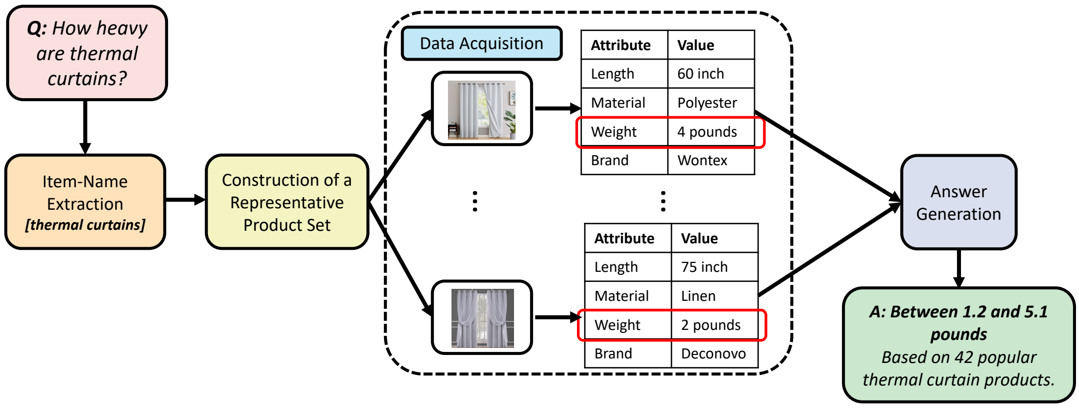

Figure 1: BPQ-PS – Broad Product Question Answering framework, based on the product specifications in the catalog.

3.1. Item-Name Extraction

First, an item-name is extracted from the user's question. An item-name refers to the phrase in the question specifying the product or product-set the user asks about. This item-name is then used to infer the product set for aggregation, therefore, it is crucial that it will be specific as possible, capturing all available information in the query. For example, for the question: *"How heavy are thermal curtains?"*, if only the item "curtains" is identified, while the "thermal" attribute is omitted, the accumulation will be applied to regular curtains, resulting with a wrong answer.

Item-names can be identified by Named Entity Recognition (NER) which is a well known task that has been studied intensively by the NLP community (e.g. [28]), and many publicly available tools exist for this task [29, 30]. In our BPQ-PS implementation, we use an in-house Transformers-based NER tool [31] that was specifically trained for identifying item-names in product related queries and questions.

3.2. Representative Product Set Extraction

Next, we collect a representative set of products that are relevant to the extracted item-name. When choosing the representative product set multiple aspects should be considered, such as freshness, availability, popularity, diversity, and completeness. In our BPQ-PS implementation we focus on popular products, rather than on a complete set, as these better represent the products that a typical user is looking for. We leverage the e-commerce product search service to retrieve a large and diverse collection of products relevant to the extracted item-name. These products are also guaranteed to be up-to-date and available for immediate purchase.

Popular items from the retrieved collection are then selected by leveraging historical purchasing data of several months; only products that were purchased more than a certain fraction of times, following a product search with the item-name as the query, are retained. As a result, the representative product set contains only up-to-date, available, popular, and most relevant products to the item-name.

3.3. Data Acquisition

In the data acquisition stage we extract the product-centric data that is relevant to the user's question, for each product in the product set. We focus on product attribute questions that can be answered by product specifications. Product specifications is a collection of *(attribute: value)* pairs representing the product, typically a few dozens per product. Attribute values may be Boolean (e.g. *(dishwasher-safe: true)*), numeric, typically accompanied with their units, (e.g. *(display size: 5.1 inches)*), or categorical (e.g. *(connectivity: 5G, 4G, Bluetooth)*). Some attributes are repeated for most products across all categories, e.g., *brand* and *weight*, while others are unique to some categories or to a subset of products, such as *calories* and *wireless connectivity*.

The attribute value extraction step requires identifying the relevant attribute to answer the user's question. For the question *"How heavy are thermal curtains?"* the relevant attribute *weight* should be identified. To this end, we follow the ***template*** method proposed in previous work [7]. For each *(attribute: value)* pair we generate a template-based answer in the form "The *attribute* is *value*" (e.g. "The weight is 32 kg") and estimate the probability that this answer satisfies the user's question. We then select the attribute with the highest probability which is above a pre-defined threshold; no answer is returned if no such attribute exists. The probability that an answer satisfies the question is estimated using a pre-trained RoBERTa model [32], fine-tuned on Amazon-PQA corpus [8][1].

In the common PQA use-case, where the goal is providing relevant information for a single product, returning an answer based on one extracted *(attribute:value)* pair might be sufficient. In our case, however, aggregation is required over many pairs, and in general, the attribute information associated with different products is provided by different manufacturers and sellers. Therefore, additional steps of data cleansing is required, including normalization, unification, and filtering. For Boolean attributes, we transform each value to Boolean and count the portion of products with a positive value ('true') and negative value ('false'). For numeric attributes, we apply an open source unit normalization library[2] and inter quartile range outlier filtering, to enable measuring the range of values and other statistics. For categorical attributes we filter out outliers, typos and invalid values using manually generated regular expressions.

3.4. Answer Generation

Recent advancement in language generation methods allow transforming the data elements collected in the previous stage into a natural language answer. However, in order to focus our analysis on the utility of the aggregated answers, rather than on the language quality, we choose to utilize a handful of manually-curated textual templates for answer generation. The answer template provides details on the number of relevant products it is based on, and the value aggregation over this set. For categorical attributes (e.g. colors or materials) we provide up to five most frequent attribute values. For numeric attributes (e.g. weight or wattage) we provide the range of values. For Boolean attributes (e.g. microwave safe), we provide the ratio between positive and negative attribute values, or a yes/no answer in case of agreement. Examples for each type of answer template are presented in Table 2.

[1] We follow the fine-tuning process described in Shen et al. [7].
[2] https://github.com/hgrecco/pint

Table 2
Examples of Aggregation-based Answers.

type	question	answer
Categorical	What is a pillow made of?	Based on 32 pillow products, there are 10 fill materials. Some of them are memory foam, polyester, gel memory foam, cotton, and poly gel fiber.
Numerical	What is the weight of a mlb bat?	Between 1.75 and 2.15 pound, based on product details from 5 mlb bat products.
Boolean	Are Calphalon pans oven safe?	Yes, based on product details from 6 Calphalon pan products.

4. Experiments

The main objective of our experiments is to evaluate the utility users find in aggregation-based answers and the merits of using these answers to augment web-based answers. For that, we asked human annotators to evaluate the quality of Web-QA and BPQ-PS answers, and a combination of the two. We also conducted an online experiment on a commercial e-commerce service and report on users' satisfaction.

Question Data. Our experiments were conducted on a subset of the Natural Questions (NQ) dataset [33]. We applied an in house NER classifier [31] to detect the subset of product questions; i.e., questions that refer to a product. In total we were able to identify $7,583$ product questions. We refer to this subset as PNQ.

BPQ-PS. While running BPQ-PS to answer the questions in PNQ, we applied several filters in order to identify questions to which the answering method is suitable. First, we retained only questions for which a single item-name was identified. Questions with zero item-names are obviously not suitable, and questions with multiple item-names pose challenges in constructing a representative product set that we defer to future work. As our focus is on broad questions, we retained only questions that had at least 5 valid attribute values, from different products, following the product set extraction and attribute-value extraction steps. This process resulted in 429 questions that can be answered by BPQ-PS, reflecting coverage of 5.7% of the PNQ dataset[3]. Leveraging additional data sources beyond product specifications, e.g. customer reviews or product descriptions, can increase the coverage of the aggregation-based approach and is left for future work.

Web-QA. We run a Web-QA system to retrieve web-based answers for all the 429 questions that were answered by BPQ-PS. The Web-QA system we used is of a commercial e-commerce service; it indexes web pages from a broad and diverse set of websites and retrieves the most relevant answer using a combination of dense passage retrieval [34] and answer selection [35] algorithm.

[3]The subset of questions we retained does not represent a uniform sample of the question traffic submitted on e-commerce platforms.

Table 3
Annotation results of Web-QA and enriched Web-QA+BPQ-PS answers over the PNQ dataset. $*$ denotes statistically significant improvement (two tails paired t-test, $p < 0.05$).

Method	Avg. (100%)	Complementary (19%)	Inclusion (21%)	Contradiction (47%)
Web-QA	4.70	4.59	4.77	4.68
Web-QA+BPQ-PS	4.73	4.76	4.79	4.66
Δ	(+0.03)	(+0.17)*	(+0.02)	(-0.02)

4.1. Results

Manual Quality Evaluation. The answers for the 429 questions, both from Web-QA and BPQ-PS, were evaluated each by 3 human annotators[4]. The annotators were asked to judge each answer independently as either: relevant, somewhat relevant, or not-relevant, the labels were determined by a majority vote. Among the BPQ-PS answers, 46% were labeled as relevant, 22% as somewhat relevant, and the remaining 32% as not-relevant. The main reason for labeling an answer as not-relevant, accounting for 72% of these cases, was that the question is not an attribute question. Additional reasons for non-relevance are wrong attribute selection or incorrect item-name extraction from the question, these account for 12% and 7% of the not-relevant cases, respectively. The Web-QA answers were labeled as relevant and somewhat relevant for 84% and 8% of the cases respectively, attesting to the high quality and maturity of the Web-QA system we used.

In order to assess the utility users can find in augmenting the high quality Web-QA answers with BPQ-PS answers we conducted a second study. We evaluated 169 questions to which both the BPQ-PS and Web-QA answers were annotated as relevant. First, each pair of answers was reviewed by 3 annotators in order to characterize the relationship between them. Among the 169 pairs the distribution over relationship type was: contradiction (47%), inclusion (21%), complementary (19%), equivalence (5%), and other (8%). Next, annotators where asked to evaluate the original Web-QA answer and an enriched Web-QA+BPQ-PS answer[5], for example, for the questions *"What is the American flag made out of?"*, the enriched answer is *"According to snippets.com: The American flag is made out of polyester materials. In addition: based on 87 popular American flag products, there are 6 materials: nylon, polyester, polypropylene, spun polyester, polyurethane, and cotton."*. The annotators were asked to score each answer from 1 to 5 to indicate how relevant, complete and helpful an answer is. The annotation results are shown in Table 3. We see that on average, the Web-QA answers received very high score of 4.7, and yet, enriching them with BPQ-PS answers further improves the scores to 4.73. In particular we find a statistically significant increase of $+0.17$ when the answers complement each other, yet when the answers contradict or contain each other the scores are on par.

Online Experiment. In addition to the manual crowd-based evaluation, we evaluate the satisfaction of e-commerce users from the aggregation-based answers in an online experiment. On a commercial e-commerce search bar users typically submit short queries seeking for

[4]We used MTurk crowdsourcing platform: https://www.mturk.com/
[5]Different annotators judged the two flavors of the answers, it was not a side-by-side comparison

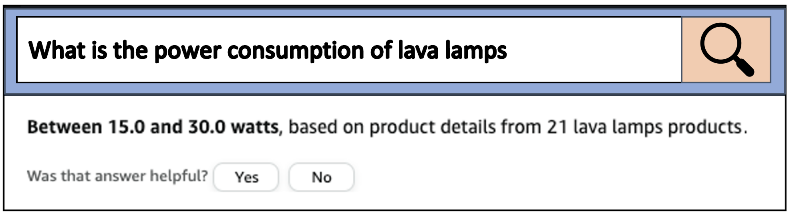

Figure 2: Aggregation-based answer presented on an e-commerce search page.

products. Additionally, as shown in Fig 2, users can submit full natural language questions and receive a direct answer in addition to the standard search results. In order to focus on user satisfaction from the new form of answers, rather than on answer quality, we validated offline the relevance of the answers before serving them to users. Specifically, in an offline process, we collected a sample of popular questions on the service, and answered them with BPQ-PS. We then manually validated the answers and retained only the relevant ones. This PQ&A bank is then used to serve online newly submitted questions that are highly similar to one of the questions in our bank. Figure 2 presents the direct answer to the question, for which users were asked to provide feedback. We measure the Positive Response Rate (PRR), i.e., the percentage of times users selected "*Yes*". The PRR for BPQ-PS yield relative improvement of $\sim$34% compared to the PRR for the Web-QA. While part of the high PRR may be attributed to the manual validation of the BPQ-PS bank, we nevertheless believe it shows that users find the new form of answers engaging and helpful.

5. Conclusions and Future Work

In this work we addressed the task of answering BPQs, i.e., questions that refer to a broad set of products which are asked outside the context of a specific product. We described an aggregation-based answering approach, and implemented BPQ-PS which utilizes product specification data to address attribute questions. Our empirical evaluation demonstrates the merits and utility of the new form of answers. In future work we plan to extend the aggregation-based answering approach by leveraging additional information sources such as community Q&As and customer reviews.

Another interesting direction is leveraging recent large language models to produce high-quality fluent answers. With retrieval augmented generation (RAG) techniques [36], the answer generator is exposed to pieces of evidence based on information retrieved from external resources, leading to more grounded, accurate, and up-to-date answer. For example, the answer of ChatGPT [37] for the question *"how many watts does a car cigarette lighter produce?"* is *"A car cigarette lighter typically produces up to 120 watts of power (12 volts x 10 amps)."* However, when enriching the prompt with the category-based information *"In an e-commerce website we found the following Wattage values for car cigarette lighter: [10, 15, 24, 30, 38, 80, 100, 120, 150,180, 200] watts"*; ChatGPT enriched its answer to: *"The wattage values for car cigarette lighters listed on the provided e-commerce website range from 10 watts to 200 watts. However, the typical power output of a car cigarette lighter is up to 120 watts."*. Retrieval augmented generation opens an interesting direction for improving broad product question answering which we leave for future research.

References

[1] D. Carmel, L. Lewin-Eytan, Y. Maarek, Product question answering using customer generated content-research challenges, in: Proc. of SIGIR, 2018, pp. 1349–1350.

[2] S. Gao, Z. Ren, Y. Zhao, D. Zhao, D. Yin, R. Yan, Product-aware answer generation in e-commerce question-answering, in: Proc. of WSDM, 2019, pp. 429–437.

[3] A. Kulkarni, K. Mehta, S. Garg, V. Bansal, N. Rasiwasia, S. Sengamedu, Productqna: Answering user questions on e-commerce product pages, in: Proc. of WWW, 2019, pp. 354–360.

[4] L. Cui, S. Huang, F. Wei, C. Tan, C. Duan, M. Zhou, Superagent: A customer service chatbot for e-commerce websites, in: Proc. of ACL, 2017, pp. 97–102.

[5] T. Lai, T. Bui, S. Li, N. Lipka, A simple end-to-end question answering model for product information, in: Proc. of ECONLP, 2018, pp. 38–43.

[6] T. M. Lai, T. Bui, N. Lipka, S. Li, Supervised transfer learning for product information question answering, in: Proc. of ICMLA, IEEE, 2018, pp. 1109–1114.

[7] X. Shen, G. Barlacchi, M. Del Tredici, W. Cheng, A. de Gispert, semipqa: A study on product question answering over semi-structured data, in: Proc. of ECNLP, 2022, pp. 111–120.

[8] O. Rozen, D. Carmel, A. Mejer, V. Mirkis, Y. Ziser, Answering product-questions by utilizing questions from other contextually similar products, in: Proc. of NAACL, 2021, pp. 242–253.

[9] H. Mittal, A. Chakrabarti, B. Bayar, A. A. Sharma, N. Rasiwasia, Distantly supervised transformers for e-commerce product qa, in: Proc. of NAACL, 2021, pp. 4008–4017.

[10] Y. Feng, Z. Ren, W. Zhao, M. Sun, P. Li, Multi-type textual reasoning for product-aware answer generation, in: Proc. of SIGIR, 2021, pp. 1135–1145.

[11] S. Gao, X. Chen, Z. Ren, D. Zhao, R. Yan, Meaningful answer generation of e-commerce question-answering, TOIS 39 (2021) 1–26.

[12] J. McAuley, A. Yang, Addressing complex and subjective product-related queries with customer reviews, in: Proc. of WWW, 2016, pp. 625–635.

[13] M. Wan, J. McAuley, Modeling ambiguity, subjectivity, and diverging viewpoints in opinion question answering systems, in: Proc. of ICDM), IEEE, 2016, pp. 489–498.

[14] S. Zhang, J. H. Lau, X. Zhang, J. Chan, C. Paris, Discovering relevant reviews for answering product-related queries, in: Proc. of ICDM, 2019, pp. 1468–1473.

[15] D. Azari, E. Horvitz, S. T. Dumais, E. Brill, Web-based question answering: A decision-making perspective, CoRR abs/1212.2453 (2012). URL: http://arxiv.org/abs/1212.2453. arXiv:1212.2453.

[16] S. Chen, C. Li, F. Ji, W. Zhou, H. Chen, Driven answer generation for product-related questions in e-commerce, in: Proc. of WSDM, 2019, pp. 411–419.

[17] J. Zhao, Z. Guan, H. Sun, Riker: Mining rich keyword representations for interpretable product question answering, in: Proc. of, 2019, pp. 1389–1398.

[18] W. Zhang, Q. Yu, W. Lam, Answering product-related questions with heterogeneous information, in: Proc. of AACL, 2020, pp. 696–705.

[19] Y. Deng, W. Zhang, Q. Yu, W. Lam, Product question answering in e-commerce: A survey, 2023. URL: https://arxiv.org/abs/2302.08092. doi:10.48550/ARXIV.2302.08092.

[20] M. A. C. Soares, F. S. Parreiras, A literature review on question answering techniques, paradigms and systems, Journal of King Saud University-Computer and Information Sciences 32 (2020) 635–646.
[21] D. Davidov, A. Rappoport, Extraction and approximation of numerical attributes from the web, in: Proc. of ACL, 2010, pp. 1308–1317.
[22] Y. Liu, L. Wang, R. Chen, Y. Song, Y. Cai, A put-based approach to automatically extracting quantities and generating final answers for numerical attributes, Entropy 18 (2016) 235.
[23] H. Sun, H. Ma, X. He, W.-t. Yih, Y. Su, X. Yan, Table cell search for question answering, in: Proc of. WWW, 2016, pp. 771–782.
[24] P. Pasupat, P. Liang, Compositional semantic parsing on semi-structured tables, in: Proc. of ACL, 2015, pp. 1470–1480.
[25] J. Herzig, P. K. Nowak, T. Mueller, F. Piccinno, J. Eisenschlos, Tapas: Weakly supervised table parsing via pre-training, in: Proc. of ACL, 2020, pp. 4320–4333.
[26] P. Yin, G. Neubig, W.-t. Yih, S. Riedel, Tabert: Pretraining for joint understanding of textual and tabular data, in: Proc. of ACL, 2020, pp. 8413–8426.
[27] K. Chakrabarti, Z. Chen, S. Shakeri, G. Cao, Open domain question answering using web tables, arXiv preprint arXiv:2001.03272 (2020).
[28] A. Mikheev, M. Moens, C. Grover, Named entity recognition without gazetteers, in: Ninth Conference of the European Chapter of the Association for Computational Linguistics, 1999, pp. 1–8.
[29] M. Honnibal, I. Montani, spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing, 2017.
[30] S. Bird, E. Klein, E. Loper, Natural language processing with Python: analyzing text with the natural language toolkit, " O'Reilly Media, Inc.", 2009.
[31] B. Fetahu, A. Fang, O. Rokhlenko, S. Malmasi, Gazetteer enhanced named entity recognition for code-mixed web queries, in: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2021, pp. 1677–1681.
[32] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, arXiv preprint arXiv:1907.11692 (2019).
[33] T. Kwiatkowski, J. Palomaki, O. Redfield, M. Collins, A. Parikh, C. Alberti, D. Epstein, I. Polosukhin, J. Devlin, K. Lee, et al., Natural questions: a benchmark for question answering research, TACL 7 (2019) 453–466.
[34] V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, W.-t. Yih, Dense passage retrieval for open-domain question answering, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020, pp. 6769–6781.
[35] S. Garg, T. Vu, A. Moschitti, Tanda: Transfer and adapt pre-trained transformer models for answer sentence selection, in: Proceedings of the AAAI conference on artificial intelligence, volume 34, 2020, pp. 7780–7788.
[36] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, et al., Retrieval-augmented generation for knowledge-intensive nlp tasks, Advances in Neural Information Processing Systems 33 (2020) 9459–9474.
[37] OpenAI, Chatgpt, 2022. URL: https://openai.com/blog/chatgpt.