ConvKN at SemEval-2016 Task 3: Answer and Question Selection for Question Answering on Arabic and English Fora

Alberto Barrón-Cedeño[†], **Daniele Bonadiman**[‡], **Giovanni Da San Martino**[†],
Shafiq Joty[†], **Alessandro Moschitti**[†], **Fahad A. Al Obaidli**[†],
Salvatore Romeo[†], **Kateryna Tymoshenko**[‡], **Antonio Uva**[‡]
[†]ALT group, Qatar Computing Research Institute, Hamad Bin Khalifa University, Qatar
[‡]Department of Computer Science and Information Engineering, University of Trento, Italy
{albarron,gmartino,sjoty,faalobaidli,amoschitti,sromeo}@qf.org.qa
{d.bonadiman,kateryna.tymoshenko,antonio.uva}@unitn.it

Abstract

We describe our system, ConvKN, participating to the SemEval-2016 Task 3 "Community Question Answering". The task targeted the reranking of questions and comments in real-life web fora both in English and Arabic. ConvKN combines convolutional tree kernels with convolutional neural networks and additional manually designed features including text similarity and thread specific features. For the first time, we applied tree kernels to syntactic trees of Arabic sentences for a reranking task. Our approaches obtained the second best results in three out of four tasks. The only task we performed averagely is the one where we did not use tree kernels in our classifier.

1 Introduction

SemEval-2016 Task 3 challenged the participants on the different steps of the full task of Community Question Answering (cQA).[1] Given a set of existing forum questions Q, where each existing question $q \in Q$ is associated with a set of answers C_q, and a new user question q', the ultimate task is to determine whether a comment $c \in C_q$ represents a pertinent answer to q' or not. This task can be subdivided into three tasks, namely: (*A*) to assign a relevance (*goodness*) score to each answer $c \in C_q$ with respect to the existing question q; (*B*) to re-rank the set of questions Q according to their relevance against the new question q'; and finally (*C*) to predict the appropriateness of the answers $c \in C_q$ against q'.

Task 3 included these three tasks for English, whereas an adaptation of Task C was proposed for Arabic (Task D). The reader can refer to (Nakov et al., 2016) for a more detailed description of the tasks. Task A was also proposed in the SemEval-2015 edition (Nakov et al., 2015).[2]

We designed systems for all tasks. We used the feature vectors designed by Barrón-Cedeño et al. (2015) and Nicosia et al. (2015) for tasks A, B and C, whereas we just used a basic feature vector derived from the system of Belinkov et al. (2015) for Task D.

Most importantly, for tasks A, B and D, we combined feature vectors with tree kernels (Moschitti, 2006) for relational learning from short text (Moschitti et al., 2007; Moschitti, 2008). In particular, we used the improved models that have been successful applied for several tasks and datasets in standard QA, see for example, (Severyn and Moschitti, 2012; Severyn and Moschitti, 2013; Severyn et al., 2013b; Severyn et al., 2013a; Tymoshenko et al., 2014; Tymoshenko and Moschitti, 2015).

Additionally, we used Convolutional Neural Networks (CNNs) (Severyn and Moschitti, 2015) and combined them with vectors and tree kernels for Task A as we did in (Tymoshenko et al., 2016).

We acknowledge that the automatic feature engineering of tree kernels was very useful to tackle the new challenges of the SemEval-2016 Task 3. Indeed, all our three systems using relational models based on tree kernels achieved the second official

[1] http://alt.qcri.org/semeval2016/task3

[2] Note that in that paper the naming convention is slightly different. The fresh user question and the forum question are called "original" and "related", respectively.

position. In contrast, for Task C, we did not have time for using the relational model in our submitted system, this has probably caused our average performance in such task, i.e., our system was ranked at the eighth position. For similar reasons, we could apply CNNs to only Task A.

The rest of the paper is organized as follows. Section 2 describes the four CQA tasks and gives a brief overview of the corpora. Section 3 describes the features used. Section 4 discusses our models and our official results. Section 6 presents final remarks.

2 Tasks Description

In this section we sketch the four proposed tasks.

Task A: Question–Comment Similarity. Given a user question and a thread of ten comments associated with it, re-rank the comments in the thread according to their pertinence. Three classes exist in this case: (*i*) `good`: the comment is definitively relevant; (*ii*) `potentially useful`: the comment is not good, but it still contains related information worth checking; and (*iii*) `bad`: the comment is irrelevant (e.g., it is part of a dialogue or unrelated to the topic). For evaluation purposes, both `potentially useful` and `bad` comments were considered as `bad`.

Task B: Question–Question Similarity. Given a new question and a set of ten forum questions, re-rank the forum questions by assessing if they are (*i*) `perfect match`: the new and forum questions request roughly the same information, (*ii*) `relevant`: the new and forum questions ask for similar information, or (*iii*) `irrelevant`: the new and forum questions are completely unrelated. For evaluation purposes, both `perfect match` and `relevant` forum questions are considered as `relevant`.

Task C: New Question–Comment Similarity. Similar to task A, but in this case the relevance of one-hundred comments is assessed against a new out-of-the-forum question. Same evaluation considerations as in task A apply.

Task D: Question–{Forum Question+Comment}. A new question and a set of thirty forum question–answer pairs are provided (it is known in advance that the answer is correct with respect to the forum question). Re-rank the question+comment pairs according to three classes: (*i*) `direct`: a direct answer to the new question; (*ii*) `relevant`: not a direct answer to the question but with information related to the topic; and (*iii*) `irrelevant`: an answer to another question, not related to the topic. For evaluation purposes, both `direct` and `relevant` forum questions are considered as `good`.

Tasks A, B, and C use English instances extracted from *Qatar Living*, a forum for people to pose questions on multiple aspects of daily life in Qatar.[3] Task D uses Arabic instances extracted from three medical fora: *webteb*, *altibbi*, and *consult islamweb*.[4]

As this is a reranking task, mean average precision (MAP) is the referring evaluation metric. We also evaluate our models in terms of average Recall (AvgRec), Precision (P), Recall (R), F-measure (F_1), and Accuracy.

Further details about the corpora, evaluation and other settings can be found in (Nakov et al., 2016).

3 Approach

In order to re-rank the comments according to their relevance, either against the forum questions or against the new questions, we train a binary SVM classifier and use its score as a measure of relevance. The classifier uses partial tree kernels (Moschitti, 2006) defined over shallow syntactic trees, along with other numeric features.

We used the DKPro Core toolkit (Eckart de Castilho and Gurevych, 2014)[5] for pre-processing the texts in English. More precisely, we used OpenNLP's tokenizer, POS-tagger and chunk annotator[6], and Stanford's lemmatizer (Manning et al., 2014), all accessible through DKPro Core.

We used the MADAMIRA toolkit (Pasha et al., 2014) for segmenting Arabic texts. In order to split the texts into sentences, we used the Stanford splitter.[7] For parsing Arabic texts into syntactic trees, we

[3] `http://www.qatarliving.com/forum`

[4] `https://www.webteb.com/`, `http://www.altibbi.com/`, and `http://consult.islamweb.net`.

[5] `https://dkpro.github.io/dkpro-core/`

[6] `https://opennlp.apache.org/`

[7] `http://stanfordnlp.github.io/CoreNLP`

used the Berkeley parser (Petrov and Klein, 2007). Following, we briefly describe the numeric features used in different tasks.

3.1 SemEval-2015 Features

For English texts, we consider three kinds of similarity measures: lexical, syntactic, and semantic (Barrón-Cedeño et al., 2015; Nicosia et al., 2015)

In the case of Task A, the context of a comment is a relevant factor. Comments are organized sequentially according to the time line of the comment thread. Important factors to assess the value of a comment is whether the thread includes further comments by the person who originally asked the question, if the same user is behind various comments in the thread, or what forum category the thread belongs to. Therefore, we consider a set of features that try to describe a comment in the context of the entire thread. Other Boolean context features characterize different situations including the identification of potential dialogues, which usually represent a bunch of `bad` comments, or the position of the comment in the thread. We also considered the categories of the questions in the forum (as some of them tend to include more open-ended questions and even invite for discussion on ambiguous topics), as well as the occurrence of specific strings or the length of a comment. In-depth descriptions of these features are available in (Nicosia et al., 2015).

For Arabic texts, we utilize the embedding vectors as obtained by Belinkov et al. (2015): employing word2vec (Mikolov et al., 2013) on the Arabic Gigaword corpus (Parker et al., 2011). More specifically, we concatenate the vectors representing a new question and an existing question in the question–answer pair, which is then fed to the SVM classifier.

3.2 Rank Feature

The meta-information in the English corpus includes the position of the forum threads in the rank generated by the Google search engine for a given new question. We exploit this information in tasks B and C. We employ the inverse of such position as a feature and refer to it as the rank feature.

3.3 Tree Kernels

Tree kernels are similarity functions that measure the similarity between tree structures. We constructed a syntactic tree for each comment or question. Each task involves a pair of trees, question and comment (tasks A and C) and new and forum questions (tasks B and D). Replicating Severyn and Moschitti (2012), we link the two trees by connecting (i) part-of-speech nodes with a lexical match between the corresponding non-stop words; and (ii) chunk nodes such as NP, PP, VP, when there is a link above between POS-tags. Such links are marked with the presence of a specific tag. We then apply the partial tree kernel (PTK) or the syntactic tree kernels[8] (STK) both defined in (Moschitti, 2006) on the pairs as:

$$K((t_1, t_2), (u_1, u_2)) = TK(t_1, u_1) + TK(t_2, u_2), \qquad (1)$$

where t and u are parse trees extracted from the text pair, i.e., either question and comment for task A or question and question for tasks B and D.

4 Submissions and Results

We describe our primary submissions for the four tasks in Section 4.1. The contrastive submissions are discussed in Section 4.2. Table 1 shows our official competition results for both primary and contrastive submissions.

In all submissions we employed Support Vector Machines (SVM) (Joachims, 1999) using either SVM-Light (Joachims, 1999), KeLP[9] (Filice et al., 2015), or SVM-light-TK[10] (Moschitti, 2006) (only the last two can handle tree kernels).

4.1 Primary Submissions

Task A. The submission consists in an SVM operating on two kernels: (*i*) the tree kernel described in Section 3.3, applied to the structures described by Tymoshenko and Moschitti (2015) without question and focus classification; (*ii*) a polynomial kernel of degree 3 applied to the feature vector that is a concatenation of the feature vector described in Section 3.1, and question and answer embeddings learned on the training set by the Convolutional Neural Network (CNN) described in (Severyn and Moschitti, 2015). More details about the used

[8] Also called SST.

[9] `https://github.com/SAG-KeLP`

[10] `http://disi.unitn.it/moschitti/Tree-Kernel.htm`

A	MAP	AvgRec	MRR	P	R	F$_1$	Acc
primary^2	77.66	88.05	84.93	75.56	58.84	66.16	75.54
cont$_1$	78.71	88.98	86.15	77.78	53.72	63.55	74.95
cont$_2$	77.29	87.77	85.03	74.74	59.67	66.36	75.41
best	79.19	88.82	86.42	76.96	55.30	64.36	75.11
baseline	59.53	72.60	67.83				
B							
primary^2	76.02	90.70	84.64	68.58	66.52	67.54	78.71
cont$_1$	75.57	89.64	83.57	63.77	72.53	67.87	77.14
best	76.70	90.31	83.02	63.53	69.53	66.39	76.57
baseline	74.75	88.30	83.79				
C							
primary^8	47.15	47.46	51.43	45.97	8.72	14.65	90.51
cont$_1$	43.31	44.19	48.89	30.00	3.21	5.80	90.26
cont$_2$	41.12	38.89	44.17	33.55	32.11	32.81	87.71
best	55.41	60.66	61.48	18.03	63.15	28.05	69.73
baseline	40.36	45.97	45.83				
D							
primary^2	45.50	50.13	52.55	28.55	64.53	39.58	62.10
cont$_1$	38.33	42.09	43.75	20.38	96.95	33.68	26.50
cont$_2$	39.98	43.68	46.41	26.26	68.39	37.95	57.00
best	45.83	51.01	53.66	34.45	52.33	41.55	71.67
baseline	28.88	28.71	30.93				

Table 1: Performance of our official primary and contrastive submissions to SemEval-2016 Task 3 for tasks A, B, C, and D. Best-performing and baseline systems included for comparison. The super-index in the primary submission stands for the position in the challenge ranking. The baselines are as provided by the task organizers; they are based on search engine rankings (except for task D, which is random).

embeddings and the resulting kernels can be seen in (Tymoshenko et al., 2016). The SVM was trained on the union of both training and development sets.

Task B. The submission consists in an SVM operating on three kernels: (*i*) an RBF kernel on the features described in Section 3.1, (*ii*) an RBF kernel on the features described in Section 3.2; and (*iii*) the tree kernel described in Section 3.3. The C parameter of the SVM was set to 1. Both the tree and the RBF kernels use default values for the parameters. The SVM was trained on the union of the training and development sets.

Task C. The submission consists in an SVM operating on two RBF kernels (with default parameter values): the first one is on the features described in Section 3.1. The second one is on the features described in Section 3.2 plus the score obtained from the prediction of a comment according to a classifier built for task A, computed by cross-validation. The SVM is trained on the union of the training part 2 and development sets.

Task D. The submission consists in an SVM operating on two kernels: (*i*) the syntactic tree kernel (SST) (Moschitti, 2006), applied as described in Section 3.3, to the constituency trees of the question texts; (*ii*) a linear kernel applied to the features in (Belinkov et al., 2015). In tasks A and B we used PTK, which is slower but more accurate. However, the trees of the Arabic data were rather large and very noisy. Thus we used SST, which is faster and uses less features. The value 0.1 for parameter L served the purpose of removing noise. The SVM was trained on the union of the training and development sets.

4.2 Contrastive Submissions

Task A. We submitted a contrastive run (cont$_1$), where we use a joint learning and inference approach based on a Fully-connected Conditional Random Field (FCCRF) (Joty et al., 2016) to classify all the comments in a thread collectively. We used the numeric (non-tree) features used previously in (Joty et al., 2015; Barrón-Cedeño et al., 2015), and also the predictions of the SVM used in our primary run. The FCCRF model uses an Ising-like edge potential, which distinguishes between only *same* and *different* (as opposed to all four possibilities) state transitions to model all pair dependencies.

The second contrastive run (cont$_2$) is as the primary submission, but without tree kernels.

Task B. We submitted one contrastive submission which is identical to the primary one, with the only exception that SVM is trained on the training part 2 and development sets only.

Task C. We submitted one contrastive submission which is identical to the primary one, with the exception that SVM is trained on all training and development sets.

The second contrastive submission consists of a rule-based system which relies on the outputs from tasks A and B. A comment is labeled as `good` if it is considered `good` with respect to the related question (Task A) and the related question is considered `relevant` with respect to the new question (Task B). The comment is considered `bad` otherwise.

Task D. The contrastive systems did not use tree kernels. Our first contrastive run used only feature vectors. Our second contrastive run also used additional features borrowed from machine translation evaluation: BLEU (Papineni et al., 2002), TER (Snover et al., 2006), Meteor (Lavie and Denkowski, 2009), NIST (Doddington, 2002), Precision and Recall, and length ratio between the question and the comment.

5 Results and Discussion

Table 1 shows the results obtained in the four tasks. We achieved the second position for tasks A, B, and D. In Task A, tree kernels give no major boost, but without them our model would be cont$_2$, which achieved the third position on the test set. The joint model cont$_1$, run on top of our primary system, was able to improve it by more than one point. We were not sure about the outcome of this model, thus we preferred not to use it as our primary submission, even though we got an improvement also on the development set.

Our cont$_1$ system for Task B, trained only on the train part 2 and development sets, scored less than our primary one. Even if our preliminary observations had suggested that the distributions of the different training and development sets were too different and potentially damaging the model learning, having more diverse data ended up as a better solution to the task.

Our submission for Task C is very limited as it does not include tree kernel models. The use of our feature vectors only (the same used for tasks A and B), results in an average performance in the challenge.

Regarding Task D, cont$_1$, using embedding features from (Belinkov et al., 2015), is an average system. When we add the machine translation evaluation (MTE) features the MAP increases from 38.33 to 39.98. We did not trust the MTE features as in the development set they obtained a lower result than the simple embedding features. This resulted to be a mistake from the competition viewpoint as they could have been combined with tree kernels. Indeed, our Primary system just combines tree kernels with the embedding features improving them by more than 7 absolute points, achieving the second position with a MAP of 45.50, very close to the best system, i.e., 45.83.

6 Conclusions

In this paper, we have presented the systems developed by the teams of the Qatar Computing Research Institute (QCRI) and the University of Trento for participating in SemEval-2016 Task 3 on Community Question Answering.

We used supervised machine learning approaches based on various combinations of the convolution tree kernels, text embedding features, including those learned by the convolutional neural networks, and a number of task-specific features from our previous work for SemEval-2015, Task 3.

For each task we submitted one primary and two contrastive runs incorporating various combinations of the above components. Our primary runs scored second for tasks A, B and D and eighth for task C. Finally, we analyzed the performance of our runs and discussed which components are more beneficial for a specific task/language.

In future work, we plan to devise better ways of combining convolution tree kernels with CNNs, e.g. by embedding the CNN similarities into the structural kernels, and encoding more complex relations into the structural representations of the text snippets.

Acknowledgments

This research is developed by the Arabic Language Technologies (ALT) group at the Qatar Computing Research Institute (QCRI), HBKU, Qatar Foundation in collaboration with MIT. It is part of the Interactive sYstems for Answer Search (IYAS) project. This work has been partially supported by the EC project CogNet, 671625 (H2020-ICT-2014-2, Research and Innovation action) and by an IBM Faculty Award.

References

Alberto Barrón-Cedeño, Simone Filice, Giovanni Da San Martino, Shafiq Joty, Lluís Màrquez, Preslav Nakov, and Alessandro Moschitti. 2015. Thread-level information for comment classification in community question answering. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 687–693, Beijing, China, July. Association for Computational Linguistics.

Yonatan Belinkov, Mitra Mohtarami, Scott Cyphers, and James Glass. 2015. Vectorslu: A continuous word vector approach to answer selection in community question answering systems. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 282–287, Denver, Colorado, June. Association for Computational Linguistics.

George Doddington. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the Second International Conference on Human Language Technology Research*, HLT '02, pages 138–145, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Richard Eckart de Castilho and Iryna Gurevych. 2014. A broad-coverage collection of portable nlp components for building shareable analysis pipelines. In *Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT*, pages 1–11, Dublin, Ireland, August. Association for Computational Linguistics and Dublin City University.

Simone Filice, Giuseppe Castellucci, Danilo Croce, Giovanni Da San Martino, Alessandro Moschitti, and Roberto Basili. 2015. KeLP: a Kernel-based Learning Platform in java. In *The workshop on Machine Learning Open Source Software (MLOSS): Open Ecosystems*, Lille, France, July. International Conference of Machine Learning.

Thorsten Joachims. 1999. Making Large-scale Support Vector Machine Learning Practical. In Bernhard Schölkopf, Christopher J. C. Burges, and Alexander J. Smola, editors, *Advances in Kernel Methods*, pages 169–184. MIT Press, Cambridge, MA, USA.

Shafiq Joty, Alberto Barrón-Cedeño, Giovanni Da San Martino, Simone Filice, Lluís Màrquez, Alessandro Moschitti, and Preslav Nakov. 2015. Global thread-level inference for comment classification in community question answering. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 573–578, Lisbon, Portugal, September. Association for Computational Linguistics.

Shafiq Joty, Llus Mrquez, and Preslav Nakov. 2016. Joint learning with global inference for comment classification in community question answering. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages xx–xx, San Diego, California, June. Association for Computational Linguistics.

Alon Lavie and Michael Denkowski. 2009. The METEOR metric for automatic evaluation of machine translation. *Machine Translation*, 23(2–3):105–115.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013. Linguistic Regularities in Continuous Space Word Representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL-HLT '13, pages 746–751, Atlanta, GA, USA.

Alessandro Moschitti, Silvia Quarteroni, Roberto Basili, and Suresh Manandhar. 2007. Exploiting syntactic and shallow semantic kernels for question answer classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 776–783, Prague, Czech Republic, June. Association for Computational Linguistics.

Alessandro Moschitti. 2006. Efficient Convolution Kernels for Dependency and Constituent Syntactic Trees. In Johannes Fürnkranz, Tobias Scheffer, and Myra Spiliopoulou, editors, *Machine Learning: ECML 2006*, volume 4212 of *Lecture Notes in Computer Science*, pages 318–329. Springer Berlin Heidelberg.

Alessandro Moschitti. 2008. Kernel Methods, Syntax and Semantics for Relational Text Categorization. In *Proceeding of ACM 17th Conf. on Information and Knowledge Management (CIKM'08)*, Napa Valley, CA, USA.

Preslav Nakov, Lluís Màrquez, Walid Magdy, Alessandro Moschitti, Jim Glass, and Bilal Randeree. 2015. Semeval-2015 task 3: Answer selection in community question answering. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 269–281, Denver, Colorado, June. Association for Computational Linguistics.

Preslav Nakov, Lluís Màrquez, Alessandro Moschitti, Walid Magdy, Hamdy Mubarak, Abed Alhakim Freihat, Jim Glass, and Bilal Randeree. 2016. SemEval-2016 task 3: Community question answering. In *Proceedings of the 10th International Workshop on Semantic Evaluation*, SemEval '16, San Diego, California, June. Association for Computational Linguistics.

Massimo Nicosia, Simone Filice, Alberto Barrón-Cedeño, Iman Saleh, Hamdy Mubarak, Wei Gao, Preslav Nakov, Giovanni Da San Martino, Alessandro Moschitti, Kareem Darwish, Lluís Màrquez, Shafiq Joty, and Walid Magdy. 2015. QCRI: Answer selection for community question answering - experiments for Arabic and English. In *Proceedings of the 9th International Workshop on Semantic Evaluation*, SemEval '15, Denver, Colorado, USA.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meting of the Association for Computational Linguistics*, ACL '02, pages 311–318, Philadelphia, Pennsylvania, USA.

Robert Parker, David Graff, Ke Chen, Junbo Kong, and Kazuaki Maeda. 2011. *Arabic Gigaword Fifth Edition*. Linguistic Data Consortium (LDC), Philadelphia.

Arfath Pasha, Mohamed Al-Badrashiny, Mona Diab, Ahmed El Kholy, Ramy Eskander, Nizar Habash, Manoj Pooleery, Owen Rambow, and Ryan Roth. 2014. Madamira: A fast, comprehensive tool for morphological analysis and disambiguation of arabic. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Slav Petrov and Dan Klein. 2007. Improved inference for unlexicalized parsing. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 404–411, Rochester, New York, April. Association for Computational Linguistics.

Aliaksei Severyn and Alessandro Moschitti. 2012. Structural relationships for large-scale learning of answer re-ranking. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '12, pages 741–750, New York, NY, USA. ACM.

Aliaksei Severyn and Alessandro Moschitti. 2013. Automatic feature engineering for answer selection and extraction. In *EMNLP*, pages 458–467. ACL.

Aliaksei Severyn and Alessandro Moschitti. 2015. Learning to rank short text pairs with convolutional deep neural networks. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 373–382. ACM.

Aliaksei Severyn, Massimo Nicosia, and Alessandro Moschitti. 2013a. Building structures from classifiers for passage reranking. In *Proceedings of the 22nd ACM international conference on Conference on information &#38; knowledge management*, CIKM '13, pages 969–978, New York, NY, USA. ACM.

Aliaksei Severyn, Massimo Nicosia, and Alessandro Moschitti. 2013b. Learning adaptable patterns for passage reranking. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, CoNLL '13, pages 75–83, Sofia, Bulgaria.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Biennial Conference of the Association for Machine Translation in the Americas*, AMTA '06, Cambridge, Massachusetts, USA.

Kateryna Tymoshenko and Alessandro Moschitti. 2015. Assessing the impact of syntactic and semantic structures for answer passages reranking. In *Proceedings of the 24th ACM International on Conference on In-*

formation and Knowledge Management, CIKM ’15, pages 1451–1460, New York, NY, USA. ACM.

Kateryna Tymoshenko, Alessandro Moschitti, and Aliaksei Severyn. 2014. Encoding semantic resources in syntactic structures for passage reranking. In *14th Conference of the European Chapter of the Association for Computational Linguistics 2014, EACL 2014*, pages 664–672, Gothenburg, Sweden, April. Association for Computational Linguistics (ACL).

Kateryna Tymoshenko, Daniele Bonadiman, and Alessandro Moschitti. 2016. Convolutional neural networks vs. convolution kernels: Feature engineering for answer sentence reranking. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL ’16, San Diego, CA, USA, June. Association for Computational Linguistics.