useful links for a question-answering approach :

1. <https://github.com/sanjogh777/NLP-Project> -> slide show with lots of explanations
2. <https://github.com/th3darkprince/BiDAF_SQUAD>
3. <https://github.com/pranavajitnair/BiDAF>
4. <https://github.com/shrutigrover/Question-Answer-System>
5. <https://github.com/AjinkyaZ/QA682> -> there is a report too
6. <https://github.com/Varun-Mulchandani/roBERTa_based_SQuAD_QA>
7. <https://github.com/GauthierDmn/question_answering>
8. <https://github.com/Avee-81/Question_Answering_on_SQuAD>
9. <https://github.com/kushalj001/pytorch-question-answering> -> important, lots of explanations (implementation of DRQA,BIDAF and QANET)
10. <https://github.com/MurtyShikhar/Question-Answering>

Architecture :

1. Bidirectional Attention Flow Implementation (bidaf)
2. dynamic co attention network
3. DRQA
4. fine tuned transformer (roberta,albert,bert,....) (https://huggingface.co/)
5. QANet (self attention and convolutions)
6. Question answering with multi perspective context matching (<https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1174/reports/2760414.pdf>)

(https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1174/reports/2761065.pdf)

1. seq2seq with attention

The simplest model I can think of is a simple classification task:

* the target is the index of the sentence in the context which contains the answer
* we embed question and context with glove
* we train the classifier (with cosine similarity)
* > similar to what we have done in the assignment 4 but we make not use of the “answer start” data