Improving RNN-based Answer Selection for Morphologically Rich Languages

Marek Medveď, Radoslav Sabol and Aleš Horák
Natural Language Processing Centre, Faculty of Informatics, Masaryk University, Botanická 68a, 602 00, Brno, Czech Republic

Keywords: Question Answering, Question Classification, Answer Classification, Czech, Simple Question Answering Database, SQAD.

Abstract: Question answering systems have improved greatly during the last five years by employing architectures of deep neural networks such as attentive recurrent networks or transformer-based networks with pretrained contextual information. In this paper, we present the results and detailed analysis of experiments with the largest question answering benchmark dataset for the Czech language. The best results evaluated in the text reach the accuracy of 72 %, which is a 4 % improvement to the previous best result. We also introduce the newest version of the Czech Question Answering benchmark dataset SQAD 3.0, which was substantially extended to more than 13,000 question-answer pairs, and we report the first answer selection results on this dataset which indicate that the size of the training data is important for the task.

1 INTRODUCTION

Comparable evaluation of question answering systems for the mainstream languages, mainly English, is currently well established thanks to very large benchmark dataset, such as the Stanford Question Answering Dataset (SQuAD (Rajpurkar et al., 2018)), consisting of more than 100,000 questions with multiple good answers, or the ReAding Comprehension from Examinations (RACE (Lai et al., 2017)) dataset of again nearly 100,000 questions with 4 candidate answers each. The results using SQuAD currently surpass the Human Performance by nearly 3% reaching more than 92% F1 score. The current best algorithms rely on variations of the transformer-based networks with pretrained contextual information (with BERT (Devlin et al., 2019) being the core algorithm with many improved variations) or attentive recurrent networks, e.g. (Ran et al., 2019). All these approaches rely heavily on a large training dataset available, which inevitably brings a drawback when working with less-resourced languages, potentially with complex underlying morphological structure. In order to test such setup, we are developing the Simple Question Answering Dataset, or SQAD,[1] with thousands of question-answer pairs based on Czech Wikipedia texts supplemented with detailed information related to the Question Answering process (morphological annotation, question and answer types, document selection, answer selection, answer context and answer extraction).

In the current paper, we show the latest results of the answer selection module of the AQA (Medveď and Horák, 2016) question answering system for the Czech language based on the attentive recurrent networks, see Section 2 for details.

In Section 3 we introduce the new extended version of the SQAD database in version 3.0, which consists of more than 13,000 question-answer pairs, and describe the latest answer selection developments evaluated with SQAD in Section 4 that allow us to improve the latest published results of the dataset (Sabol et al., 2018).

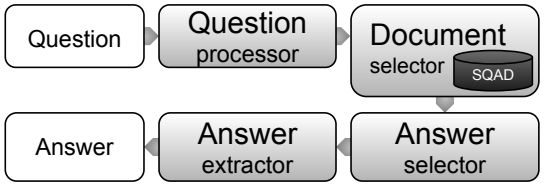

Figure 1: AQA pipeline schema.

[1]The similarity of the name with SQuAD is a mere coincidence, the SQAD naming actually precedes the introduction of the well known Stanford database.

Medveď, M., Sabol, R. and Horák, A.
Improving RNN-based Answer Selection for Morphologically Rich Languages.
DOI: 10.5220/0008979206440651
In *Proceedings of the 12th International Conference on Agents and Artificial Intelligence (ICAART 2020) - Volume 2*, pages 644-651
ISBN: 978-989-758-395-7; ISSN: 2184-433X

2 THE QUESTION ANSWERING PIPELINE

Determining the appropriate answer for a given question is a complicated task which has to go through multiple stages of processing, see the detailed schema of the AQA system in Figure 1. The tasks range from acquiring the essential information about the question and its composition to developed models for identifying adequate text parts that contain the final answer.

The presented AQA system consists of several modules which form a processing pipeline. The pipeline consists of:

- *Question Processing Module:*
  Whenever a question is submitted to the system, the system has to analyze it and acquire all possible information about it. In the question processing module, the input question is transformed into the vertical format that apart from question words consists of their base forms and part-of-speech (POS) tags for each word (for this task we use Majka (Šmerk, 2009) and Desamb (Šmerk, 2010) tools). The base form (or lemma) helps the system to correctly detect the answer sentence where the words can be in different form.[2] The POS tags enable the system to filter out non important words such as punctuation and emphasize important words like verbs, nouns and adjectives.

  A subpart of this module is the question/answer type detection tool. According to the given question, the algorithm computes the question type of the question and the probable expected answer type. The result of this algorithm is than used in the last module where the final answer is provided (see an example in Figure 3). The question-answer type tool was introduced in (Medved' et al., 2019) where the detailed description can be found.

- *Document Selection Module:*
  To be able to provide the final answer, each QA system has to process some kind of underlying knowledge base. The AQA system forms this knowledge base with more than 6,000 articles from the Czech Wikipedia that were manually annotated in the SQAD benchmark dataset with all the information needed for training and testing the question answering process. The new SQAD 3.0 consists of 13,473 records (see Section 3.1 for details), where each record contains the information about: the question, the original full size Wikipedia article, the answer selection[3] and the answer extraction.[4]

  The document selection module processes the original full size Wikipedia texts to identify the (most probable) documents to be searched for the exact answer. According to the information extracted from the input question the module is able to go through all the texts in the knowledge base and rank them according to the input question relevance. In detail, the module computes combined TF-IDF scores with the base word forms of both the question and the full text. The module offers document ordering with the possibility to choose *N* best candidates that are highly related to the information required by the question. These *N* documents are then passed to next module for further processing. In the whole processing pipeline, this module is the only one that communicates with the complete knowledge base in the current system implementation (as can be seen in Figure 1).

- *Answer Selection Module:*
  After discovering the candidate documents in the document selection module, the system can proceed to the next level of processing. In the answer selection module, the document or documents with the high score(s) are searched for sentences that contain relevant answer to the input question.

  This module incorporates a neural network model that is based on attentional bidirectional GRU network. Before the model deployment in the answer selection module, it has to go through a learning phase where the model is trained on [question, answer-selection] tuples from the SQAD database training set. The resulting model is than incorporated in the answer selection module where it provides a list of candidate answers. Detail description of the AQA's answer selection network is presented in Figure 2.

- *Answer Extraction Module:*
  The final phase of the AQA system is performed in the answer extraction module. This module takes the best scored candidate answer sentence from the previous step and extracts an adequate part of this sentence that is suitable for answering the asked question. The extracted part is the shortest possible but with sufficient amount of information to satisfy the user question. This is the

[2] Czech is a fusional language that has rich system of morphology and relatively flexible word order.

[3] One sentence from the original text that contains the answer.

[4] A part of the answer selection sentence that is the shortest possible answer to the given question with enough information for the user.

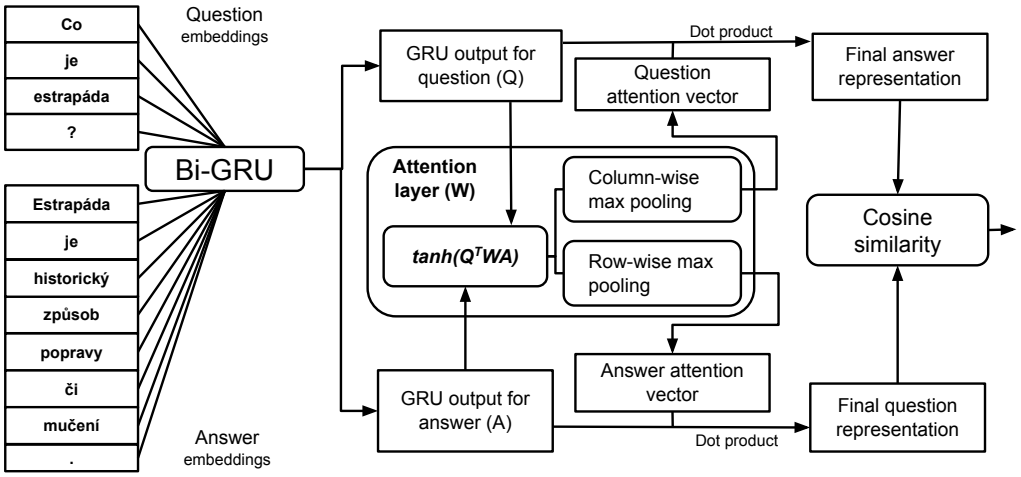

Figure 2: The AQA answer selection architecture with an example question: "Co je estrapáda?" (What is a strappado?) and the answer: Estrapáda je historický způsob popravy či mučení. (Strappado is a historical form of execution or torture.)

Question:	Kdo se narodil ve Stratfordu nad Avonou? (Who was born in Stratford Upon Avon?)
Answer selection:	Shakespeare se narodil a vyrůstal ve Stratfordu nad Avonou, v anglickém hrabství Warwickshire. (Shakespeare was born and raised in Stratford-upon-Avon, Warwickshire, England.)
Answer :	Shakespeare
Question type:	PERSON
Answer type:	PERSON

Figure 3: Question-answer type example.

final part of the processing pipeline and the result of it is provided to the user as the final answer. In the current system version this module is based on set of rules that have been presented in (Medved' and Horák, 2016).

3 EXPERIMENTS

3.1 The Benchmark Dataset

The development and evaluation process of a question answering system requires a benchmark dataset with adequate amount of records. For the case of the Czech language, we have developed the Simple Question Answering Database (SQAD) that has been first introduced in 2014 (see (Horák and Medved', 2014)) and is constantly being improved through multiple modifications (SQAD v2 and v2.1, (Sabol et al., 2018) and (Šulganová et al., 2017)).

This benchmark dataset is based on Czech Wikipedia articles harvested and annotated by human annotators. The harvesting process has multiple stages. The first stage is fully automatic where, according to the article name chosen by the annotator for processing, the article full text is downloaded using the Wikipedia API. Then the obtained text is tokenized, lemmatized and tagged with detailed morphological information (Šmerk, 2009). After this stage, the annotator designs a question suitable for the selected Wikipedia article. According to the question, the annotator identifies the appropriate question and expected answer types. The final stage consists of picking up a sentence from the text that contains the answer (the list of sentences from the article is provided by the system and the annotator only chooses the correct one) and selecting the appropriate part of this sentence as the final answer.

The latest version of the SQAD dataset is 3.0 which differs from the previous one in several ways:

- *Number of records:* The new SQAD v3.0 is larger than all previous versions. It contains 13,473 records (question-answer pairs), which is almost 5,000 more than in SQAD v2.1. For fine-grained statistics about the new version see Table 1.
- *Answer Context:* For questions that are not fully answered within one sentence due to anaphoric references, SQAD 3.0 contains a new information about the answer selection context. This information covers multiple sentences containing both the anaphora and its antecedent. Currently, there are 378 of such contexts present in SQAD v3.

Table 1: SQAD v3 statistics.

No. of records		13,473	
No. of different articles		6,571	
No. of tokens (words)		28,825,824	
No. of answer contexts		378	
Q-Type :		A-Type :	
DATETIME	14.7 %	DATETIME	14.6 %
PERSON	13.1 %	PERSON	13.2 %
VERB_PHR	16.8 %	YES_NO	16.8 %
ADJ_PHR	11.2 %	OTHER	16.7 %
ENTITY	18.4 %	ENTITY	13.1 %
CLAUSE	3.5 %	NUMERIC	7.4 %
NUMERIC	7.3 %	LOCATION	12.3 %
LOC	12.4 %	ABBR	2.4 %
ABBR	2.5 %	ORG	2.1 %
OTHER	0.1 %	DENOTATION	1.4 %

- *Answer vs Answer Extraction:* The previous versions of SQAD used an annotation process which allowed to adjust the exact answer by the annotators. While the answer at its essence was correct, it was difficult to automatically check the consistency of the exact answer with the answer selection sentence. Since the Czech language is almost free word order and it is highly flective, the annotator's answer in many times differed from the actual content of the answer selection sentence. Because of this inconsistency, the new version distinguishes the *answer* and the *answer extraction* result as the human made answer and the exact subphrase of the answer selection sentence.
- *Raw Text vs Vertical Format:* In the first version, SQAD contained strictly only the plain text versions of the question, the answer and the text. To improve handling automatic errors that appear in tokenization, lemmatization or tagging (that have to be manually corrected), SQAD now stores all the text in the annotated (vertical) format as the main data source. For purposes where the plain text form is required, the plain text is automatically extracted from the structured form.

3.2 Document Selection

As we have described in Section 2, the document selection module is a crucial part of the whole AQA system. The core part of this module is implemented using the *Gensim*[5] library (Řehůřek and Sojka, 2010).

The algorithm starts with indexing the whole corpus using the *corpora.dictionary* module (this step is important for the time efficiency of the algorithm). Then all documents are transformed into a matrix representation by using the *corpora.mmcorpus* module and a TF-IDF score is computed by *models.tfidfmodel* module. The final step is to build a similarity matrix using *similarities.docsim* module.

The final document selection process takes the question and the query similarity matrix to obtain a ranked list of the top N most similar documents according to the question content.

In the current version, the resulting score is obtained by parametric weighting of the TF-IDF score. The purpose of this feature is to allow putting more emphasis on words that appear in the question and thus shift the final score of relevant documents up in the resulting ranked list. A detailed evaluation of the parametric weight settings is presented in Section 4.1

3.3 Answer Selection

The answer selection module utilizes a deep neural network architecture, which was originally proposed in (Santos et al., 2016). Major parts of this module were implemented using the *PyTorch* neural network framework and open source machine learning module (Paszke et al., 2017).

As an input, the answer selection module receives the question and a single candidate answer – both of which are in form of pre-trained 100-dimensional word embeddings trained on a large Czech corpus csTenTen17 (Jakubíček et al., 2013) following the schema in Figure 2. The input is forwarded through a *biGRU (bidirectional Gated Recurrent Unit)* layer which allows to learn contextual features of the input. As the next step, the network uses two-way attention mechanism that highlights important words in the question with the respect to the candidate answer and vice versa. The attention vectors are then combined with *GRU* outputs for both the question and the candidate answer. The final step computes a confidence score by measuring the cosine similarity between those two vectors. The whole network is then trained with a *hinge loss* maximizing the similarity with correct answers and dissimilarity with (a sample of) incorrect answers.

The latest development consist mainly in intensive experiments regarding hyper-parameter optimizations, different approaches to sampling and filtering available data. All of those were performed on the SQAD v2.1 benchmark dataset partitioned into predefined balanced train (*approx. 50%, 4,271 entries*), validation (*approx. 10%, 889 entries*) and test set (*approx. 40%, 3,406 entries*), which remained consistent between the experiments to create comparable results. The results of these experiments are presented in Section 4.2.

Preliminary experiments were also performed

[5]https://radimrehurek.com/gensim/

with the latest version of the SQAD database, SQAD v3.0. The training set has been allocated to *approx 60%, 8,061 entries*, the validation set remains as *aprrox 10%, 1,403 entries*, and the remaining 4,014 entries (*30%*) are used as the test set.

3.4 Tag Filtering

One set of experiments regards filtering the input tokens of the neural network with the intention to reduce noise during the training and evaluation. These experiments were based on filtering rules based on the part-of-speech tags of question and answer tokens:

- removing all punctuation;
- keeping only the nouns;
- keeping nouns and adjectives;
- keeping nouns, adjectives, verbs; and
- keeping nouns, adjectives, numerals, and verbs.

These filters were applied in both training and evaluation of the model. Only the best hyperparameters were used, and the run was repeated three times for each filter, resulting in 15 produced models. See the next section for a detailed evaluation of the results.

4 EVALUATION

4.1 Document Selection

The new version of the document selection module was compared with the original implementation and the best settings of new TF-IDF weighting feature were determined.

The evaluation of the original document selection approach is presented in Table 2 together with a comparison of the new results. The first column represents the position of document that has to be selected for the correct answer from the document selection ranked list. The second column represents the number of matches and the third column is the percentage representation of this number.

Table 2: A comparison of the original document selection module with the new weighted document selection module.

	Original DS		New DS weight = 0.24	
1	9,872	73.27 %	10,305	76.49 %
2	1,260	9.35 %	1,367	10.15 %
3	541	4.02 %	483	3.58 %
4	297	2.20 %	279	2.07 %
5	191	1.42 %	169	1.25 %
Not found	1,312	9.74 %	870	6.46 %

The main difference between the original and new approach is better displayed when we combine the first 5 top ranked documents, see Table 3, where the fist column represents the range of positions in the ranked list and the second column represents the match in percents. The new implementation outperforms the original one by **4.01%** at the first position and by **3.28%** with the first five documents.

Table 3: Combined Document selection with top 5 best scored documents.

	Original	Combined score (new weight = 0.24)
1:2	82.62 %	86.63 %
1:3	86.64 %	90.22 %
1:4	88.84 %	92.29 %
1:5	90.26 %	93.54 %

Table 4 offers an overview of multiple settings of the TF-IDF weight. The comparison between SQAD v3.0 and SQAD V2.1 on the document selection level is demonstrated in Figure 4.

Table 4: Document selection with multiple TF-IDF weight settings.

W	1:2	1:3	1:4	1:5
0	76.34	81.46	84.35	86.41
0.1	84.29	88.77	91.21	92.62
0.2	86.36	90.10	92.20	93.49
0.24	**86.63**	**90.22**	**92.29**	**93.54**
0.25	**86.66**	**90.15**	**92.28**	**93.54**
0.26	**86.68**	**90.17**	**92.26**	**93.54**
0.3	86.74	90.24	92.19	93.38
0.4	86.33	89.91	91.85	92.99
0.5	85.64	89.14	91.21	92.38
0.6	84.96	88.54	90.66	91.94
0.7	84.34	87.98	90.12	91.40
0.8	83.66	87.43	89.58	90.94
0.9	83.26	87.04	89.19	90.57
1	82.62	86.64	88.84	90.26

4.2 Answer Selection

Since the last published results (Sabol et al., 2018), new combinations of hyperparameter settings were evaluated. Improvements in the implementation include also performance optimizations of critical sections, which allowed to reduce the average experiment running time from 4 hours to 2.5 hours when measured with NVIDIA GeForce GTX 1080 Ti GPU.

The hyperparameter settings include extending the original hidden layer vector dimensions from the range between *200* and *280* to *100, 200, 300, 400, 500*. The learning rate parameter values were extended to *0.1, 0.2, 0.25, 0.4, 1.1, 1.2, 1.3*. The dropout values did not change from *0.2, 0.4, 0.6* as outside values drastically degraded the accuracy of models. All

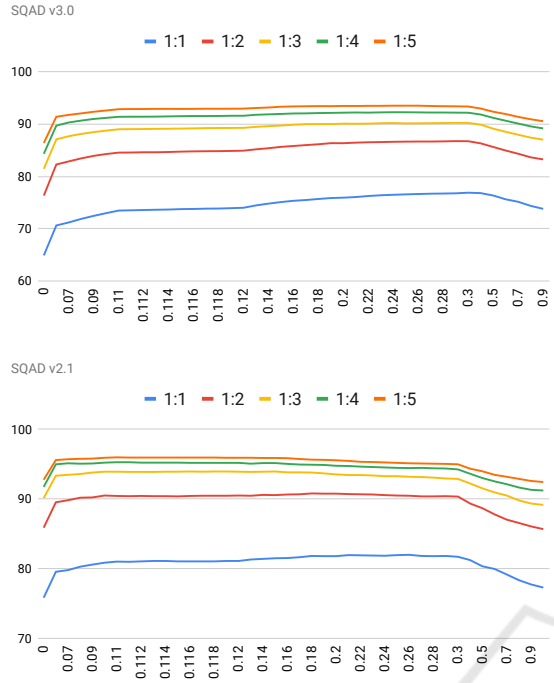

Figure 4: Comparison of SQAD v2.1 and SQAD v3.0 on the document selection level (combined score on top 5 documents).

Table 5: The answer selection results for various hyperparameter settings in descending order (sorted by MAP). The last line shows the configuration of the old best hyperparameters.

Hidden size	Initial learning rate	Dropout	MAP	MRR
400	0.4	0.2	**72.36**	80.12
500	0.4	0.2	71.11	80.11
300	0.4	0.2	71.1	79.91
200	0.4	0.2	70.97	79.86
100	0.4	0.2	70.32	79.39
400	0.25	0.2	70.02	79.01
500	0.25	0.2	70.02	79.17
400	0.4	0.4	69.72	79.07
260	0.2	0.2	68.29	-

the models were trained on 25 epochs choosing the best results based on the evaluation with the validation set.

The results are presented using the *Mean Average Precision* (MAP) and *Mean Reciprocal Rank* (MRR) measures as it is common with the answer selection algorithms. Each run was repeated 3 times, so the total MAP/MRR is averaged between those models. Overall, 315 models were trained, and the overall accuracy of the new best hyper-parameter combination is **72.36 %**, which is a **4.07%** improvement from the previously published best result of 68.29%.

Table 5 shows that varying the hidden layer dimension has only a limited effect on the overall accuracy of the model. One of the possible explanations is that the model cannot fully utilize the amount of features for a relatively small dataset of SQAD 2.1. As can be seen in Figure 5, both the learning rate and the dropout play much more significant role in the performance.

The next Table 6 summarizes the results of the tag filtering experiment. As was already mentioned in Section 3.4, only the best hyperparameter combination was used for these runs. The accuracy decreases significantly for each filter, and the decrease ratio depends on how restrictive the filter is. Removing the punctuation decreased the accuracy by approximately 4 percent, proving that even punctuation can play relevant role in this task. Some of the more restrictive filters can even take out important parts of the candidate answer like numeric data for which the question asks. As a positive side effect, filtering noticeably increases performance, as it is cheaper in computational resources to throw some information away instead of passing them through the entire network.

Table 6: The answer selection results for various tag filtering settings.

Experiment	MAP
Punctuation removed	67.09
Nouns, adjs, numerals and verbs only	64.00
Nouns, adjectives and verbs only	61.66
Nouns and adjectives only	49.54
Nouns only	43.36

4.3 Discussion and Error Analysis

4.3.1 Document Selection

An in-depth evaluation of the document selection module has unveiled imbalances between document selection accuracies per question type (see Table 7). These experiments show that the least precise results of the approach are connected with the question types of *ENTITY*, *LOCATION* and *PERSON* as the accuracies of these types of questions never reach more than 91% accuracy at any TF-IDF weighting setup. These 3 kinds of questions are often represented by histor-

Table 7: Document selection accuracy per question type with the TF-IDF weight set to 0.24.

Question t.	Accuracy	Question t.	Accuracy
ABBREV.	92.22	LOCATION	**90.88**
ADJ_PHR.	96.82	NUMERIC	92.16
CLAUSE	95.39	OTHER	100.00
DATETIME	95.91	PERSON	**90.32**
ENTITY	**90.46**	VERB_P.	97.49

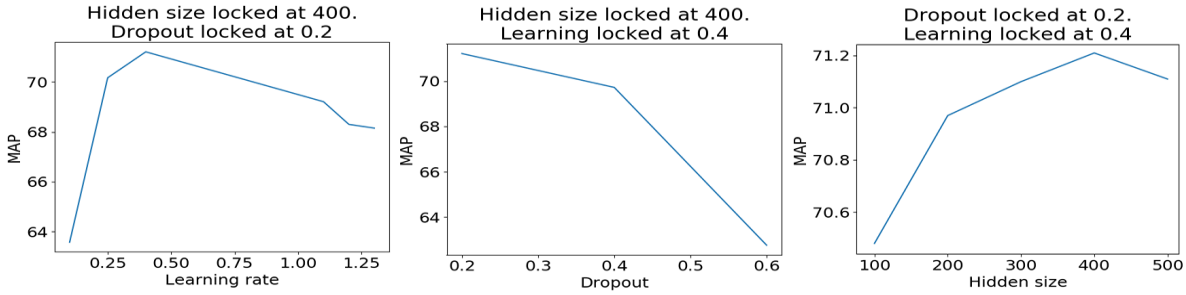

Figure 5: Impact of the answer selection model parameters on the overall accuracy.

ical and real-world facts, which are connected with more than one topical document in Wikipedia.
A review of the weighted document selection results also demonstrates that each question type has it own optimum of the weight. Table 8 enlists the best TF-IDF weighting setting for each question type.

Table 8: The best TF-IDF weights in the document selection per question type.

Question type	Best TF-IDF weight
ABBREVIATION	0.117–0.13
ADJ_PHRASE	0.24–0.25
CLAUSE	0.18–0.3
DATETIME	0.24
ENTITY	0.25; 0.28–0.3
LOCATION	0.18
NUMERIC	0.14
OTHER	0.06–0.8
PERSON	0.26–0.27
VERB_PHRASE	0.17; 0.26–0.28

4.3.2 Answer Selection

Detailed error analysis was performed with one of the best evaluated models (reaching the accuracy of 72.36%). Table 9 shows the percentages of questions, where the correct answer was found in the specified ranked position of 1–10. Most of the incorrectly answered questions end up in position 2, that is why the future work is planned to be directed into fine-tuning the selection parameters in order to distinguish between these top 2 answers.

Table 9: Answer selection Precision at k .

Pos.	Count	%	Pos.	Count	%
1	2,432	71.42	6	41	1.2
2	360	10.57	7	28	0.82
3	152	4.46	8	26	0.76
4	96	2.82	9	17	0.5
5	62	1.82	≥ 10	191	5.61

Tables 10 display the answer selection results for each type of the question or answer. For questions, the types of *ENTITY* and *CLAUSE* are the most complicated. The class of *OTHER* contains only 4 questions in SQADv2, which is too little to make a noticeable difference. As for the answer types, *ENTITY* and *OTHER* achieve noticeably worse accuracy than the other classes. Unlike with question types, the OTHER answer type has enough questions in the benchmarking dataset for the result to be significant.

Table 10: Answer selection accuracy for various question and answer types.

Question type	MAP	Answer type	MAP
NUM	68.68%	NUM	68.77%
VERB_PHR	70.05%	YES_NO	70.05%
DATETIME	74.93%	DATETIME	74.90%
PERSON	70.02%	PERSON	70.19%
ENTITY	**65.85**%	ENTITY	**61.86**%
CLAUSE	**48.94**%	ORG	75.29%
ADJ_PHR	69.23%	DENOTATION	67.50%
LOCATION	80.07%	LOC	79.87%
ABBR	90.62%	ABBR	90.62%
OTHER	**25.0**%	OTHER	**64.31**%

Figure 5 depicts the sensitivity of the network to varying values of the 3 selected hyperparameters – the learning rate, the dropout of the biGRU layer and the hidden layer vector dimension. In all charts, the values of the remaining variables were fixed at the best resulting values.

Evaluation with SQADv3 proceeded similarly to SQADv2. However, the best hyper-parameter values were identified as the hidden size 300, dropout 0.2, and the learning rate 0.6. With these parameters, the answer selection module has reached the values of **78.87** MAP and **85.94** MRR, which is substantially higher than for SQADv2. The detailed error analysis of the SQADv3 results is yet to be performed.

5 CONCLUSIONS AND FUTURE WORK

In the paper, we have presented a summary and a detailed evaluation of experiments that lead to an improvement of 4% in the document selection module

(with more than 90% accuracy at the top 3 documents) and 4% in the answer selection module of the AQA question answering system for the Czech language with the final Mean Average Precision of 72%.

We have also introduced the latest version of the SQAD question answering benchmark dataset, which now offers more than 13,000 richly annotated question-answer pairs. The evaluation of the system with this enlarged dataset indicates that the size of the training set allows the approach to be more specific in identifying the correct answer when the current best accuracy reaches almost 79% with SQAD 3.0.

In the future work, the development will focus on analysis of the broader context of the answers, with evaluation based both on the preprocessing steps as well as employing the new transformer-based networks. The results of the detailed error analysis also direct the future improvements to processing particular question and answer types with specifically adapted parameter values.

ACKNOWLEDGEMENTS

This work has been partly supported by the Czech Science Foundation under the project GA18-23891S.

Access to computing and storage facilities owned by parties and projects contributing to the National Grid Infrastructure MetaCentrum provided under the programme "Projects of Large Research, Development, and Innovations Infrastructures" (CESNET LM2015042), is greatly appreciated.

REFERENCES

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the NAACL 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Horák, A. and Medveď, M. (2014). SQAD: Simple Question Answering Database. In *Eighth Workshop on Recent Advances in Slavonic Natural Language Processing, RASLAN 2014*, pages 121–128, Brno. Tribun EU.

Jakubíček, M., Kilgarriff, A., Kovář, V., Rychlý, P., and Suchomel, V. (2013). The tenten corpus family. *7th International Corpus Linguistics Conference CL 2013*.

Lai, G., Xie, Q., Liu, H., Yang, Y., and Hovy, E. (2017). RACE: Large-scale ReAding Comprehension Dataset From Examinations. In *Proceedings of EMNLP 2017*, pages 785–794.

Medveď, M. and Horák, A. (2016). AQA: Automatic Question Answering System for Czech. In Sojka, P. et al., editors, *Text, Speech, and Dialogue, TSD 2016*, pages 270–278, Switzerland. Springer.

Medveď, M., Horák, A., and Kušniráková, D. (2019). Question and answer classification in czech question answering benchmark dataset. In *Proceedings of the 11th International Conference on Agents and Artificial Intelligence, Volume 2*, pages 701–706, Prague, Czech Republic. SCITEPRESS.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. (2017). Automatic differentiation in PyTorch. In *NIPS Autodiff Workshop*.

Rajpurkar, P., Jia, R., and Liang, P. (2018). Know What You Don't Know: Unanswerable Questions for SQuAD. In *Proceedings of the ACL 2018 (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Associationfor Computational Linguistics.

Ran, Q., Li, P., Hu, W., and Zhou, J. (2019). Option comparison network for multiple-choice reading comprehension. *arXiv preprint arXiv:1903.03033*.

Řehůřek, R. and Sojka, P. (2010). Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA.

Sabol, R., Medveď, M., and Horák, A. (2018). Recurrent networks in aqa answer selection. In Aleš Horák, P. R. and Rambousek, A., editors, *Proceedings of the Twelfth Workshop on Recent Advances in Slavonic Natural Languages Processing, RASLAN 2018*, pages 53–62, Brno. Tribun EU.

Santos, C. d., Tan, M., Xiang, B., and Zhou, B. (2016). Attentive pooling networks. *arXiv preprint arXiv:1602.03609*.

Šmerk, P. (2009). Fast Morphological Analysis of Czech. In *Proceedings of Recent Advances in Slavonic Natural Language Processing, RASLAN 2009*, pages 13–16.

Šulganová, T., Medveď, M., and Horák, A. (2017). Enlargement of the Czech Question-Answering Dataset to SQAD v2.0. In *Proceedings of Recent Advances in Slavonic Natural Language Processing, RASLAN 2017*, pages 79–84.

Šmerk, P. (2010). *K počítačové morfologické analýze češtiny (in Czech, Towards Computational Morphological Analysis of Czech)*. PhD thesis, Faculty of Informatics, Masaryk University.