SUper Team at SemEval-2016 Task 3: Building a Feature-Rich System for Community Question Answering

Tsvetomila Mihaylova, Pepa Gencheva, Martin Boyanov, Ivana Yovcheva, Todor Mihaylov, Momchil Hardalov, Yasen Kiprov, Daniel Balchev, Ivan Koychev
FMI, Sofia University "St. Kliment Ohridski", Sofia, Bulgaria

Preslav Nakov
ALT Research Group, Qatar Computing Research Institute, HBKU, Doha, Qatar

Ivelina Nikolova, Galia Angelova
IICT, Bulgarian Academy of Sciences, Sofia, Bulgaria (iva@lml.bas.bg)

Abstract

We present the system we built for participating in SemEval-2016 Task 3 on Community Question Answering. We achieved the best results on subtask C, and strong results on subtasks A and B, by combining a rich set of various types of features: semantic, lexical, metadata, and user-related. The most important group turned out to be the metadata for the question and for the comment, semantic vectors trained on QatarLiving data and similarities between the question and the comment for subtasks A and C, and between the original and the related question for Subtask B.

1 Introduction

SemEval-2016 Task 3 on Community Question Answering[1] (Nakov et al., 2016) aims to solve a real-life application problem. The main **subtask C (Question-External Comment Similarity)** asks to find an answer in the forum that will be appropriate as a response to a newly posted question. This is achieved by retrieving similar questions and ranking their answers with respect to the new question. Two additional supporting subtasks are defined:

Subtask A (Question-Comment Similarity): Given a question from a question-comment thread, rank the comments within the thread based on their relevance with respect to the question.

Subtask B (Question-Question Similarity): Given a new question, re-rank the similar questions retrieved by a search engine with respect to that question.

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

2 Related Work

We build our preprocessing and feature extraction pipeline based on the system of Zamanov et al. (2015), which was developed by a subset of our 2016 team for SemEval-2015 Task 3 on Answer Selection in Community Question Answering (Nakov et al., 2015). The task in 2015 was to classify comments in a thread as *relevant*, *potentially useful*, or *bad* with respect to the thread question. This year's Community Question Answering subtask A is similar to subtask A in 2015, but now it is a ranking task, asking to rank the answers in a thread based on their relevance with respect to the thread's question. Given this similarity, most of the techniques used by participants in the 2015 subtask A are potentially valuable for this year's subtask A as well. Below we mention just the few most relevant among them.

In their 2015 system, Belinkov et al. (2015) used vectors of the question and of the comment, metadata features, and text-based similarities. Nicosia et al. (2015) used similarity measures, URLs in the comment text and statistics about the user profile: number of good, bad, and potentially useful comments. Similarly, we use the number of posts by the same user in the thread, the ID of the question's author, topic model-based feature, special words, etc.

Determing the overall sentiment of the question can also be useful, and it was used in 2015 (Nicosia et al., 2015). One way to do it is to build a sufficiently large question taxonomy as described in (Li and Roth, 2006). This may help determine the quality of the answer, but it requires significant efforts in order to build such a taxonomy.

Proceedings of SemEval-2016, pages 761–768,
San Diego, California, June 16-17, 2016.

3 Data

For training and testing, we used data provided by the SemEval-2016 Task 3 organizers. The datasets consist of 6,398 questions and 40,288 comments for Subtask A, 317 original + 3,169 related questions for Subtask B, and 317 original questions + 3,169 related questions + 31,690 comments for Subtask C.

For subtask A, the comments in a question-answer thread are annotated as *Good*, *PotentiallyUseful* and *Bad*. A good ranking is one that ranks all *Good* comments above *PotentiallyUseful* and *Bad* ones (without distinguishing between the latter two).

For subtask B, the potentially relevant questions are annotated as *PerfectMatch*, *Relevant* and *Irrelevant* with respect to the original question. A good ranking is one where the *PerfectMatch* and the *Relevant* questions (without distinguishing between them) are both ranked above the *Irrelevant* ones.

We also used semantic vectors (Mikolov et al., 2013a) pretrained on Google News data: 300-dimensional vectors, available for three million words and phrases.

For all subtasks, we further trained semantic vectors using Gensim (Řehůřek and Sojka, 2010) on 200,000 questions and two million comments from the Qatar Living Forum,[2], which were provided by the task organizers.

Finally, using this same data, and following (Mihaylov et al., 2015a; Mihaylov et al., 2015b), we scraped information about the users from the forum and we extracted for each of them the time in the forum, the active period, the number of questions, the comments in the forum, etc.

4 Method

We build our system on top of the framework developed by our colleagues (Zamanov et al., 2015). In particular, we approach the task as a classification problem similarly to the approach we took for SemEval 2015 Task 3 (Nakov et al., 2015). However, unlike 2015, this year we have a ranking problem for all subtasks, e.g., for subtask A we have to rank the comments depending on how likely the classifier thinks they are to be *Good* vs. them being *Bad* or *PotentiallyUseful*.

[2]Qatar Living: http://www.qatarliving.com/forum

We use variety of features like question and comment metadata; question and comment lexical features; distance measures between the question and the comment; text readability measures applied to the question and to the comment; lexical semantics vectors for the question and for the comment; features modeling the likelihood of a user being a troll.

These features proved quite useful for ranking comments with respect to a given question (Subtask A and C), but they did not achieve as high results when ranking questions with respect to other questions (Subtask B).

4.1 Features

Metadata Features

These features are based on surface observations of the thread's structure and properties. From the comments' attributes we extract whether the comment is written by the author of the question. We further extract the comment's position in the thread, and the ID of the author of the comment. Next, we tokenize the text and we calculate the ratio of the comment length and of the question length (in terms of number of tokens). In terms of the threads we measure, the number of comments from the same user in a particular thread and the order in which they are written by the user, i.e., first, second, etc. comment by the same user. In terms of the whole QatarLiving forum, we calculate the number of questions in a category.

Another family of metadata features explores the presence and the number of links in the question and in the comment. We counted both inbound (i.e., to `qatarliving.com`) and outbound links. Our hypothesis was that the presence of a reference to another resource is indicative of a relevant comment. Investigations on the training set showed that a relevant comment was more likely to contain such a link. Unfortunately, less than 10% of the comments had links, and ultimately these features did not have a very high impact on the results.

Lexical Features

These features represent the lexical content of questions and comments. They are obtained with the help of the GATE (Cunningham et al., 2011; Cunningham et al., 2002) preprocessing pipeline with some hand-crafted rules and various statistics.

We use token-, NP-, and sentence-based features as well as features based on the following entities: Person, Location, Organisation and Address. The latter ones are used to mark whether the comment contains an answer to a **wh**-question (**wh**ere, **wh**o, **wh**at, etc.), e.g., if the question contains the word "where". We further add a boolean feature modeling whether the comment contains a Location or an Address. We tagged the named entities using the high-quality named entity recognition pipeline of Ontotext.[3] We further extracted statistics about the number of verbs, nouns, pronouns, and adjectives in the question and in the comment, as well as the number of question marks in the comments, and the number of question words in the question and in the comment.

Another group of lexical features are extracted from the comment text only and show whether it contains smileys, currency units, e-mails, phone numbers, only laughter, "thank you" phrases, personal opinions, or disagreement.

Other lexical features relate to spelling and include number of misspelled words that are within edit distance of 1 from a word in our vocabulary and number of offensive words from a predefined list.

We also borrow a dictionary from the PMI-cool system (Balchev et al., 2016), which is based on unigram and bigram occurrences across the classes. We use it to compute the *Pointwise Mutual Information* (PMI) between a dictionary entry and a class. Based on it, we add features that sum the PMI for all tokens in a given comment. This family of features are weighed most heavily by the classifier.

We further computed lexical similarity between a question and a comment using *SimHash* (Sadowski and Levin, 2007), which is a near-duplicate similarity measure but it did not help much.

We also apply a set of statistical scores to measure the **level of readability** and complexity of the text (Aluisio et al., 2010). The standard readability measures include Automated Readability Index, Coleman-Liau Index, Flesch Reading Ease, Gunning Fog Index, Flesch-Kincaid Grade Level, LIX, SMOG grade. We also employ statistics about the average number of words per sentence in the comment or question, and type-to-token ratios.

[3] http://ontotext.com/

Semantic Features

Our semantic features try to capture the proximity between the meanings encoded in the word sequences of the questions and of the answers.

One of the semantic features uses **Mallet topic modelling** (McCallum, 2002). We build a topic model with 100 topics. Then we measure the cosine distance between the topics in the first text and the topics in the second text, i.e., in the question and in the answer (Subtasks A and C), or in the original and in the related question (Subtask B).

We also used **semantic vectors** trained with word2vec (Mikolov et al., 2013b). We performed experiments to select the best vectors for the task. We tried pre-trained vectors from Google News. We further trained vectors from the unannotated data from the QatarLiving forum. We used the latter vectors in our system as they yielded better results.

We experimented with training vectors of different sizes and different minimal word counts. Because of the many common misspelled words, the smaller word count yields better results. We tried different tokenizations for the words. To capture the specifics of the forum language, we added identifiers for numbers, smileys, URLs and images. For each question-comment pair, we calculated the centroid vectors of the question and of the comment and we used the components of the resulting vectors as features for the classifier. We used Gensim (Řehůřek and Sojka, 2010) for building the vectors.

User Features

We downloaded and used characteristics about the users from the QatarLiving forum, such as number of questions, comments, classifieds; time since registration, time since last activity in the forum, time of the day in which the user was active, etc. We also added as user characteristics the number of good and bad comments from the annotated training data. However, the user features did not improve the results. We noticed that over time, the number of both good and bad comments for a user in the forum grew, and the number of good and bad comments for most of the users was similar.

We also used troll user features, e.g., number of mentions of the user as troll and troll behavior characteristics as described in (Mihaylov et al., 2015a; Mihaylov et al., 2015b; Mihaylov and Nakov, 2016a).

Features	Dev-2016 as test set MAP	Dev-2016 as test set Accuracy	Test-2016 as test set MAP	Test-2016 as test set Accuracy
All	69.89	76.60	77.83	74.43
All − semantic vectors	65.93	73.11	74.61	70.76
All − metadata	65.51	74.96	74.30	72.91
All − comment characteristics	69.30	75.49	77.38	73.30
All − distances	68.22	76.19	76.90	73.70
All − URLs	69.96	76.27	77.84	74.04
All − User stats	70.08	76.48	78.34	74.31
All − Wh-words in Q and C	69.55	76.56	77.72	74.80
All − Wh-words in Q	69.73	76.97	77.66	74.40
All − Wh-words in C	69.98	76.48	77.88	74.28
All − Loc/Org in Comment	69.95	76.56	77.82	74.28
All − POS count in Q	69.85	76.07	77.36	74.50
All − POS count in C	69.61	76.02	77.62	74.22
All − POS and Wh-words in Q	70.02	76.43	77.81	74.59
Primary	70.67	77.62	77.16	74.50
Contrastive-1	70.06	76.84	77.68	74.50
Contrastive-2	—	—	76.97	74.34

Table 1: Subtask A: Experiments with all features and excluding some feature groups.

Credibility Features

We further added some of the credibility features described in (Castillo et al., 2013). We trained a prediction model on tweets from the dataset described in that paper. We used **linear Support Vector Machines** (linear SVMs) with Stochastic Gradient Descent (SGD) and L2 regularization. We used an off-the-shelf implementation of SVM, provided in the **Apache Spark** library (Apache Software Foundation Team, 2015).

For each answer we extracted the following features: length of the answer (characters); does the answer contain special punctuation, like question marks, exclamation marks, etc.; is there an emoticon in the text; is there a first person singular (*I, me, my, mine, we*) or plural pronoun (*our, ours, we, us*); is there a third person singular (*he, she, it, his, hers, him, her*) or plural pronoun (*they, them, their*); does the answer contain a user mention (@user); does the text contain URLs in it.

Based on these features we trained an SVM model to classify items as credible or not. For our submission, we used all the features used to train the credibility module as well as the predicted label and the probability it is predicted with.

4.2 Classifier

Using the above features, we formed vectors associated with each question-answer pair. Those vectors are a concatenation of the extracted features, including the centroid of the semantic vectors for the question and for the comment.

We then used an SVM classifier as implemented in LibSVM (Chang and Lin, 2011) for classification. We experimented with different kernels (Hsu et al., 2003), and we achieved the best results with the RBF kernel, which we use to train the model for our submissions. The ranking score for a question-comment pair in Subtask A is the calculated probability of the pair to be classified as *Good*.

For Subtask C, we used the same approach as in Subtask A. We first ranked the comments with respect to the relevant question. For the final ranking, we multiplied the probability of the pair "relevant question – comment" being *Good* by the reciprocal rank of the related question as given by Google.

For subtask B, we passed to the classifier characteristics of the pair "original question – relevant question". For ranking, we used the probability of the pair to be classified as *Good*.

Features	Dev-2016 as test set MAP	Dev-2016 as test set Accuracy	Test-2016 as test set MAP	Test-2016 as test set Accuracy
All	41.46	69.32	55.62	70.21
All − semantic vectors	35.57	71.52	52.51	71.04
All − metadata	39.90	69.08	54.58	71.10
All − comment characteristics	40.57	68.92	56.20	70.50
All − distances	40.96	69.66	52.97	70.64
All − URLs	40.31	69.44	56.20	70.57
All − User stats	41.30	69.22	55.57	70.07
All − Wh-words in Q and C	39.20	68.76	53.58	70.40
All − Wh-words in Q	40.19	69.12	54.61	70.50
All − Wh-words in C	39.83	69.16	55.01	70.69
All − Loc/Org in Comment	40.14	69.24	55.70	70.34
All − POS count in Q	40.62	69.12	54.70	70.47
All − POS count in C	40.09	69.26	56.47	70.31
All − POS and Wh-words in Q	41.57	69.14	54.62	70.44
Primary	42.42	68.46	55.41	69.73
Contrastive-1	42.54	81.38	48.23	82.49
Contrastive-2	42.56	68.64	53.48	69.20

Table 2: Subtask C. Experiments with all features and excluding some feature groups.

Features	Dev-2016 as test set MAP	Dev-2016 as test set Accuracy	Test-2016 as test set MAP	Test-2016 as test set Accuracy
Only semantic vectors	71.17	67.20	74.91	68.71
Semantic vectors + cosine distance	71.76	69.00	74.43	72.43
Above + topic distance	72.34	70.80	75.22	74.43
Above + metadata	72.84	70.20	74.82	74.86
Above + text distance	72.39	72.00	75.17	75.57
All − semantic vectors	71.98	71.20	74.43	77.14

Table 3: Subtask B. Experiments with the different feature sets for the related and the original question.

5 Experiments and Evaluation

We grouped our features in several groups and we ran experiments by excluding some of them in order to identify the most important types of features. In particular, we used the LibSVM `fselect` script for feature selection. We achieved the best results by combining the features with the semantic vectors of the question and of the comment trained on Qatar-Living data.

In Table 1, we present the results for Semeval-2016 Task 3, Subtask A using all features, as well as when excluding individual feature groups. Our primary submission includes the top-rated features and semantic vectors. We selected our primary submission as it achieved higher score on Dev than Contrastive1 and Contrastive2.

Compared to Contrastive-1, our Primary has some additional features: number of user comments in the thread, cosine between the comment text with question subject and category, more locations and organizations. Our Contrastive-1 submission included the top-rated features and semantic vectors, and our Contrastive-2 submission included the same features as our Primary submission, but used Dev-2016 as additional training set.

In Table 2, we show the results for Semeval-2016 Task 3, Subtask C using all features, as well as when excluding individual feature groups. Our Primary submission includes all features excluding user statistics and troll features. Our Contrastive-1 submission includes all features, including PMI, while Contrastive-2 includes all features, excluding PMI.

Features	Dev-2016 as test MAP	Accuracy
Vectors from Google News		
Nouns	54.95	68.52
Nouns + verbs	55.21	69.06
Nouns + verbs + adjectives	54.97	68.48
Vectors from QatarLiving		
Nouns + verbs	61.48	70.82
Nouns + verbs + adjectives	60.43	70.37
MWC=40; words only	58.95	71.27
MWC=40; + special symbols	59.65	71.27
All words; MWC=5	62.68	71.80
Included cosine distance	63.88	72.99

Table 4: Selection of semantic vectors. Experiments with different sources, vector size, and minimum word count.

Tables 1 and 2 have shown that the most important feature groups are the metadata characteristics, the distance measures between the question and the comment, and the semantic vectors. Other features that `fselect` scored highly are the credibility score, text readability measures and the number of tokens of some parts of speech in the comment text (namely, number of adjectives and nouns). The least useful features are statistics about the forum users and characteristics of the question: question length and number of tokens of different parts of speech in the question text. In all above reported results, we used vectors for which we achieved the best results on the development dataset.

In Table 4, we present the results from experiments with semantic vectors. We experimented with pre-trained vectors from Google News and we also trained vectors with word2vec on the unannotated Qatar Living forum data. When training vectors on Qatar Living data, we experimented with different vector sizes and minimum word frequencies. We also added the following entities as words: numbers, images, URLs, smileys (referred to in the table as "special symbols"). We achieved best results with vectors from QatarLiving, vector size 100, and minimum word frequency of 5. Including special symbols as words also improved the results. We experimented with calculating the centroids of the question and of the comment using specific parts of speech only; however, ultimately we found that using all words from all parts of speech worked best.

For **Subtask B**, we used a similar approach as for Subtask A: we passed to the classifier the semantic vectors of the original and of the related question, some metadata and distance features. However, we could not experiment much with this subtask, and thus our results are not as strong, as Table 3 shows.

6 Conclusion

We have presented the system developed by our team for participating in SemEval-2016 Task 3 on Community Question Answering. We achieved the best results on subtask C, and strong results on subtasks A and B, by combining a rich set of various types of features: semantic, lexical, metadata, and user-related. The most important group turned out to be the metadata for the question and for the comment, semantic vectors trained on QatarLiving data and similarities between the question and the comment for subtasks A and C, and between the original and the related question for Subtask B.

In future work, we would like to experiment with new, interesting features, e.g., based on various word embeddings as in the SemanticZ system (Mihaylov and Nakov, 2016b). We also want to use our features in a deep learning architecture, e.g., as in the MTE-NN system (Guzmán et al., 2016a; Guzmán et al., 2016b), which borrowed an entire neural network framework and architecture from previous work on machine translation evaluation (Guzmán et al., 2015).

We further plan to use information from entire threads to make better predictions, as using thread-level information for answer classification has already been shown useful for SemEval-2015 Task 3, subtask A, e.g., by using features modeling the thread structure and dialogue (Nicosia et al., 2015; Barrón-Cedeño et al., 2015), or by applying thread-level inference using the predictions of local classifiers (Joty et al., 2015; Joty et al., 2016). How to use such models efficiently in the ranking setup of 2016 is an interesting research question.

Finally, we would like to address subtask C in a more solid way, making good use of the data, the gold annotations, the features, the models, and the predictions for subtasks A and B.

Acknowledgments

This research was performed by a team of students from MSc programs in Computer Science in the Sofia University "St Kliment Ohridski".

It is also part of the Interactive sYstems for Answer Search (Iyas) project, which is developed by the Arabic Language Technologies group at the Qatar Computing Research Institute, HBKU, part of Qatar Foundation in collaboration with MIT-CSAIL.

We would also like to thank Ontotext for providing us with their high-quality NER pipeline.

References

Sandra Aluisio, Lucia Specia, Caroline Gasperin, and Carolina Scarton. 2010. Readability assessment for text simplification. In *Proceedings of the NAACL HLT 2010 Fifth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 1–9, Los Angeles, California, USA.

Apache Software Foundation Team. 2015. Spark programming guide.

Daniel Balchev, Yasen Kiprov, Ivan Koychev, and Preslav Nakov. 2016. PMI-cool at SemEval-2016 Task 3: Experiments with PMI and goodness polarity lexicons for community question answering. In *Proceedings of the 10th International Workshop on Semantic Evaluation*, SemEval '16, San Diego, California, USA.

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*, ACL-IJCNLP '15, pages 687–693, Beijing, China.

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 '15, pages 282–287, Denver, Colorado, USA.

Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. 2013. Predicting information credibility in time-sensitive social media. *Internet Research*, 23(5):560–588.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27.

Hamish Cunningham, Diana Maynard, Kalina Bontcheva, and Valentin Tablan. 2002. GATE: an architecture for development of robust HLT applications. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, ACL '12, pages 168–175, Philadelphia, Pennsylvania, USA.

Hamish Cunningham, Diana Maynard, Kalina Bontcheva, Valentin Tablan, Niraj Aswani, Ian Roberts, Genevieve Gorrell, Adam Funk, Angus Roberts, Danica Damljanovic, Thomas Heitz, Mark A. Greenwood, Horacio Saggion, Johann Petrak, Yaoyong Li, and Wim Peters. 2011. *Text Processing with GATE (Version 6)*.

Francisco Guzmán, Shafiq Joty, Lluís Màrquez, and Preslav Nakov. 2015. Pairwise neural machine translation evaluation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, ACL-IJCNLP '15, pages 805–814, Beijing, China.

Francisco Guzmán, Lluís Màrquez, and Preslav Nakov. 2016a. Machine translation evaluation meets community question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, ACL '16, Berlin, Germany.

Francisco Guzmán, Lluís Màrquez, and Preslav Nakov. 2016b. MTE-NN at SemEval-2016 Task 3: Can machine translation evaluation help community question answering? In *Proceedings of the 10th International Workshop on Semantic Evaluation*, SemEval '16, San Diego, California, USA.

Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin. 2003. A practical guide to support vector classification. Technical report, Department of Computer Science, National Taiwan University.

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*, EMNLP '15, pages 573–578, Lisbon, Portugal.

Shafiq Joty, Lluís Màrquez, 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*, NAACL-HLT '16, San Diego, California, USA.

Xin Li and Dan Roth. 2006. Learning question classifiers: The role of semantic information. *Nat. Lang. Eng.*, 12(3):229–249, September.

Andrew Kachites McCallum. 2002. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu.

Todor Mihaylov and Preslav Nakov. 2016a. Hunting for troll comments in news community forums. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, ACL '16, Berlin, Germany.

Todor Mihaylov and Preslav Nakov. 2016b. SemanticZ at SemEval-2016 Task 3: Ranking relevant answers in community question answering using semantic similarity based on fine-tuned word embeddings. In *Proceedings of the 10th International Workshop on Semantic Evaluation*, SemEval '16, San Diego, California, USA.

Todor Mihaylov, Georgi Georgiev, and Preslav Nakov. 2015a. Finding opinion manipulation trolls in news community forums. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, CoNLL '15, pages 310–314, Beijing, China.

Todor Mihaylov, Ivan Koychev, Georgi Georgiev, and Preslav Nakov. 2015b. Exposing paid opinion manipulation trolls. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, RANLP '15, pages 443–450, Hissar, Bulgaria.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013a. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, NIPS '13, pages 3111–3119. Curran Associates, Inc.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013b. 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, Georgia, 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 '15, pages 269–281, Denver, Colorado, USA.

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, USA.

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, pages 203–209, Denver, Colorado, USA.

Radim Řehůřek and Petr Sojka. 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.

Caitlin Sadowski and Greg Levin. 2007. Simihash: Hash-based similarity detection. Technical Report UCSC-SOE-11-07, University of California, Santa Cruz, USA.

Ivan Zamanov, Marina Kraeva, Nelly Hateva, Ivana Yovcheva, Ivelina Nikolova, and Galia Angelova. 2015. Voltron: A hybrid system for answer validation based on lexical and distance features. In *Proceedings of the 9th International Workshop on Semantic Evaluation*, SemEval '15, pages 242–246, Denver, Colorado, USA.