SemEval-2018 Task 11: Machine Comprehension Using Commonsense Knowledge

Simon Ostermann **Michael Roth** **Ashutosh Modi** **Stefan Thater** **Manfred Pinkal**
Saarland University
Saarbrücken, Germany
{simono|mroth|ashutosh|stth|pinkal}@coli.uni-saarland.de

Abstract

This report summarizes the results of the SemEval 2018 task on machine comprehension using commonsense knowledge. For this machine comprehension task, we created a new corpus, *MCScript*. It contains a high number of questions that require commonsense knowledge for finding the correct answer. 11 teams from 4 different countries participated in this shared task, most of them used neural approaches. The best performing system achieves an accuracy of 83.95%, outperforming the baselines by a large margin, but still far from the human upper bound, which was found to be at 98%.

1 Introduction

Developing algorithms for understanding natural language is not trivial. Natural language comes with its own complexity and inherent ambiguities. Ambiguities can occur, for example, at the level of word meaning, syntactic structure, or semantic interpretation. Traditionally, Natural Language Understanding (NLU) systems have resolved ambiguities using information from the textual context (e.g. neighboring words and sentences), for example via distributional methods (Lenci, 2008). However, many times context may be absent or may lack sufficient information to resolve the ambiguity. In such cases, it would be beneficial to include commonsense knowledge about the world in an NLU system. For example, consider example (1).

(1) The waitress brought Rachel's order. She ate the food with great pleasure.

Looking at the example in isolation, the person eating the food could be either Rachel or the waitress. Using commonsense knowledge, or, more specifically, script knowledge about the RESTAURANT scenario, helps to resolve the referent of the pronoun: Rachel ordered the food. The person who orders the food is the customer. So Rachel should eat the food, *she* thus refers to Rachel.

This shared task assesses how the inclusion of commonsense knowledge benefits natural language understanding systems. In particular, we focus on commonsense knowledge about everyday activities, referred to as *scripts*. Scripts are sequences of events describing stereotypical human activities (also called *scenarios*), for example baking a cake, taking a bus, etc. (Schank and Abelson, 1975). The concept of scripts has its underpinnings in cognitive psychology and has been shown to be an important component of the human cognitive system (Bower et al., 1979; Schank, 1982; Modi et al., 2017). From an application perspective, scripts have been shown to be useful for a variety of tasks, including story understanding (Schank, 1990), information extraction (Rau et al., 1989), and drawing inferences from texts (Miikkulainen, 1993).

Factual knowledge is mentioned explicitly in texts from sources such as Wikipedia and news papers. On the contrary, script knowledge is often implicit in the texts as it is assumed to be known to the comprehender. Because of this implicitness, learning script knowledge from texts is very challenging. There are few exceptions of corpora containing narrative texts that explicitly instantiate script knowledge. An example is the *InScript* (Modi et al., 2016), which contains short and simple narratives, that very explicitly mention script events and participants. The *Dinners from Hell* corpus (Rudinger et al., 2015a) is a similar dataset centered around the EATING IN A RESTAURANT scenario.

In the past, script modeling systems have been evaluated using intrinsic tasks such as event ordering (Modi and Titov, 2014), paraphrasing (Regneri et al., 2010; Wanzare et al., 2017), event prediction (namely, the narrative cloze task) (Chambers and Jurafsky, 2008, 2009; Rudinger et al., 2015b; Modi, 2016) or story completion (e.g. the story cloze task

Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018), pages 747–757
New Orleans, Louisiana, June 5–6, 2018.

T It was a long day at work and I decided to stop at the gym before going home. I ran on the treadmill and lifted some weights. I decided I would also swim a few laps in the pool. Once I was done working out, I went in the locker room and stripped down and wrapped myself in a towel. I went into the sauna and turned on the heat. I let it get nice and steamy. I sat down and relaxed. I let my mind think about nothing but peaceful, happy thoughts. I stayed in there for only about ten minutes because it was so hot and steamy. When I got out, I turned the sauna off to save energy and took a cool shower. I got out of the shower and dried off. After that, I put on my extra set of clean clothes I brought with me, and got in my car and drove home.

Q1 Where did they sit inside the sauna?
a. on the floor b. on a bench

Q2 How long did they stay in the sauna?
a. about ten minutes b. over thirty minutes

Figure 1: An example for a text from MCScript with 2 reading comprehension questions.

(Mostafazadeh et al., 2016)). These tasks test a system's ability to learn script knowledge from a text but they do not provide a mechanism to evaluate how useful script knowledge is in natural language understanding tasks.

Our shared task bridges this gap by directly relating commonsense knowledge and language comprehension. The task has a machine comprehension setting: A machine is given a text document and asked questions based on the text. In addition to what is mentioned in the text, answering the questions requires knowledge beyond the facts mentioned in the text. In particular, a substantial subset of questions requires inference over commonsense knowledge via scripts. For example, consider the short narrative in (1). For the first question, the correct choice for an answer requires commonsense knowledge about the activity of going to the sauna, which goes beyond what is mentioned in the text: Usually, people sit on benches inside a sauna, an information that is not given in the text. The dataset also comprises questions that can just be answered from the text, as the second question: The information about the duration of the stay is given literally in the text.

The paper is organized as follows: In Section 2, we give an overview of other machine comprehension datasets. In Section 3, we describe the dataset used for our shared task. Section 4.2 gives details about the setup of our task. In Section 5, information about participating systems is given. Results are presented and discussed in Sections 6 and 8, respectively.

2 Related Work

Recently, a number of datasets have been proposed for machine comprehension. One example is *MCTest* (Richardson et al., 2013), a small curated dataset of 660 stories, with 4 multiple choice questions per story. The stories are crowdsourced and not limited to a domain. Answering questions in *MCTest* requires drawing inferences from multiple sentences from the text passage. In our dataset, in contrast, answering requires drawing inferences using knowledge not explicit in the text. Another recently published multiple choice dataset is RACE (Lai et al., 2017), which contains 100,000 questions on reading examination data.

Rajpurkar et al. (2016) have proposed the *Stanford Question Answering Dataset (SQuAD)*, a data set of 100,000 questions on Wikipedia articles collected via crowdsourcing. In that dataset, the answer to a question corresponds to a segment/span from the reading passage. Since Wikipedia articles mostly contain factual knowledge, *SQuAD* does not assess how in practice, language comprehension relies on implicit and underrepresented knowledge about everyday activities i.e. script knowledge.

Weston et al. (2015) have created the *BAbI* dataset. *BAbI* is a synthetic reading comprehension data set testing different types of reasoning to solve different tasks. In contrast to our dataset, the artificial texts in *BAbI* are not reflective of a typically occurring narrative text.

Two recently published datasets that also have a larger focus on commonsense reasoning are *NewsQA* and *TriviaQA*. NewsQA (Trischler et al., 2017) contains newswire texts from CNN with crowdsourced questions and answers. During the question collection, workers were only presented with the title of the text, and a short summary. This

method ensures that literal repetitions of the text are avoided and the generation of non-trivial questions requiring background knowledge is supported. The NewsQA text collection differs from MCScript in domain and genre (newswire texts vs. narrative stories about everyday events). Knowledge required to answer the questions is mostly factual knowledge and script knowledge is only marginally relevant.

TriviaQA (Joshi et al., 2017) contains automatically collected question-answer pairs from 14 trivia and quiz-league websites, together with web-crawled evidence documents from *Wikipedia* and *Bing*. While a majority of questions require world knowledge for finding the correct answer, it is mostly factual knowledge.

3 Data

In 3.1, we now briefly describe the machine comprehension dataset used for the shared task, *MCScript*. Parts of the following Section are taken from Ostermann et al. (2018). For a more detailed description of the resource collection and a more thorough discussion of the dataset, we refer to the original paper. Section 3.2 gives details about script data collections that were made available to the participants.

3.1 Machine Comprehension Data - MCScript

For our shared task, we use the *MCScript* data set (Ostermann et al., 2018). It is a collection of narrative texts, questions of various types referring to these texts, and pairs of answer candidates for each question. It comprises 2,119 such texts and a total of 13,939 questions. The texts in the data set talk about everyday activities and cover 110 script scenarios of differing complexity. For the text collection, we followed Modi et al. (2016): All texts are simple and explicit in the description of script events and script participants.

The data set was crowdsourced via Amazon Mechanical turk[1]. In the crowdsourcing experiments, participants were asked to write questions independent of a concrete narrative, but only based on short descriptions of a scenario. By doing so, the collected questions were related to the scenario only and could be answered from different texts, independent of story details.

The scenario-based questions were paired randomly with texts from the same scenario. The subsequent answer collection was divided up into two steps: First, crowdsourcing workers had to annotate whether a question could be answered based on the given text. If it could be answered, they had to explicitly mark whether it could be answered from the text directly or based on commonsense knowledge. Second, they had to write a plausible correct and incorrect answer, if the question was answerable. Afterwards, all texts, questions and answers were manually validated by trained annotators, and corrected, if necessary.

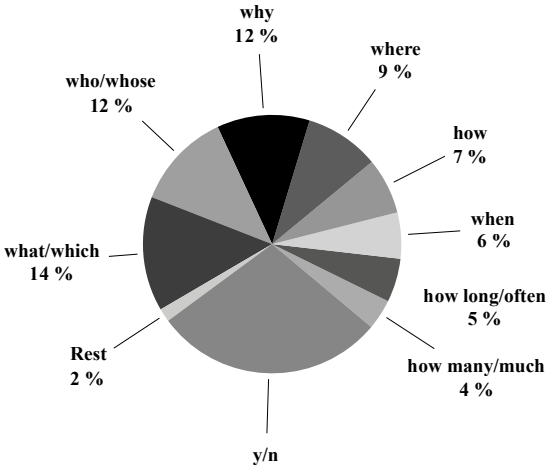

Figure 2: Distribution of question types in MCScript, from Ostermann et al. (2018).

Due to the design of the data acquisition process, a substantial subset of questions (27.4%) require commonsense inference about everyday activities. Figure 2 gives an overview of the distribution of question types on the data. *Yes/No* questions form the largest group, with 29%, followed by questions asking for details of a narration or scenario (*what/which* and *who*).

For the task, the corpus was split into training (9,731 questions on 1,470 texts), development (1,411 questions on 219 texts), and test set (2,797 questions on 430 texts). For 5 scenarios, all texts were held out for the test set, in order to avoid that models overfit and memorize the scenarios in the training data. Texts, questions, and answers contain on average 196.0 words, 7.8 words, and 3.6 words, respectively. There are 6.7 questions per text on average.

[1] www.mturk.com

3.2 Script and Commonsense Knowledge Data

We also encouraged participants to make use of existing script data collections. Thus, we provided several existing collections of script data together with the machine comprehension corpus: DeScript (Wanzare et al., 2016), RKP (Regneri et al., 2010) and the OMCS stories (Singh et al., 2002). The three datasets contain sequences of short, telegraph-styled descriptions of all events that need to be conducted in a scenario (*event sequence descriptions, ESDs*). The data sets contain ESDs for different numbers of scenarios, with a total coverage of over 200 scenarios. The complexity of scenarios varies from simple activities, such as *opening a window*, to more complex ones, such as *attending a wedding*.

For 90 of the 110 scenarios in MCScript, there exist multiple ESDs per scenario in at least one of the 3 script data collections.

We also advised participants to make use of other representations for script knowledge, such as narrative chains (Chambers and Jurafsky, 2008), or event embeddings (Modi and Titov, 2014).

Some participants also made use of *ConceptNet* (Speer et al., 2017) as a resource for commonsense knowledge. ConceptNet is a large-scale knowledge graph that is built from several handcrafted and crowdsourced sources, and that encodes various types of commonsense knowledge.

4 Shared Task Setup

4.1 Evaluation Method

In our evaluation, we measured how well a system was capable of correctly answering questions that may involve commonsense knowledge. As evaluation metric, we used accuracy, calculated as the ratio between correctly answered questions and all questions in our evaluation data. We also evaluated systems with regard to specific question types and based on whether a question is directly answerable, or only inferable from the text.

4.2 Baselines

We provide results of two baseline systems as lower bounds for comparison: a rule-based baseline (*Sliding Window*) and a neural end-to-end system (*Attentive Reader*). Both baselines are described in more detail below. For details about the tuning of hyperparameters, we refer to Ostermann et al. (2018).

Sliding Window

The sliding window baseline is a simple rule-based method that answers a question on a text by predicting the answer option with the highest similarity to the text. The intuition underlying this method is that answers similar to a text should be more plausible than answer options that are different from the text (independent of the question). In our baseline implementation, we compute similarity using a sliding window that compares each answer option to any possible "window" of w tokens of the text. For comparison, each window and each answer is represented by an average vector, computed over the components of word embeddings corresponding to the words in the window and answer, respectively. For each possible window, we compute similarity as the cosine similarity between the window and the answer representation. The answer with the higher maximum similarity (over possible windows) is predicted to be the correct answer.

Attentive Reader

The attentive reader is an established machine comprehension model that reaches good performance e.g. on the *CNN/Daily Mail* corpus (Hermann et al., 2015; Chen et al., 2016). It is a neural network-based approach, which scores answers to a question on a text by finding ("paying attention to") and scoring relevant passages in the text. The scoring and attention mechanisms are learned directly ("end-to-end") from text–question–answer triples, without the need for manual rule writing or feature engineering. As a baseline for the shared task, we use the model formulation by Chen et al. (2016) and Lai et al. (2017), who employ bilinear weight functions to compute both attention and answer-text fit. Bi-directional gated recurrent units (GRUs) are used to encode questions, texts and answers into hidden representations. For a question q and an answer a, the last state of the GRU, $\mathbf{q}$ and $\mathbf{a}$, are used as representations, while the text is encoded as a sequence of hidden states $\mathbf{t}_1 \ldots \mathbf{t}_n$. We compute an attention score s_j for each hidden state $\mathbf{t}_j$ using the question representation $\mathbf{q}$, a weight matrix $\mathbf{W}_a$, and an attention bias b. The text representation $\mathbf{t}$ is computed as a weighted average of hidden

[2] IUCM cluster MCScript texts and try to find answers also in other texts, that are topically similar. In that sense, MCScript itself is used to represent commonsense knowledge.

Rank	Team name	Main model	Commonsense	Other resources	Acc.
1	Yuanfudao	LSTM	ConceptNet	GloVe, Wikipedia, POS and NE tagging	0.84
2	MITRE	LSTM	–	word2vec, Twitter, stemming	0.82
3	Jiangnan	LSTM	–	GloVe, CoVe, POS and NE tagging	0.81
4	ELiRF-UPV	LSTM	ConceptNet	–	0.75
5	YNU_Deep	LSTM	–	GloVe	0.75
6	ZMU	LSTM	–	word2vec, GloVe	0.74
7	ECNU	LSTM	–	GloVe	0.73
8	YNU_AI1799	LSTM/CNN	–	word2vec, GloVe	0.72
9	YNU-HPCC	LSTM/CNN	–	word2vec	0.71
10	CSReader	LSTM	–	lemmatization, GloVe	0.63
11	IUCM	k-means	DeScript, MCScript[2]	NLTK	0.61

Table 1: Overview of techniques and resourced used by the participating systems.

representations:

$$s_j = softmax_j(\mathbf{t}_j^\top \mathbf{W}_a \mathbf{q} + b)$$
$$\mathbf{t} = \sum_j s_j \mathbf{t}_j \quad (1)$$

The probability p of answer a being correct is predicted using another bilinear weight matrix $\mathbf{W}_s$, followed by an application of the softmax function over both answer options for the question:

$$p(a|t,q) = softmax(\mathbf{t}^\top \mathbf{W}_s \mathbf{a}) \quad (2)$$

5 Participants

We ran our shared task through the CodaLab platform[3]. 24 teams submitted results during the evaluation period, out of which 11 teams provided system descriptions: 8 teams from China, and one team each from Spain, Russia and the US. The full leader board containing all 24 submissions can be found on the shared task website.

Except for one team, all participating models rely on recurrent neural network techniques to encode texts, questions and/or answers. The one team that did not apply neural methods proposed an alternative approach based on clustering techniques and scoring word overlap. Only 3 of the 11 teams made explicit use of commonsense knowledge: Two approaches used ConceptNet, either in the form of features extracted from ConceptNet relations or in the form of pretrained *Numberbatch* embeddings (Speer et al., 2017). One participating system made use of script knowledge in the form of event sequence descriptions. Resources commonly used by participants include pretrained word embeddings such as GloVe (Pennington et al., 2014) or word2vec (Mikolov et al., 2013), and preprocessing pipelines such as NLTK[4]. In the following, we provide short summaries of the participants' systems and we give an overview of models and resources used by them (Table 1).

Non-neural methods *IUCM* (Reznikova and Derczynski, 2018) applied an unsupervised approach that assigns the correct answer to a question based on text overlap. Text overlap is computed based on the given passage and text sources of the same topic. Different clustering and topic modeling techniques are used to identify such text sources in MCScript and DeScript.

Neural-network based models Apart from *IUCM*, all participating systems are neural end-to-end models that employ recurrent and/or convolutional neural network architectures. Systems mainly differ with respect to details of the architecture and the form of how words are represented.

Yuanfudao (Liang Wang, 2018) applies a three-way attention mechanism to model interactions between the text, question and answers, on top of BiLSTMs. Each word in a text, question, and answer is represented by a vector of GloVe embeddings and additional information from part-of-speech tagging, name entity recognition, and relation extraction

[3] https://competitions.codalab.org/competitions/17184

[4] https://www.nltk.org/

Rank	Team name	Total	Commonsense	Text	Out of Domain
1	Yuanfudao	**0.84***	**0.82**	**0.85**	**0.79**
2	MITRE	0.82	0.79	0.83*	0.78
3	Jiangnan	0.81*	0.80	0.81*	0.75*
4	ELiRF-UVP	0.75	**0.82**	0.73	0.70
5	YNU_Deep	0.75	0.79	0.73	0.66
6	ZMU	0.74	0.80	0.72	0.66
7	ECNU	0.73	0.77	0.72	0.69
8	YNU_AI1799	0.72	0.76	0.71	0.67
9	YNU_HPCC	0.71*	0.78*	0.69*	0.64*
10	CSReader	0.63	0.64*	0.63	0.59
11	IUCM	0.61	0.54	0.64	0.58
–	Attentive Reader	0.72	0.75	0.71	0.69
–	Sliding Window	0.55	0.53	0.56	0.52
–	Human Performance	0.98			

Table 2: The accuracy of participating systems and the two baselines in total, on commonsense-based questions (*CS*), text-based questions (*TXT*) and on out-of-domain questions (from the 5 held-out testing scenarios). The best performance for each column is marked in **bold print**. Significant differences in results between two adjacent lines are marked by an asterisk (* $p<0.05$) in the upper line. The last line shows the human upper bound (Ostermann et al., 2018) as comparison.

(based on ConceptNet). The model is pretrained on another large machine comprehension dataset, namely the RACE corpus.

MITRE (Merkhofer et al., 2018) use a combination of 3 systems - two LSTMs with attention mechanisms, and one logistic regression model using patterns based on the vocabulary of the training set. The two neural models use different word embeddings - one trained on GoogleNews, another one trained on Twitter, which were enriched with word overlap features. Interestingly, the simple logistic regression model achieves competitive performance and would have ranked 4th as an individual system.

Jiangnan (Xia, 2018) applies a BiLSTM over GloVe and CoVe embeddings (McCann et al., 2017) with an additional attention mechanism. The attention mechanism computes soft word alignment between words in the question and the text or answer. Manual features, including part-of-speech tags, named entitity types, and term frequencies, are employed to enrich word token representations.

ELiRF-UPV (José -Ángel González et al., 2018) employs a BiLSTM with attention to find similarities between texts, questions, and answers. Each word is represented based on Numberbatch embeddings, which encode information from ConceptNet.

YNU_Deep (Ding and Zhou, 2018) test different LSTMs and BiLSTMs variants to encode questions, answers and texts. A simple attention mechanism is applied between question–answer and text–answer pairs. The final submission is an ensemble of five model instances.

ZMU (Li and Zhou, 2018) consider a wide variety of neural models, ranging from CNNs, LSTMs and BiLSTMs with attention, together with pretrained Word2Vec and GloVe embeddings. They also employ data augmentation methods typically used in image processing. Their best performing model is a BiLSTM with attention mechanism and combined GloVe and Word2Vec embeddings.

ECNU (Sheng et al., 2018) use BiGRUs and BiLSTMs to encode questions, answers and texts. They implement a multi-hop attention mechanism from question to text (a Gated Attention Reader (Dhingra et al., 2017)).

YNU_AI1799 (Liu et al., 2018) submitted an ensemble of neural network models based on LSTMs, RNNs, and BiLSTM/CNN combinations, with attention mechanisms. In addition to word2vec embeddings, positional embeddings are used that are generated based on word embeddings.

Rank	Team name	y/n	what	why	who	where	when
1	Yuanfudao	**0.76**	**0.87**	**0.85**	**0.93**	**0.88**	0.81
2	MITRE	**0.76**	0.83	0.82	0.91	0.85	0.77
3	Jiangnan	0.75*	0.80*	0.80	0.88	0.84*	**0.82***
4	ELiRF-UVP	0.72	0.68	0.79	0.86	0.69	0.74
5	YNU_Deep	0.73	0.66	0.75	0.86	0.71	0.72
6	ZMU	0.73	0.65	0.77	0.81	0.72	0.75
7	ECNU	0.71	0.66	0.75	0.82	0.73	0.68
8	YNU_AI1799	0.70	0.68	0.78*	0.80	0.67	0.72
9	YNU_HPCC	0.72*	0.66*	0.71	0.83*	0.65	0.66
10	CSReader	0.54	0.59*	0.68	0.76*	0.62*	0.63
11	IUCM	0.54	0.75	0.66	0.45	0.77	0.61
–	Attentive Reader	0.67	0.66	0.75	0.84	0.73	0.71
–	Sliding Window	0.47	0.69	0.56	0.48	0.61	0.51

Table 3: Accuracy of participating systems and the baselines on the six most frequent question types. The best performance for each column is marked in **bold print**. Significant differences in results between two adjacent lines are marked by an asterisk (* $p<0.05$) in the upper line.

YNU-HPCC (Yuan et al., 2018) use an ensemble of neural networks with stacked CNN and LSTM layers and attention.

CSReader (Jiang and Sun, 2018) use GRUs to encode questions and texts. Answer and text are combined by using an attention mechanism that models soft word alignments, inspired by work on Natural Language Inference (Bowman et al., 2015). Two answer classifiers based on these representations are ensembled for prediction.

6 Results

Tables 2 and 3 give detailed results for all participating systems. We performed pairwise significance tests using an approximate randomization test (Yeh, 2000) over texts. At an accuracy of 84%, the best participating team Yuanfudao performed significantly better ($p<0.05$) than the second best system, MITRE (82%).

Except for *when* questions, Yuanfudao also achieved the best performance at each question type. However, individual differences in results over the 2nd place system were not found to be significant. The top three participating teams, Yuanfudao, MITRE and Jiangnan, all significantly outperform the remaining teams on text-based questions (>80% vs. <74%) as well as on *yes/no*, *what*, *where* and *when* questions.

In comparison to our baselines, all teams but Innopolis significantly outperform Sliding Window. Results of the Attentive Reader are in line with those of the participating systems ranked 7–9: ECNU, YNU_AI1799 and YNU_HPCC. The six top-ranked systems all significantly outperform both of our baselines. On out-of-domain questions, only the top 3 performing models significantly outperform the Attentive Reader baseline, while all models significantly outperform the Sliding Window approach.

For commonsense-based questions as well as for questions on *why* and *who*, results are considerably less consistent: while the top ranked system significantly outperforms teams ranked 7th or lower, most pairwise differences between the top teams are not statistically significant. This implies that the set of correctly answered questions considerably varies between systems, either due to randomness or because they excel at different inference problems.

We found that 19.3 % of the questions in the test set were answered correctly by each participating system. These questions mainly contain text-based questions with an answer that is literally given in the text. Also, there are many commonsense-based questions with a standardized correct answer, as shown in Example 2. Only few of the stories in MCScript cover a long timespan, so the answer to such questions is always similar.

(2) **Q:** How long did it take to pump up the tires?
a. just a few minutes b. a few hours

In contrast, only 1% of questions could not be answered by any of the participating models. Answer-

ing these questions mainly requires complicated inference steps, such as counting or plausibility judgements.

7 Discussion

We briefly highlight some of the findings by the shared task participants.

External knowledge sources. One of the main goals of this shared task was to provide an extrinsic evaluation framework for models of commonsense knowledge. However, only three participants actually made use of resources of commonsense knowledge.

Most prominent is the use of ConceptNet, a large-scale knowledge graph that is built from several handcrafted and crowdsourced sources. It was employed by two of the top 5 scoring models: Yuanfudao use it to learn their own ConceptNet-based relation embeddings. ELiRF-UPV make use of Numberbatch word embeddings, which are learned based on ConceptNet data. Ablation analyses conducted by Yuanfudao indicate that the addition of ConceptNet increases overall accuracy by almost 1% absolute. In contrast, only one participant used crowdsourced script data from the DeScript corpus in their final submission, IUCM. They found that the use of script data, instead of or in addition to texts, improved performance by up to 0.3% absolute.

CSReader tried to extend their neural model with script data from OMCS, but report that it did not result in an improvement.

No participant made use of narrative chains or other forms of structured/learned representations of scripts or events (such as event embeddings).

Pretraining. Most participants made use of pretraining in the form of word embeddings such as word2vec or GloVe, that were build on large data collections. Yuanfudao used the RACE dataset, which is the largest available multiple-choice machine comprehension corpus, for pretraining the complete model for several epochs. In their ablation analysis, they found pretraining to have the largest effect on model performance, with improvements in accuracy of up to 1.4% absolute. This result underlines that the comparably small size of MCScript naturally restricts how much neural approaches can learn from the data without overfitting.

Word representations. For representing tokens, most participants used word2vec embeddings, GloVe embeddings, or combinations thereof. The participating teams used different dimensionality sizes, and some of them refitted the vectors while others did not, leading to differing outcomes for both embedding types. In summary, none of both representations seems to clearly outperform the other.

In contrast, participants consistently found that extending word representations with additional features improves results: For example, Yuanfudao and Jiangnan use predicted part-of-speech tags and named entity information, as well as term frequency, and report improvements of up to 1% absolute in accuracy. Also, some participants report the use of word overlap features. Most notably, MITRE found that a logistic regression classifier based on overlap features can achieve performance competitive with neural approaches.

In general, additional features seem to be beneficial, since they provide more explicit or additional information that can be leveraged by neural networks and other classifiers.

Preprocessing. Several participants reported that lemmatizing and stop word removal further improved their results. A prominent example is the submission by MITRE, who use a stemmer to derive root forms for all words, in order to compute overlap and co-occurrence statistics between answers and text/questions.

8 Conclusions

This shared task provides an evaluation framework for commonsense knowledge in a machine comprehension setting. We create the MCScript corpus, which provides 2,119 stories and 13,939 answers for 110 everyday activities of different complexity. In contrast to previous datasets, MCScript was created in a way that results in a relatively large amount of *common sense questions,* i.e. questions which can not be answered directly from the text but require some form of common-sense knowledge about the scenario under consideration to be answered correctly.

24 teams submitted systems during the the evaluation period of the shared task, of which 11 teams submitted task description papers. The best-performing system achieves an overall accuracy of 84%, which outperforms the two baselines by a

large margin; yet, there gap to the human upper bound (98%) is still relative large.

Although participants were explicitly encouraged to use additional common-sense knowledge resources like DeScript of OMCS, only 3 systems (including the best-performing system) actually used such additional resources. The evaluation results suggest that additional common-sense knowledge is in fact beneficial for overall accuracy. However, the positive effect is relatively small, which might be due to the fact that our dataset has been created in a way that leads to relatively "easy" stories, and that the systems are able to learn a certain amount of common sense knowledge directly from the stories. In future work, it would be interesting to see if the results of our shared task carry over to other, presumably more complex stories like for instance personal blog stories from the Spinn3r corpus (Burton et al., 2011).

Acknowledgments

We thank the reviewers for their helpful comments. Also, we thank all teams for their participation and the effort they put into their submissions and the discussions, making this shared task a success.

This research was funded by the German Research Foundation (DFG) as part of SFB 1102 Information Density and Linguistic Encoding and EXC 284 Multimodal Computing and Interaction.

References

Gordon H Bower, John B Black, and Terrence J Turner. 1979. Scripts in memory for text. *Cognitive psychology*, 11(2):177–220.

Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642.

Kevin Burton, Niels Kasch, and Ian Soboroff. 2011. The icwsm 2011 spinn3r dataset. In *Proceedings of the Annual Conference on Weblogs and Social Media (ICWSM 2011)*.

Nathanael Chambers and Dan Jurafsky. 2008. Unsupervised Learning of Narrative Event Chains. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 789–797, Columbus, Ohio. Association for Computational Linguistics.

Nathanael Chambers and Dan Jurafsky. 2009. Unsupervised Learning of Narrative Schemas and their Participants. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 602–610, Singapore. Association for Computational Linguistics.

Danqi Chen, Jason Bolton, and Christopher D Manning. 2016. A Thorough Examination of the CNN/Daily Mail Reading Comprehension Task. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 2358–2367.

Bhuwan Dhingra, Hanxiao Liu, Zhilin Yang, William Cohen, and Ruslan Salakhutdinov. 2017. Gated-attention readers for text comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1832–1846.

Peng Ding and Xiaobing Zhou. 2018. YNU_Deep at SemEval-2018 Task 11: An Ensemble of Attention-based BiLSTM Model for Machine Comprehension. In *Proceedings of the 12th International Workshop on Semantic Evaluations (SemEval-2018)*.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1693–1701.

Zhengping Jiang and Qi Sun. 2018. CSReader at SemEval-2018 Task 11: Multiple Choice Question Answering as Textual Entailment. In *Proceedings of the 12th International Workshop on Semantic Evaluations (SemEval-2018)*.

José -Ángel González, Lluís-F. Hurtado, Encarna Segarra, and Ferran Pla. 2018. ELiRF-UPV at SemEval-2018 Task 11: Machine Comprehension using Commonsense Knowledge. In *Proceedings of the 12th International Workshop on Semantic Evaluations (SemEval-2018)*.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. 2017. TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada. Association for Computational Linguistics.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. *arXiv preprint arXiv:1704.04683*.

Alessandro Lenci. 2008. Distributional semantics in linguistic and cognitive research. *Italian journal of linguistics*, 20(1):1–31.

Yongbin Li and Xiaobing Zhou. 2018. ZMU at SemEval-2018 Task 11: Machine Comprehension Task using Deep Learning Models. In *Proceedings of the 12th International Workshop on Semantic Evaluations (SemEval-2018)*.

Liang Wang. 2018. Yuanfudao at SemEval-2018 Task 11: Three-way Attention and Relational Knowledge for Commonsense Machine Comprehension. In *Proceedings of the 12th International Workshop on Semantic Evaluations (SemEval-2018)*.

Qingxun Liu, HongDou Yao, and Xiaobing Zhou. 2018. YNU_AI1799 at SemEval-2018 Task 11: Machine Comprehension using Commonsense Knowledge of Different model ensemble. In *Proceedings of the 12th International Workshop on Semantic Evaluations (SemEval-2018)*.

Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. In *Advances in Neural Information Processing Systems*, pages 6297–6308.

Elizabeth M. Merkhofer, John Henderson, David Bloom, Laura Strickhart, and Guido Zarrella. 2018. MITRE at SemEval-2018 Task 11: Commonsense Reasoning without Commonsense Knowledge. In *Proceedings of the 12th International Workshop on Semantic Evaluations (SemEval-2018)*.

Risto Miikkulainen. 1993. *Subsymbolic natural language processing: An integrated model of scripts, lexicon, and memory*. MIT press.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Ashutosh Modi. 2016. Event Embeddings for Semantic Script Modeling. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 75–83, Berlin, Germany. Association for Computational Linguistics.

Ashutosh Modi, Tatjana Anikina, Simon Ostermann, and Manfred Pinkal. 2016. InScript: Narrative texts annotated with script information. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia. European Language Resources Association (ELRA).

Ashutosh Modi and Ivan Titov. 2014. Inducing Neural Models of Script Knowledge. In *Proceedings of the Conference on Computational Natural Language Learning (CoNLL)*, Baltimore, MD, USA.

Ashutosh Modi, Ivan Titov, Vera Demberg, Asad Sayeed, and Manfred Pinkal. 2017. Modeling semantic expectations: Using script knowledge for referent prediction. *Transactions of the Association for Computational Linguistics*, 5:31–44.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California. Association for Computational Linguistics.

Simon Ostermann, Ashutosh Modi, Michael Roth, Stefan Thater, and Manfred Pinkal. 2018. MCScript: A Novel Dataset for Assessing Machine Comprehension Using Script Knowledge. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392.

Lisa F Rau, Paul S Jacobs, and Uri Zernik. 1989. Information extraction and text summarization using linguistic knowledge acquisition. *Information Processing & Management*, 25(4):419–428.

Michaela Regneri, Alexander Koller, and Manfred Pinkal. 2010. Learning Script Knowledge with Web Experiments. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 979–988, Uppsala, Sweden. Association for Computational Linguistics.

Sofia Reznikova and Leon Derczynski. 2018. IUCM at SemEval-2018 Task 11: Similar-Topic Texts as a Knowledge Source. In *Proceedings of the 12th International Workshop on Semantic Evaluations (SemEval-2018)*.

Matthew Richardson, Christopher J.C. Burges, and Erin Renshaw. 2013. MCTest: A Challenge Dataset for the Open-Domain Machine Comprehension of Text. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 193–203, Seattle, Washington, USA. Association for Computational Linguistics.

Rachel Rudinger, Vera Demberg, Ashutosh Modi, Benjamin Van Durme, and Manfred Pinkal. 2015a. Learning to predict script events from domain-specific text. *Lexical and Computational Semantics (* SEM 2015)*, page 205.

Rachel Rudinger, Pushpendre Rastogi, Francis Ferraro, and Benjamin Van Durme. 2015b. Script induction

as language modeling. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1681–1686.

Roger C Schank. 1982. *Dynamic memory: A theory of learning in people and computers*. Cambridge University Press.

Roger C Schank. 1990. *Tell me a story: A new look at real and artificial memory*. Scribner New York.

Roger C Schank and Robert P Abelson. 1975. Scripts, Plans, and Knowledge. In *Proceedings of the 4th international joint conference on Artificial intelligence-Volume 1*, pages 151–157. Morgan Kaufmann Publishers Inc.

Yixuan Sheng, Man Lan, and Yuanbin Wu. 2018. ECNU at SemEval-2018 Task 11: Using Deep Learning Method to Address Machine Comprehension Task. In *Proceedings of the 12th International Workshop on Semantic Evaluations (SemEval-2018)*.

Push Singh, Thomas Lin, Erik T Mueller, Grace Lim, Travell Perkins, and Wan Li Zhu. 2002. Open Mind Common Sense: Knowledge Acquisition from the General Public. In *On the move to Meaningful Internet Systems 2002: CoopIS, DOA, and ODBASE*, pages 1223–1237. Springer.

Robert Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)*, pages 4444–4451.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2017. NewsQA: A Machine Comprehension Dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200.

Lilian Wanzare, Alessandra Zarcone, Stefan Thater, and Manfred Pinkal. 2016. DeScript: A Crowdsourced Database for the Acquisition of High-quality Script Knowledge. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia. European Language Resources Association (ELRA).

Lilian Wanzare, Alessandra Zarcone, Stefan Thater, and Manfred Pinkal. 2017. Inducing Script Structure from Crowdsourced Event Descriptions via Semi-Supervised Clustering. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*, pages 1–11, Valencia, Spain. Association for Computational Linguistics.

Jason Weston, Antoine Bordes, Sumit Chopra, Alexander M Rush, Bart van Merriënboer, Armand Joulin, and Tomas Mikolov. 2015. Towards AI-complete question answering: A set of prerequisite toy tasks. *arXiv preprint arXiv:1502.05698*.

Jiangnan Xia. 2018. Jiangnan at SemEval-2018 Task 11: Attention-Based Reading Comprehension System. In *Proceedings of the 12th International Workshop on Semantic Evaluations (SemEval-2018)*.

Alexander Yeh. 2000. More accurate tests for the statistical significance of result differences. In *Proceedings of the 18th Conference on Computational Linguistics*, volume 2, pages 947–953.

Hang Yuan, Jin Wang, and Xuejie Zhang. 2018. YNU-HPCC at Semeval-2018 Task 11: Using an Attention-based CNN-LSTM for Machine Comprehension using Commonsense Knowledge. In *Proceedings of the 12th International Workshop on Semantic Evaluations (SemEval-2018)*.