Tweaking Word Embeddings for FAQ Ranking

Erick R. Fonseca
University of São Paulo, Brazil
Fondazione Bruno Kessler
rocha@fbk.eu

Simone Magnolini
Fondazione Bruno Kessler
University of Brescia, Italy
magnolini@fbk.eu

Anna Feltracco
Fondazione Bruno Kessler
University of Pavia, Italy
University of Bergamo, Italy
feltracco@fbk.eu

Mohammed R. H. Qwaider
Fondazione Bruno Kessler
Povo-Trento, Italy
qwaider@fbk.eu

Bernardo Magnini
Fondazione Bruno Kessler
Povo-Trento, Italy
magnini@fbk.eu

Abstract

English. We present the system developed at FBK for the EVALITA 2016 Shared Task "QA4FAQ – Question Answering for Frequently Asked Questions". A peculiar characteristic of this task is the total absence of training data, so we created a meaningful representation of the data using only word embeddings. We present the system as well as the results of the two submitted runs, and a qualitative analysis of them.

Italiano. *Presentiamo il sistema sviluppato presso FBK per la risoluzione del task EVALITA 2016 "QA4FAQ - Question Answering for Frequently Asked Questions". Una caratteristica peculiare di questo task é la totale mancanza di dati di training, pertanto abbiamo creato una rappresentazione significativa dei dati utilizzando solamente word embeddings. Presentiamo il sistema assieme ai risultati ottenuti dalle due esecuzioni che abbiamo inviato e un'analisi qualitativa dei risultati stessi.*

1 Introduction

FAQ ranking is an important task inside the wider task of question answering, which represents at the moment a topic of great interest for research and business as well. Analyzing the Frequent Asked Questions is a way to maximize the value of this type of knowledge source that otherwise could be difficult to consult. A similar task was proposed in two SemEval editions (Màrquez et al., 2015) and (Nakov et al., 2016).

Given a knowledge base composed of about 470 questions (henceforth, FAQ question), their respective answers (henceforth, FAQ answers) and metadata (tags), the task consists in retrieving the most relevant FAQ question/answer pair related to the set of queries provided by the organizers.

For this task, no training data were provided, ruling out machine learning based approaches. We took advantage of the *a priori* knowledge provided by word embeddings, and developed a word weighting scheme to produce vector representations of the knowledge base questions, answers and the user queries. We then rank the FAQs with respect to their cosine similarity to the queries.

The paper is organized as follows. Section 2 presents the system we built and Section 3 reports development data we created in order to test our system. In Section 4 we show the results we obtained, followed by Section 5 that presents an error analysis. Finally, Section 6 provides some conclusions.

2 System Description

Our system was based on creating vector representations for each user query (from the test set), question and answer (from the knowledge base), and then ranking the latter two according to the cosine distance to the query.

We created the vectors using the word embeddings generated by Dinu and Baroni (2014) and combined them in a way to give more weight to more important words, as explained below. Since no training data was available, using word embeddings was especially interesting, as they could provide our system with some kind of *a priori* knowl-

edge about similar words.

We applied similar the same operations to queries, FAQ questions and answers, and here we will use the term *text* to refer to any of the three. In order to create vector representations for texts, the following steps were taken:

1. **Tokenization.** The text is tokenized with NLTK's (Bird et al., 2009) Italian model, yielding a token list X.

2. **Filtering.** Stopwords (obtained from NLTK's stopword list) and punctuation signs are discarded from X.

3. **Acronyms Substitution.** Some words and expressions are replaced by their acronyms. We performed this replacement in order to circumvent cases where a query could have an acronym while the corresponding FAQ has the expression fully written, which would lead to a similarity score lower than expected. For example, we replaced *Autorità Idrica Pugliese* with AIP and *Bari* with BA. In total, 21 expressions were checked.

4. **Out-of-vocabolary terms.** When words out of the embedding vocabulary are found in a FAQ question or answer, a random embedding is generated for it[1], from a normal distribution with mean 0 and standard deviation 0.1. The same embedding is used for any new occurrences of that word. This includes any acronyms used in the previous step.

5. **IDF computation.** We compute the document frequency (DF) of each word as the proportion of questions or answers in which it appears[2]. Then, we compute the inverse document frequency (IDF) of words as:

$$\mathrm{IDF}(w) = \begin{cases} \frac{1}{\mathrm{DF}(w)}, & \text{if } \mathrm{DF}(w) > 0 \\ 10, & \text{otherwise} \end{cases} \quad (1)$$

   We found that tweaking the DF by decreasing FAQ tags count could improve our system's performance. When counting words in questions and answers to compute their DF, we ignore any word present among the tags for that FAQ entry. Thus, tag words, which are supposed to be more relevant, have a lower DF and higher IDF value.

6. **Multiword expressions.** We compute the embeddings for 15 common multiword expressions (MWEs) we extracted from the FAQ. They are computed as the average of the embeddings of the MWE components, weighted by their IDF. If an MWE is present in the text, we add a token to X containing the whole expression, but do *not* remove the individual words. An example is *codice cliente*: we add *codice_cliente* to X, but still keep *codice* and *cliente*.

7. **SIDF computation.** We compute the Similarity-IDF (SIDF) scores. This metric can be seen as an extension of the IDF which also incorporates the DF of similar words. It is computed as follows:

$$\mathrm{SIDF}(w) = \frac{1}{\mathrm{SDF}(w)} \quad (2)$$

$$\mathrm{SDF}(w) = \mathrm{DF}(w) + \sum_{w_i \in W_{sim}} \cos(w, w_i)\,\mathrm{DF}(w_i) \quad (3)$$

   Here, W_{sim} denotes the set of the n most similar words to w which have non-zero DF. Note that under this definition, SDF is never null and thus we don't need the special case as in the IDF computation. We can also compute the SIDF for the MWEs introduced to the texts.

8. **Embedding averaging.** After these steps, we take the mean of the embeddings, weighted by the SIDF values of their corresponding words:

$$v = \frac{\sum_{w \in X} \mathrm{E}(w)\,\mathrm{SIDF}(w)}{|X|} \quad (4)$$

   Here, v stands for the vector representation of the text and $\mathrm{E}(\cdot)$ is the function mapping words and MWEs to their embeddings. Note that we do not remove duplicate words.

[1] Out of vocabulary words that only appear in the queries are removed from X.

[2] When we are comparing queries to FAQ questions, we only count occurrences in questions. Likewise, when comparing queries to answers, we only count in answers.

FAQ Entry	id	272
	question	Cos'è la quota fissa riportata in fattura?
	answer	La quota fissa, prevista dal piano tariffario deliberato, è addebitata in ciascuna fattura, fattura, ed è calcolata in base ai moduli contrattuali ed ai giorni di competenza della fattura stessa. La quota fissa è dovuta indipendentemente dal consumo in quanto attiene a parte dei costi fissi che il gestore sostiene per erogare il servizio a tutti. Quindi nella fattura è addebitata proporzionalmente al periodo fatturato.
	tag	fattura, quota, fissa, giorni, canone acqua e fogna, quota fissa, costi fissi, quote fisse
DevSet	paraphrased query	Cosa si intende per quota fissa nella fattura?
	answer-driven query	La quota fissa è indipendente dai consumi?

Table 1: Example of our development set.

In this process, the IDF and SIDF values are calculated independently for answers and questions in the FAQ When processing queries, the value actually used depends on which one we are comparing the query vectors with.

After computing vectors for all texts, we compute the cosine similarity between query vectors and FAQ questions and also between queries and answers. For each FAQ entry, we take the highest value between these two as the system confidence for returning that entry as an answer to the query.

3 Evaluating our system

In order to evaluate our system, we created a development set and we calculate a baseline as a reference threshold.

3.1 Development Set

We manually created a dataset of 293 queries to test our systems. Each query in the dataset is associated to one of the entries provided in the knowledge base. In particular, the dataset is composed by 160 *paraphrased queries* and 133 *answer driven queries*. The *paraphrased queries* are queries obtained by paraphrasing original questions; the *answer queries* are generated without considering the original FAQ questions, but have an answer in the knowledge base. Table 1 shows an example of a *paraphrased query* and an *answer driven query* for FAQ 272 of the knowledge base.

Given the technical domain of the task, most of the generated *paraphrases* recall lexical items of the original FAQ question (e.g. "uso commerciale", "scuola pubblica", etc..). Differently, the *answer driven queries* are not necessarily similar in content and lexicon to the FAQ question; instead we expected it to have a very high similarity with the answer.

We guided the development of our system evaluating it with different versions of this dataset. In particular, version 1 is composed by 200 queries, begin 160 *paraphrased* and 40% answer driven, and version 2 is composed by 266 queries, 133 *paraphrased* and 133 *answer driven*.

Merging *paraphrased queries* and *answer driven queries* (in different proportions) allows us to create a very heterogeneous dataset; we expected the test set and, in general, the questions by users to be as much varied.

3.2 Baseline

Two baseline systems were built using Apache Lucene[3]. *FBK-Baseline-sys1* was built by indexing for each FAQ entry a Document with two fields (id, FAQ question), while *FBK-Baseline-sys2* was built by indexing for each FAQ entry a Document with three fields (id, FAQ question, FAQ answer).

4 Results

In Table 2 we report the results of the two runs of our system compared with the official baseline provided by the organizers. The only difference in our first two runs was that the first one always tried to retrieve an answer, while the second one would abstain from answering when the system confidence was below 0.5.

The organizers baseline (*qa4faq-baseline*[4]) was built using Lucene by having a weighted-index. For each FAQ entry a Document with four fields (id, FAQ question(*weight=4*), FAQ answer(*weight=2*), tag(*weight=1*)).

We use three different metrics to evaluate the system: Accuracy@1, that is the official score to

[3] https://lucene.apache.org/

[4] https://github.com/swapUniba/qa4faq

	Test set		
	Accuracy@1	MAP	Top 10
run 1	35.87	51.12	73.94
run 2	37.46	50.10	71.91
qa4faq-baseline	40.76	58.97	81.71
FBK-Baseline-sys1	39.79	55.36	76.15
FBK-Baseline-sys2	35.16	53.02	80.92

Table 2: Results on the test set. Accuracy@1: official score, MAP: Mean Average Precision, Top 10: correct answer in the first 10 results.

rank the systems, MAP and $Top10$. Accuracy@1 is the precision of the system taking into account only the first answer; it is computed as follows:

$$Accuracy@1 = \frac{(n_c + n_u * \frac{n_c}{n})}{n} \quad (5)$$

Where n_r is the number of correct quries, n_u is the number of unanswered queries and n is the number of questions. MAP is the Mean Average Precision that is the mean of the average precision scores for each query, i.e. the inverse of the ranking of the correct answer. $Top10$ is the percentage of query with the correct answer in the first 10 positions. Both our approach runs underperformed compared with the baseline in all the three metrics we use to evaluate the systems.

Comparing our runs, it is interesting to notice that *run 2* performs better while evaluated with $Accuracy@1$, but worse in the other two metrics; this suggests that, even in some cases where the system confidence was below the threshold, the correct answer was among the top 10.

5 Error Analysis

The results of our system on the development set, described in Section 3.1, compared with the official baseline are reported in Table 3.

As can be seen, both the runs outperform the baseline in every metric, especially in the $Accuracy@1$.

This difference of behavior enlightens that there is a significant difference between the development set and the test set. The systems were developed without knowing the target style, and without training data, so is not surprising that the system is not capable of style adaptation.

An interesting aspect that describes the difference between development set and test set is reported in Table 4: the average and the standard deviation of the number of tokens of every query. In the first line is possible to notice that, not only, our development queries has , in average, more tokens than the test queries, but also that the standard deviation is significantly lower. This distribution of tokens is in line with a qualitative check of the test set. The test set includes incomplete sentences, with only keywords, e.g. *"costo depurazione"*, alongside long questions that include verbose description of the situation e.g. *"Mia figlia acquisterà casa a bari il giorno 22 prossimo. Come procedere per l intestazione dell utenza? Quali documenti occorrono e quali i tempi tecnici necessari?"*. Instead the development set is composed by queries more similar in their structure and well formed.

All systems perform, almost, in the same way according to the data sets: in the two versions of the development set the correct queries are longer with a higher standard deviation compared to the wrong ones; on the other hand, in the test set the correct queries are shorter with a lower standard deviation.

We did a qualitative analysis of the result of our systems; we limited our observation to the 250 queries of the test set for which the right answer was not in the first ten retrieved by our systems. We considered these cases to be the worst and wanted to investigate whether they present an issue that cannot be solved using our approach.

We present in this section some of these cases. In Example 1, the answer of the system is weakly related with the query: the query is very short and its meaning is contained in both the gold standard and in the system answer. In the gold standard the substitution of the counter (*"sostituzione del contatore"*) is the main focus of the sentence, and the other part is just a specification of some detail (*"con saracinesca bloccata"*).

In the system answer the substitution of the counter (*"sostituzione del contatore"*) is the effect of the main focus (*"Per la telelettura"*), but our approach cannot differentiate these two types of text not directly related with the query.

Example 1
Query: *sostituzione del contatore*
Gold standard: *Come effettuare il cambio del contatore vecchio con saracinesca bloccata?*
System answer: *Per la telelettura il contatore sara sostituito con un nuovo contatore?*

A similar issue is visible in Example 2. In this

	Version 1			Version 2		
	Accuracy@1	MAP	Top 10	Accuracy@1	MAP	Top 10
Run 1	72.00	79.64	95.00	66.17	74.77	92.48
Run 2	72.45	77.55	92.00	66.36	73.26	90.23
qa4faq-baseline	69.00	76.22	89.50	60.15	70.22	88.72
FBK-baseline-sys1	49.00	58.63	76.50	39.47	49.53	68.05
FBK-baseline-sys2	52.00	62.69	82.50	49.62	62.10	86.09

Table 3: Results on the development sets. Accuracy@1: official score, MAP: Mean Average Precision, Top 10: correct answer in the first 10 result.

		Version 1	Version 2	Test set
Queries		11.42 +- 4.12	11.20 +- 3.95	7.96 +- 7.27
R1	Answered queries	11.42 +- 4.12	11.20 +- 3.95	7.96 +- 7.27
	Right queries	11.63 +- 4.15	11.41 +- 4.06	7.32+- 5.44
	Wrong queries	10.88 +- 4.00	10.78 +- 3.69	8.32 +- 8.09
R2	Answered queries	11.56 +- 4.12	11.30 +- 3.94	8.09 +- 7.41
	Right queries	11.77 +- 4.12	11.52 +- 4.04	7.37 +- 5.47
	Wrong queries	11.02 +- 4.06	10.86 +- 3.71	8.52 +- 8.33
B	Answered queries	11.42 +- 4.12	11.20 +- 3.95	7.97 +- 7.27
	Right queries	11.94 +- 4.34	11.73 +- 4.35	7.54 +- 5.98
	Wrong queries	10.26 +- 3.31	10.40 +- 3.09	8.26 +- 8.02

Table 4: Average and standard deviation of the number of tokens per query. R1: Run1, R2: Run2, B: Organizers Baseline *qa4faq-baseline*.

case, the first part (*"Quali sono i tempi di allaccio di un contatore"*) of the system answer matches, almost exactly, the query, but as in Example 1, the second part (*"in caso di ripristino in quanto l'abitazione aveva già la fornitura?"*), which is not very relevant to the query, was not enough to reduce the overall ranking of this FAQ. We think this issue could be avoided with some more features, but this would require some training data for a machine learning approach, or some knowledge of the domain to craft a rule approach.

Example 2
Query: *quali sono i tempi di attivazione di un contatore ?*
Gold standard: *Quali sono i tempi previsti per ottenere un allacciamento?*
System answer: *Quali sono i tempi di allaccio di un contatore in caso di ripristino in quanto l'abitazione aveva già la fornitura?*

In some cases, like in Example 3, the semantic match (like common or related words in both sentences) is not enough to understand the relationship, or could me misleading. Some knowledge of the world and some cause-effect reasoning is needed to understand that the gold standard is more related to the query than the system answer. Even if the balance (*"conguaglio"*) and time expressions (*"quando"*, *"luglio e agosto e un po di settembre"*) are present in both query and system answer, and not in the gold standard, they are not useful to find the correct answer.

Example 3
Query: *ho ricevuto una bolletta di conguaglio di e 426.69 , ma son mancata da casa a luglio e agosto e un po di settembre , senza consumare , come mai?*
Gold standard: *Perche ho ricevuto una fattura elevata?*
System answer: *Il conguaglio quando avviene?*

Alongside this issue, there are some cases (Example 4) where our system answers correctly, but due to the semi-automatic nature of the gold standard it has been considered wrong.

Example 4
Query: *chi paga la portella del contatore?*
Gold standard: *Come richiedere la sostituzione dello sportello della nicchia contatore?*
System answer: *Chi paga la portella del contatore?*

Example 5 represents one of the cases in which the systems answer has been considered wrong but is more related with the query than the gold standard.

Example 5
Query: *abito in un condominio con 5 famiglie . se alla scadenza di una bolletta uno dei condomini non vuole pagare la sua quota , possono gli altri 4 pagare la loro parte su un altro bollettino postale?*
Gold standard: *Quali sono le modalita di pagamento delle fatture?*
System answer: *Contratto condominiale, di cui uno moroso come comportarsi?*

6 Conclusion

We reported the system we used in the EVALITA 2016 QA4FAQ shared task, as well as the development set we created to evaluate it and an analysis of our results.

We found that while our system performed below the baseline in the official test set, we had superior performance on our in-house development set. This is apparently related to the different style of the two sets: ours has longer queries, which are more homogeneous with respect to size, while the official one has many very short queries and a few very large ones.

It could be argued that the official test set represents a more realistic scenario than the development set we created, since it contains actual user queries, thus diminishing the relevance of our results. However, further analysis showed that in a number of cases, our system returned a more appropriate FAQ question/answer than what was in the gold standard, due to the gold standard semi-automatic nature.

We hypothesize that our system performed better than what seems from the official results; however, due to the size of the test set, it would be prohibitive to check it manually and arrive at a more precise accuracy.

References

Steven Bird, Edward Loper, and Ewan Klein. 2009. *Natural Language Processing with Python*. O'Reilly Media Inc.

Georgiana Dinu and Marco Baroni. 2014. Improving zero-shot learning by mitigating the hubness problem. *arXiv preprint arXiv:1412.6568*.

Lluís Màrquez, James Glass, Walid Magdy, Alessandro Moschitti, Preslav Nakov, 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)*.

P Nakov, L Marquez, A Moschitti, W Magdy, H Mubarak, AA Freihat, J Glass, and B Randeree. 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*.