Osservazioni

- I retangoli che vedete ai bordi servono per vedere quanto sono grandi i margini scrivibili della pagina, si possono togliere commentando la riga `\usepackage{showframe}`
- L'abstract adesso conta 267 parole. Di solito gli abstract non sono piu' lunghi di 300
- Il nome del nostro strumento di QA sta in una macro, per usarlo nel report scrivere `\nomefico`
- Il link al repo github sta in una macro, per usarlo scrivere `\github`
- Il link alla pagina web per QA (ancora da definire) sta in una macro, chiamata `\app`
- le immagini sono solo una bozza, vorrei che fossero approvate al 100% prima di realizzarle in latex
- h

nomeFicoDaScegliere
A BERT-based, open domain question answering system

Gabriele Barreca, Mario Bonsembiante and Gemma Martini
University of Pisa

Abstract

Question answering (QA) systems can be seen as information retrieval systems which aim is to respond to queries, stated in natural language, by returning short answers or long sentences. The "so-called" *open domain QA task* adds the challenge of understanding if the answer to the selected question may or may not be found in a given paragraph, which content has been buried within large text corpora, such as Wikipedia.

Building such systems for practical applications has historically been quite challenging and involved. The spectrum of possible answers given a question and a paragraph, moves from the "simple" *yes/no answers* to the longer and more articulated *long answers*, to then get to a trade-off between expressive power and succinctness, the "so-called" *short answers*, which aim to enclose the answer in a single and possibly short sentence.

In this paper, we present a BERT-based implementation that solves an open domain QA task, providing all the three categories of answers listed above, with particular attention on the most widely studied kind, i.e. short answers. We achieve pretty good results, although not as good as the state-of-the-art, that was not the purpose of this work.

As expected and already stated in previous work, we conclude that predicting long answers per se is pretty unreliable, while much better results are achieved if the short answer is predicted and then enlarged with the whole paragraph it lies in, from the original text.

Contents

1 Introduction

Aggiungere introduzione

aggiungere una frase in cui si dice che cosa si trova in quali sezioni

2 The architecture

Before digging into the details of the machine learning core of our BERT-based QA system, let us define the outline of the responsive QA tool we developed.

In Figure 1 there is a pictorial representation of how the user interacts with the system, how it processes the information (server side) and how it prompts the results.

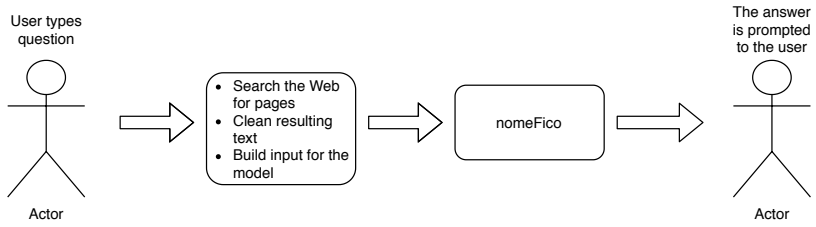

Figure 1: The full functioning of **nomeFicoDaScegliere**, combined with an effective user interface.

Aggiungere descrizione con immagini del workflow dell'applicazione, con un esempio che funziona, magari creando una sottosezione.

2.1 nomeFicoDaScegliere

We are now ready to discuss the implementation of **nomeFicoDaScegliere**.

We decided to tackle the open domain QA task problem by creating a stack of two neural networks, forming a two-layer architecture, see Figure 2.

The first layer is built using BERT's [1] checkpoints, while the second layer is a neural network that uses BERT's embeddings and the Natural

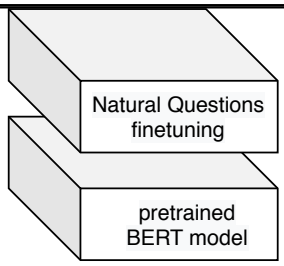

Figure 2: .

Questions (NQ) [2] dataset [1] with the aim of obtaining the answer to the question.

2.1.1 BERT

Let us dig into details a bit more and explain how the BERT layer works.

Fare digressione sull'utilizzo di BERT o BERT-large o ALBERT

2.1.2 Fine-tuning

Depending on the flag start position and end position of the answer in the given paragraph.

At this point, we assume that the system receives as input two textual items, the first one is the question and the second one is the paragraph that is supposed to contain the answer

short?long?

In this section the tools used in the project are described.

3 Experimental results

3.1 Hardware

[1] Some qualities of NQ are the following: (1) the questions were formulated by people out of genuine curiosity or out of need for an answer to complete another task, (2) the questions were formulated by people before they had seen the document that might contain the answer, (3) the documents in which the answer is to be found are much longer than the documents used in some of the existing question answering challenges.

3.2 Hyper-parameters values

4 Future work

Use a binary classification layer to check if the answer is "plausible" (i.e. the meanings in the question are covered in the answer), as done by [3] and [4].

5 Conclusions

5.1 All references

- kbqa [5]
- coqa [6]
- collobert [7]
- vaswani [8]
- weston1 [9]
- weston2 [10]
- alberti 2019 [11]
- kwiatowski 2019 [2]
- chen [12]
- liu purple [13]
- liu yellow [14]
- RoBERTa [15]

References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018.

[2] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Matthew Kelcey, Jacob Devlin, Kenton Lee, Kristina N. Toutanova, Llion Jones, Ming-Wei Chang, Andrew Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: a benchmark for question answering research. *Transactions of the Association of Computational Linguistics*, 2019.

[3] Minghao Hu, Furu Wei, Yu xing Peng, Zhen Xian Huang, Nan Yang, and Ming Zhou. Read + verify: Machine reading comprehension with unanswerable questions. In *AAAI*, 2019.

[4] Seohyun Back, Sai Chetan Chinthakindi, Akhil Kedia, Haejun Lee, and Jaegul Choo. Neurquri: Neural question requirement inspector for answerability prediction in machine reading comprehension. In *International Conference on Learning Representations*, 2020.

[5] Wanyun Cui, Yanghua Xiao, Haixun Wang, Yangqiu Song, Seungwon Hwang, and Wei Wang. Kbqa: Learning question answering over qa corpora and knowledge bases. *Proceedings of the VLDB Endowment*, 10:565–576, 01 2017.

[6] Siva Reddy, Danqi Chen, and Christoper Manning. Coqa: A conversational question answering challenge. 08 2018.

[7] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, 12:2493–2537, November 2011.

[8] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc., 2017.

[9] Mikael Henaff, Jason Weston, Arthur Szlam, Antoine Bordes, and Yann Lecun. Tracking the world state with recurrent entity networks. 12 2016.

[10] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading wikipedia to answer open-domain questions. 03 2017.

[11] Chris Alberti, Kenton Lee, and Michael Collins. A bert baseline for the natural questions. 01 2019.

[12] Yu Chen, Lingfei Wu, and Mohammed J. Zaki. Bidirectional attentive memory networks for question answering over knowledge bases. pages 2913–2923, 01 2019.

[13] Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. Multi-task deep neural networks for natural language understanding. 01 2019.

[14] Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. Improving multi-task deep neural networks via knowledge distillation for natural language understanding. 04 2019.

[15] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. 07 2019.