RDI_Team at SemEval-2016 Task 3: RDI Unsupervised Framework for Text Ranking

Ahmed Magooda[1,§], Amr M. Sayed[2,§], Ashraf Y. Mahgoub[1, §], Hany Ahmed[1, §], Mohsen Rashwan[1], Hazem Raafat[3], Eslam Kamal[4], Ahmad A. Al Sallab[5]

[1] RDI, Cairo, Egypt.

[2]Computer Science Department, Faculty of Computers and Information, Cairo University, Egypt.

[3]Computer Science Department, Kuwait University, Kuwait

[4]Microsoft Research, Cairo, Egypt.

[5]Valeo Inter-branch Automotive Software, Cairo, Egypt.

§ These four authors contributed equally to this work.

ahmed.ezzat.gawad@gmail.com, amr@fci-cu.edu.eg, ashraf.youssef.mahgoub@gmail.com, hanyahmed@aucegypt.edu, mrashwan@rdi-eg.com, hazem@cs.ku.edu.kw, eskam@microsoft.com, ahmad.el-sallab@valeo.com

Abstract

Ranking is an important task in the field of information retrieval. Ranking may be used in different modules in natural language processing such as search engines. In this paper, we introduce a competitive ranking system which combines three different modules. The system participated in SemEval 2016 question ranking task for the Arabic language. The task is a ranking task that targets reordering results retrieved from search engine. Results reordering is done based on relevancy between search result and the original query issued. The data provided in the competition is in the form of question (query) and 30 question answer pairs retrieved from search engine. For each question retrieved from the search engine the system generates a relevancy score that is to be used for ranking. The proposed system came in the third position in the Competition. Since the majority of modules are unsupervised the unsupervised naming was used.

1 Introduction

This paper describes RDI System, the system participated in SemEval 2016 Community question ranking shared task for the Arabic language (Nakov et al., 2016). The task is a ranking task for medical community questions, whenever a new user wants to submit a new question, the system should retrieve similar questions that are previously answered. Questions retrieved by the search engine are then to be ordered by its relevancy to query question in order to reduce redundancy.

The task proceeds as follows, for each query question 30 previously answered question answer pairs are retrieved through a search engine. The aim of the task is to reorder the retrieved 30 question answer pairs based on its relevancy to the query question.

As query expansion and results re-ranking are two major paradigms for search engine enhancements, (Mahgoub et al., 2014) used Google synonyms and Arabic Wikipedia to expand search queries, on the other hand (Abouenour et al., 2010) proposed a system that uses Arabic WordNet to expand queries. In this paper we are going to propose a

Proceedings of SemEval-2016, pages 739–744,
San Diego, California, June 16-17, 2016.

ranking system which consists of three modules merged with each other to produce a relevancy score. The proposed system managed to achieve the third position in the SemEval 2016 Community question ranking shared task for the Arabic. The 3 modules presented in this system are:

1- TF-IDF based module
2- Language model (LM) based module
3- Wikipedia-based module

The paper is organized as follows: Section 2 introduces some related work. Section 3 introduces data used. Section 4 contains proposed system. Section 5 contains Results obtained. In Section 6, some conclusions and perspective are discussed. Section 7 contains future work.

2 Related Work

The community question ranking problem is different than ordinary question answering system that aims to generate a satisfactory answer for a specific question. Community question ranking problem aims to find the most suitable answer for a question within a closed set of question answers pool.

(Zongcheng et al. 2012) argued that a question and answer can be considered the same topic distribution written in two different languages or written by two different writers. So they proposed extracting the latent topics from question and answer, they used the extracted topics to represent the question and answer, alongside measuring how much question and answer share topics.

(Jeon et al. 2006) used some non-textual features to cover the contextual information of questions and answers, and proposed using language modeling to process these features in order to predict the quality of answers collected from a specific CQA service. (Blooma et al. 2008) used regression to combine textual and non-textual features to generate predictive score to identify the best answer.

(Ko et al. 2007) proposed using probabilistic answer ranking model to calculate the answer relevance and answer similarity. They used logistic regression to calculate the correctness score for an answer using its relevancy to the question. The authors then improved their work by using probabilistic graphical model so that the correlation between all the answers can be taken into consideration.

3 Data

During the development of the proposed system 4 sets of data were used.

A) Training data provided by SemEval
This data consists of 1030 search queries, each query is accompanied with 30 search engine retrieved results. Data is manually annotated with relevancy score and is marked as relevant or irrelevant. Where relevant annotated questions are questions that are actually relevant to the query question, and irrelevant annotated questions are irrelevant to the query question.

B) Development data provided by SemEval
This data consists of 250 search queries, each query is accompanied with 30 search engine retrieved results. Data is manually annotated as relevant or irrelevant. Where relevant annotated questions are questions that are actually relevant to the query question, and irrelevant annotated questions are irrelevant to the query question.

C) Test data provided by SemEval
This data consists of 250 search queries, each query is accompanied with 30 search engine retrieved results. In this case, data has no annotation provided.

D) Crawled data
This data is in the form of question-answer pairs crawled from http://www.altibbi.com. 170,000 question answer pairs were crawled, all of these data samples were considered relevant samples as we considered relevancy between answer text and question text.

4 Proposed System

This section describes the proposed system in details. As shown in figure 1, three different modules are proposed. First, relevancy between search results retrieved from the search engine and the search query are calculated using three modules. The three relevancy values calculated are then converted into one relevancy score using weighted summation.

Retrieved documents are then re-ordered based on the new weighted sum scores. The best weights used for weighting the three modules (α, β and γ) were inferred through tuning over development set.

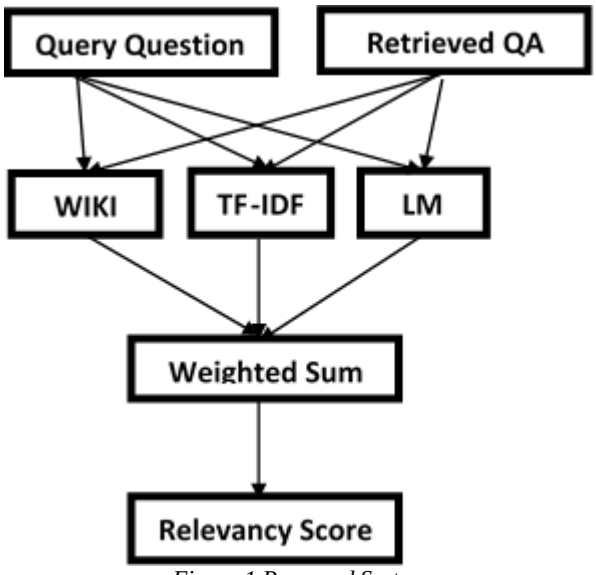

Figure 1.Proposed System

4.1 TF-IDF Module

This module uses the crawled data as background corpora to calculate the relevancy between the query question and each of the 30 retrieved question answer pairs.

The main idea of the algorithm is that for each query question, cluster the background data into relevant and irrelevant classes. The two constructed classes are then used to calculate how much the retrieved 30 question answer pairs are relevant to the query question. The algorithm flows as follows:

- Crawled data and query question are normalized by removing (non-Arabic words, Numbers, Punctuations and Stop words).
- For the query question, cluster the crawled data into two classes; 1- Relevant and 2- Irrelevant. This can be done by counting the number of common words between the query question and each question answer pair of the crawled data. If the number of common words is more than or equal 3, then this pair is considered relevant, otherwise is considered irrelevant.
- For each pair of the retrieved question answer pairs, unigrams, bigrams and trigrams are extracted. For each unigram, bigram and trigram TF-IDF is calculated twice, once using IDF extracted from the relevant class question answer pairs and once using the IDF extracted from the irrelevant class question answer pairs.
- The TF-IDF for the question pair is calculated by weighting the unigram TF-IDF by ω_1, bigrams by ω_2 and trigrams by ω_3.
- The TF-IDF with relevant class and TF-IDF with irrelevant class are normalized into probability values.
- The relevant probability is then used as the final score for the current question answer pair.

4.2 Language Model Module

This module utilizes the concept of language modeling into the ranking task. A language model was trained using the crawled data alongside the training data provided for the competition that was annotated as relevant question answer pair. In our experiments, we used the provided training data to reform new training samples as shown in figure.2.

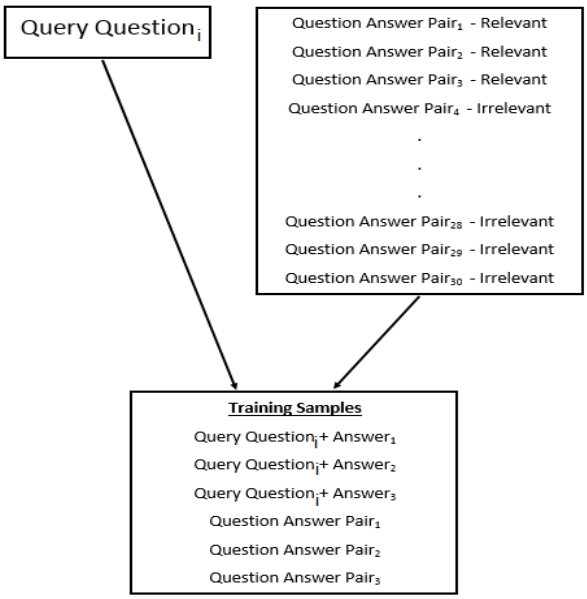

Figure 2. Generating LM training samples

As shown in figure.2, for each query question and its 30 retrieved question answer pairs, to form new text samples, query question text has been appended to answer text of pairs which are annotated as relevant. The main idea behind using language model is that, a sentence formed from question and its answer is coherent, so using the answer with the query question in case the answer is annotated as relevant; will lead to coherent sentence too.

By using the previous algorithm, we generated new training samples which represent the relevancy between questions and answers. Those training

samples are essential to feed the language model with more coherent training samples.

Using the previous generated samples a recurrent neural network for language modeling (Mikolov et al., 2011) with the following characteristics was trained;

- 1-hidden layer of size 200 neuron and sigmoid activation function.
- Hierarchical Softmax output layer over the whole vocabulary.

This module is the only supervised module used within the system, since human annotation was employed within this module.

4.3 Wikipedia-based module

This module uses medical information provided by Arabic-Wikipedia in order to provide better scoring methodology. The intuition behind using Wikipedia's medical terms is to provide higher matching score for those terms present in Wikipedia's medical tree over other matched terms.

Using the categorization system provided by Wikipedia, about 250,000 medical subcategories where extracted from the Medical category, the list is available for download from the author's website[1]. These extracted terms are then separated into three categories: unigrams, bigrams, or more, where each category weighted by a different matching factor.

To assign a total matching score between a query question and a question answer pair, the following steps are applied for each of the 30 retrieved question answer pairs:

- Top 1500 Most frequent words generated by (Zahran et al., 2015) were removed from both query question and the questions answer pairs.
- Find matching terms between the question and the question answer pair.
- For each matching term:
    - If the matching term is a unigram which exists in Wikipedia extracted terms, then increment the total matching score by *Uni_Wiki_Factor*
    - Else if the matching term is a unigram which doesn't exists in Wikipedia extracted terms, then increment the total matching score by *Uni_Factor*.
    - Repeat the previous two steps for both: bigrams and higher n-grams using (*Bi_Wiki_Factor, Bi_Factor*) for bigrams and (*N_Wiki_Factor, N_Factor*) for higher order n-grams.

Moreover, values of these matching factors are tuned over a subset of the provided training set and then validated on the development set. The optimum values of the matching factors are

- Uni_Wiki_Factor = 1.5
- Uni_Factor = 1
- Bi_Wiki_Factor = 2
- Bi_Factor = 1
- N_Wiki_Factor = 2
- N_Factor = 1

5 Results

The proposed system achieved the 3rd position in the Arabic subtask of SemEval 2016 task3.

Follows a detailed analysis of the system performance on development data set, alongside the performance of each module on the same data:

- **TF-IDF Module:**

Data	System	MAP	AvgRec	MRR
Dev	**TF-IDF Module**	**40.6**	**0.42**	**47**

Table 1.Performance of TFIDF module

- **Language Model Module:**

Data	System	MAP	AvgRec	MRR
Dev	**Language Model**	**35.8**	**0.374**	**40**

Table 2.Performance of Language Model module

- **Wikipedia Module:**

Data	System	MAP	AvgRec	MRR
Dev	**Wikipedia Module**	**42.6**	**0.46**	**48.3**

Table 3.Performance of Wikipedia module

The best parameters used to combine the three modules were inferred from tuning the system over the development data set, the best weights are

α (TF-IDF) = 0.3

β (Language Model) = 0.05

[1] https://drive.google.com/file/d/0B0FhtCSJsoyoeT-Jkd1FUQksxRmM/view

γ (Wikipedia) = 0.75

Table 4 shows the performance of the combined system on the development data set:

Data	System	MAP	AvgRec	MRR
Dev	**Combined System**	**44.8**	**0.48**	**51.1**

Table 4.Performance of combined system on development data

Simple as it may seem, the Wikipedia module achieved the best results on development data set compared to the other two modules. This behavior is due to the ability of the Wikipedia module to highlight medical terms and assign it higher weights compared to other matched terms, on the other hand the other two modules neglect this property, as both modules focus on detecting similarity between sentences regardless of the domain.

The proposed system also achieved 43.80 Mean Average Precision (MAP) on test set. Follows the final scores achieved on the test set alongside the scores achieved by the other four participants.

Data	System	MAP	AvgRec	MRR
Train	**RDI**	**78.3**	**81.2**	**83.3**
Dev	**RDI**	**44.8**	**47.9**	**51.1**
Test	SLS	45.83	51.01	53.66
	ConvKN	45.50	50.13	52.55
	RDI	**43.80**	**47.45**	**49.21**
	QU-IR	38.63	44.10	46.27
	UPC US-MBA	29.09	30.04	34.04
	Baseline	**28.88**	**28.71**	**30.93**

Table 5.Results achieved in SemEval 2016 task 3

We can see that the proposed system achieved very good results compared to the baseline system. The system also achieved comparable results to the systems that achieved first and second positions

6 Conclusion

In this paper, we have introduced a combination of different algorithms which can be used in ranking problems. The performance of different systems has been studied. The paper also presented a comparison with other systems, and the results show that the proposed system is efficient. The results achieved by the proposed system are very promising compared with other systems in the competition.

7 Future work

Future work includes the enhancement of the system using sentence representation in vector space, as well as the combination of this system with other types of features.

We are planning to try sentence representation models like recursive auto encoder (RAE) (Socher et al., 2011) to represent the whole sentence in a vector. Using a model like RAE can facilitate measuring similarity between 2 sentence using vector representations for sentences.

The use of weighted sum for merging the proposed three modules is a very naive method that also suffers from over fitting. So We are planning to merge the proposed modules by means of machine learning (SVR, Neural Network, etc..) instead of using the naïve weighted sum. We are also planning to invest some effort to perform more experiments on the language model module since it has contributed the less within the three modules.

References

Socher, R., Pennington, J., Huang, E. H., Ng, A. Y., & Manning, C. D. (2011, July). Semi-supervised recursive autoencoders for predicting sentiment distributions. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (pp. 151-161). Association for Computational Linguistics.

Zahran, M. A., Magooda, A., Mahgoub, A. Y., Raafat, H., Rashwan, M., & Atyia, A. (2015). Word Representations in Vector Space and their Applications for Arabic. In Computational Linguistics and Intelligent Text Processing (pp. 430-443). Springer International Publishing.

Mikolov, T., Karafiát, M., Burget, L., Cernocký, J., & Khudanpur, S. (2010, September). Recurrent neural network based language model. In *INTERSPEECH* (Vol. 2, p. 3).

Mikolov, T., Kombrink, S., Burget, L., Černocký, J. H., & Khudanpur, S. (2011, May). Extensions of recurrent neural network language model. In Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on (pp. 5528-5531). IEEE.

Nakov, P., Marquez, L., Magdy, W., Moschitti, A., Glass, J. & Randeree, B. (2016). SemEval-2016 Task 3: Community Question Answering. In Proceedings of the 10th International Workshop on Semantic Evaluation. San Diego, California. Association for Computational Linguistics.

Abouenour, L., Bouzouba, K., & Rosso, P. (2010). An evaluated semantic query expansion and structure-

based approach for enhancing Arabic question/answering. International Journal on Information and Communication Technologies

Mahgoub, A. Y., Rashwan, M. A., Raafat, H., Zahran, M. A., & Fayek, M. B. (2014). Semantic query expansion for Arabic information retrieval. ANLP 2014

Ji, Z., Xu, F., Wang, B. and He, B., 2012, October. Question-answer topic model for question retrieval in community question answering. In Proceedings of the 21st ACM international conference on Information and knowledge management.

J. Jeon, W. Croft, and etc. . a framework to predict the quality of answers with non-textual features. In Proc. SIGIR, 2006.

J. Ko, L. Si, and E. Nyberg. A probabilistic framework for answer selection in question answering. In Proc. NAACL/HLT, 2007.

M. Blooma, A. Chua, and D. Goh. A predictive framework for retrieving the best answer. In Proc. SAC, 2008.