## Appendix

### Preprocessing and Evaluation

The following details are shared between all the experiments on the different datasets.

**Preprocessing and Word Embeddings**. All the text is preprocessed using SpaCy [1], an off-the-shelf pipeline for tagging, parsing and NE recognition. Questions are tokenized and lowercased. Answer passages are also annotated with NEs. Sentences are padded/truncated at 60. The zero id reserved for padding will be mapped to the zero vector during the embedding operation.

We initialize our word embeddings with publicly available [2] vectors from Severyn and Moschitti (2015). The word embedding matrix is kept fixed during training given the relatively small datasets, which do not allow us to benefit from tuning the matrix weights directly on the data. On the other hand, this reduces the number of parameters and updates, resulting in a faster training.

**Training, model selection and metrics**. The networks are trained using Adam (Kingma and Ba, 2014), setting $\beta_1$ to 0.9 and $\epsilon$ to $1e-5$. During the question classifier and question focus identifier training, we monitor their accuracy on some percentage of heldout examples. The QA model is trained for 30 epochs. In this case, model selection is done by monitoring the Mean Average Precision (MAP) on the development set. The model with the highest MAP is used to compute predictions on the test data. In addition to MAP, we also report the Mean Reciprocal Rank (MRR).

### Question Classifier Hyperparameters

The network applies convolution filters of width 1, 2 and 3 and size 300 to the question, followed by max-pooling. The three pooled representations are concatenated and fed to a hidden layer with 100 dimensions, followed by a ReLU activation function and the final softmax layer. A dropout rate of 0.3 is applied to hidden layer. The optimizer learning rate is set to $5e-4$, the batch size to 32, and we use L2 regularization. We reserve $5\%$ of the data for monitoring the accuracy and picking the best model.

### Question Focus Identifier Hyperparameters

The network applies a stack of 4 convolution filter, with width 10 and size 100. Each filter output, corresponding to a sentence token, is passed through the same hidden ReLU layer of size 100, and a final linear layer outputs the focus score. The Adam learning rate is set to $1e-4$, and the batch size is set to 16. Also here, we use L2 regularization.

### Answer Sentence Selection Hyperparameters

The CNN network for ranking question/answer pairs applies two separate convolution filters of width 5 and size 100 to the question and the answer, followed by a max-pooling operation. The two pooled representations are concatenated and passed through a hidden layer with 200 dimensions, followed by a ReLU activation function and the final softmax layer. A dropout rate of 0.5 is applied to the hidden layer, and L2 regularization is applied on all the parameters. The dimensionality of the word and semantic overlap vectors is set to 5. We train the network using Adam, with a learning rate of $5e-5$, and a batch size of 50. We report mean and standard deviation of 10 random restarts of our models. Note that these settings are shared between all the answer selection experiments.

---

[1] `https://spacy.io/`

[2] `https://github.com/aseveryn/deep-qa`