Methods combination and ML-based re-ranking of multiple hypothesis for question-answering systems

Arnaud Grappy, Brigitte Grau, Sophie Rosset

Methods Combination and ML-based Re-ranking of Multiple Hypothesis for Question-Answering Systems

Arnaud Grappy
LIMSI-CNRS
arnaud.grappy@limsi.fr

Brigitte Grau
LIMSI-CNRS
ENSIIE
brigitte.grau@limsi.fr

Sophie Rosset
LIMSI-CNRS
sophie.rosset@limsi.fr

Abstract

Question answering systems answer correctly to different questions because they are based on different strategies. In order to increase the number of questions which can be answered by a single process, we propose solutions to combine two question answering systems, QAVAL and RITEL. QAVAL proceeds by selecting short passages, annotates them by question terms, and then extracts from them answers which are ordered by a machine learning validation process. RITEL develops a multi-level analysis of questions and documents. Answers are extracted and ordered according to two strategies: by exploiting the redundancy of candidates and a Bayesian model. In order to merge the system results, we developed different methods either by merging passages before answer ordering, or by merging end-results. The fusion of end-results is realized by voting, merging, and by a machine learning process on answer characteristics, which lead to an improvement of the best system results of 19 %.

1 Introduction

Question-answering systems aim at giving short and precise answers to natural language questions. These systems are quite complex, and include many different components. Question-Answering systems are generally organized within a pipeline which includes at a high level at least three components: questions processing, snippets selection and answers extraction. But each module of these systems is quite different. They are based on different knowledge sources and processing. Even if the global performance of these systems are similar, they show great disparity when examining local results. Moreover there is no question-answering system able to answer correctly to all possible questions. Considering all QA evaluation campaigns in French like CLEF, EQUER or Quæro, or for other languages like TREC, no system obtained 100% correct answers at first rank. A new direction of research was built upon these observations: how can we combine correct answers provided by different systems?

This work deals with this issue[1] . In this paper we describe different experiments concerning the combination of QA systems. We used two different available systems, QAVAL and RITEL, while RITEL includes two different answer extraction strategies. We propose to merge the results of these systems at different levels. First, at an intermediary step (for example, between snippet selection and answer extraction). This approach allows to evaluate a fusion process based on the integration of different strategies. Another way to proceed is to execute the fusion at the end of each system. The aim is then to choose between all the candidate answers the best one for each question. Such an approach has been successfully applied in the information retrieval field, with the definition of different functions for combining results of search engines (Shaw and Fox, 1994). However, in QA, the problem is different as answers to questions are not made of a list of answers, but are made of excerpts of texts, which may be different in their writing, but which correspond to a unique and same answer. Thus, we propose fusion methods that rely on the information generally computed by QA systems, such as score, rank, an-

[1]This work was partially financed by OSEO under the Quro program

Proceedings of the Workshop on Innovative Hybrid Approaches to the Processing of Textual Data (Hybrid2012), EACL 2012, pages 87–96,
Avignon, France, April 23 2012.

swer redundancy, etc. We defined new voting and scoring functions, and a machine learning system to combine these features. Most of the strategies presented here allow a clear improvement (up to 19 %) on the first ranked correct answers.

In the following, related work is presented in the section 2. We then describe the different systems used in this work (Section 3.1 and 3.2). The proposed approach are presented (Section 4 and 5). The methods and the different systems are then evaluated on the same corpus.

2 Related work

QA system hybridization often consists in merging end-results. The first studies presented here aim at merging the results of different strategies for finding answers in the same set of documents. (Jijkoun and Rijke, 2004) developed several strategies for answering questions, based on different paradigms for extracting answers. They search for answers in a knowledge base or by applying extraction patterns or by selecting the n-grams the closest to the question words. They defined different methods for recognizing the similarity of two answers: equality, inclusion and an edit distance. The merging of answers is realized by summing the confidence scores of similar answers and leads to improve the number of right answers at first rank of 31 %.

(Tellez-Valero et al., 2010) combine the output of QA systems, whose strategy is not known. They only dispose of the provided answers associated with a supporting snippet. Merging is done by a machine learning approach, which combines different criteria such as the question category, the expected answer type, the compatibility between the provided answer and the question, the system which was applied and the rate of question terms in the snippet. When applying this module on the CLEF QA systems which were run on the Spanish data, they obtain a better MRR[2] value than the best system from 0.62 up to 0.73.

In place of diversifying the answering strategies, another possibility is to apply a same strategy on different collections. (Aceves-Pérez et al., 2008) apply classical merging strategies to multilingual QA systems, by merging answers according to their rank or by combining their confidence scores, normalized or not. They show that the combination of normalized scores obtains results which are better than a monolingual system (MRR from 0.64 up to 0.75). They also tested hybridization at the passage level by extracting answers from the overall set of passages which proved to be less relevant than answer merging.

(Chalendar et al.,) combine results obtained by searching the Web in parallel to a given collection. The combination which consists in boosting answers if they are found by the two systems is very effective, as it is less probable to find same incorrect answers on different documents.

The hybridization we are interested in concerns the merging of different strategies and different system capabilities in order to improve the final result. We tested different hybridization levels, and different merging methods. One is closed to (Tellez-Valero et al., 2010) as it is based on a validation module. Other are voting and scoring methods which have been defined according to our task, and are compared to classical merging scheme which have been proposed in information retrieval (Shaw and Fox, 1994), ComSum and CombMNZ.

3 The Question-Answering systems

3.1 The QAVAL system

3.1.1 General overview

QAVAL(Grappy et al., 2011) is made of sequential modules, corresponding to five main steps (see Fig. 1). The question analysis provides main characteristics for retrieving passages and for guiding the validation process. Short passages of about 300-character long are obtained directly from the search engine Lucene and are annotated with question terms and their weighted variants. They are then parsed by a syntactic parser and enriched with the question characteristics, which allows QAVAL to compute the different features for validating or discarding candidate answers.

A specificity of QAVAL relies on its validation module. Candidate answers are extracted according to the expected answer type, i.e. a named entity or not. In case of a named entity, all the named entities corresponding to the expected type are extracted while, in the second case, QAVAL extracts all the noun phrases which are not question phrases. As many candidate answers can be extracted, a first step consists in recognizing obvious false answers. Answers from a passage that does

[2]Mean Reciprocal Rank

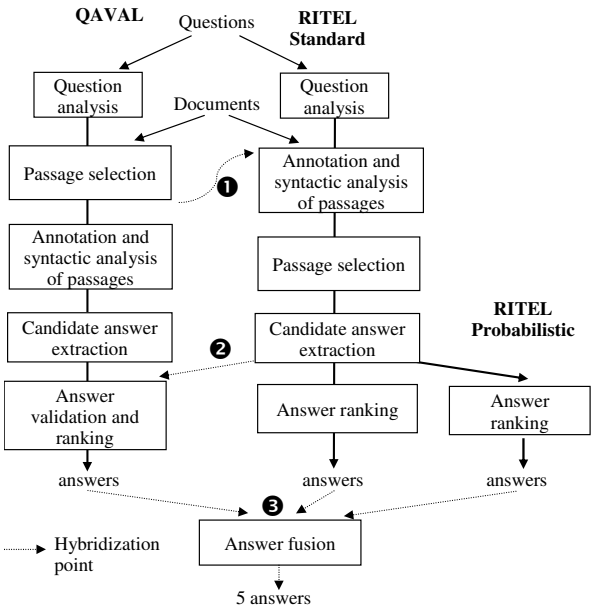

Figure 1: The QAVAL and RITEL systems and their possible hybridizations

not contain all the named entities of the question are discarded. The remaining answers are then ranked based on a learning method which combines features characterizing the passage and the candidate answer it provides. The QAVAL system has been evaluated on factual questions and obtains good results.

3.1.2 Answer ranking by validation

A machine based learning validation module provides scores to each candidate answer. Features relative to passages aim at evaluating in which part a passage conveys the same meaning as the question. They are based on lexical features, as the rate of question words in the passage, their POS tag, the main terms of the question, etc.

Features relative to the answer represent the property that an answer has to be of an expected type, if explicitly required, and to be related to the question terms. Another kind of criterion concerns the answer redundancy: the most frequent an answer is, the most relevant it is. Answer type verification is applied for questions which give an explicit type for the answer, as in "Which president succeeded Georges W. Bush?" that expects as answer the name of a president, more specific than the named entity type PERSON. This module (Grappy and Grau, 2010) combines results given by different kinds of verifications, based on named entity recognizers and searches in corpora. To evaluate the relation degree of an answer with the question terms, QAVAL computes i) the longest chain of consecutive common words between the question plus the answer and the passage; ii) the average distance between the answer and each of the question words in the passage.

Other criteria are the passage rank given by using results of the passage analysis, the question category, i.e. definition, characterization of an entity, verb modifier or verb complement, etc.

3.2 The RITEL systems

3.3 General overview

The RITEL system (see Figure 1) which we used in these experiments is fully described in (Bernard et al., 2009). This system has been developed within the framework of the Ritel project which aimed at building a human-machine dialogue system for question-answering in open domain (Toney et al., 2008).

The same multilevel analysis is carried out on both queries and documents. The objective of this analysis is to find the bits of information that may be of use for search and extraction, called *pertinent information chunks*. These can be of different categories: named entities, linguistic entities (e.g., verbs, prepositions), or specific entities (e.g., scores). All words that do not fall into such chunks are automatically grouped into chunks via a longest-match strategy. The analysis is hierarchical, resulting in a set of trees. Both answers and important elements of the questions are supposed to be annotated as one of these entities.

The first step of the QA system itself is to build a search descriptor (SD) that contains the important elements of the question, and the possible answer types with associated weights. Answer types are predicted through rules based on combinations of elements of the question. On all secondary and mandatory chunks, the possible transformations (synonym, morphological derivation, etc.) are indicated and weighted in the SD. Documents are selected using this SD. Each element of the document is scored with the geometric mean of the number of occurrences of all the SD elements that appear in it, and sorted by score, keeping the n-best. Snippets are extracted from the document using fixed-size windows and scored using the geometrical mean of the number of oc-

currences of all the SD elements that appear in the snippet, smoothed by the document score.

3.3.1 Answer selection and ranking

Two different strategies are implemented in RITEL. The first one is based on distance between question words and candidate answer, named RITEL Standard. The second one is based on a Bayesian model, named RITEL Probabilistic.

Distance-based answer scoring The snippets are sorted by score and examined one by one independently. Every element in a snippet with a type found in the list of expected answer types of the SD is considered an answer candidate. RITEL associates to each candidate answer a score which is the sum of the distances between itself and the elements of the SD. That score is smoothed with the snippet score through a δ-ponderated geometric mean. All the scores for the different instances of the same element are added together. The entities with the best scores then win. The scores for identical (type,value) pairs are added together and give the final scoring to the candidate answers.

Answer scoring through Bayesian modeling This method of answer scoring is built upon a Bayesian modeling of the process of estimating the quality of an answer candidate. This approach relies on multiple elementary models including element co-occurrence probabilities, question element appearance probability in the context of a correct answer and out of context answer probability. The model parameters are either estimated on the documents or are set empirically. This system has not better result than the distance-based one but is interesting because it allows to obtain different correct answers.

3.4 Systems combination

The systems we used in these experiments are very different especially with respect to the passage selection and the answer extraction and scoring methods. The QAVAL system proceeds to the passage selection before any analysis while the two RITEL systems do a complete and multi-level analysis on the documents before the passage selection. Concerning the answer extraction and scoring, the QAVAL system uses an answer validation process based on machine learning approach while the answer extraction of the RITEL-S system uses a distance-based scoring and the RITEL-P Bayesian models. It seems then interesting to combine these various approaches in a in-system way (see Section 4): (1) the passages selected by the QAVAL system are provided as document collection to the RITEL systems; (2) the candidate answers provided by the RITEL systems are given to the answer validation module of the QAVAL system.

We also worked, in a more classical way, on interleaving results of answer selection methods (see Section 5 and 6). These methods make use of the various information provided by the different systems along with all candidate answers.

4 Internal combination

4.1 QAVAL snippets used by RITEL

The RITEL system proceeds to a complete analysis of the document which is used during the document and selection extraction procedure and obtains 80.3% of the questions having a correct answer in at least one passage. The QAVAL system extracts short passages (150) using Lucene and obtains a score of 88%. We hypothesized that the RITEL's fine-grained analysis could better work on small collection than on the overall document collection (combination 1 Fig. 1). We consider the passages extracted by the QAVAL system being a new collection for the RITEL system. First, the analysis is done on this new collection and the analysis result is indexed. Then the general question-answering procedures are applied: question analysis, SD construction, document and snippet extraction and then answer selection and ranking. The two answer extraction methods have been applied and the results are presented in the Table 1. This simple approach does not allow any

	All documents		QAVAL' snippets	
	Ritel-S	Ritel-P	Ritel-S	Ritel-P
top-1	34.0%	22.4%	29.9%	22.4%
MRR	0.41	0.29	0.38	0.32
top-20	61.2%	48.7%	54.4%	49.7%

Table 1: Results of Ritel systems (Ritel-S used the distance-based answer scoring, Ritel-P used the Bayesian modeling) working on the QAVAL' snippets.

improvement. Actually all the results are worsening, except maybe for the Ritel-P systems (which is actually not the best one). One of our hypothesis is that the QAVAL snippets are too short and

do not fit the criteria used by the RITEL system.

4.2 Answer validation

In QAVAL, answer ranking is done by an answer validation module (fully described in section 3.1). The candidate answers ranked by this module are associated to a confidence score. The objective of this answer validation module is to decide whether the candidate answer is correct or not given an associated snippet. The objective is to use this answer validation module on the candidate answers and the snippets provided by all the systems (combination 2 Fig. 1). Unfortunately, this method did not obtain better results than the best system. We assume that this module being learnt on the QAVAL data only is not robust to different data and more specifically to the passage length which is larger in RITEL than in QAVAL. A possible improvement could be to add answers found by the RITEL system in the training base.

5 Voting methods and scores combination

These methods are based on a comparison between the candidate answers: are they identical ? An observation that can be made concerning the use of a strict equality between answers is that in some cases, 2 different answers can be more or less identical. For example if one system returns "Sarkozy" and another one "Nicolas Sarkozy" we may want to consider these two answers as identical. We based the comparison of answers on the notion of *extended equality*. For that, we used morpho-syntactic information such as the lemmas and the part of speech of each words of the answers. The TreeTagger tool[3] has been used. An answer R_1 is then considered as included in an answer R_2 if all non-empty words of R_1 are included in R_2. Two words having the same lemma are considered as identical. For example "chanta" and "chanterons" are identical because they share the same lemma "chanter". Adjectives, proper names and substantives are considered as non-empty words. Following this definition, two answers R_1 and R_2 are considered identical if R_1 is included in R_2 and R_2 in R_1.

[3] www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger

5.1 Merge based on candidate answer rank

The first information we used takes into account the rank of the candidate answers. The hypothesis beyond this is that the systems often provide the correct answer at first position, if they found it.

5.1.1 Simple interleaving

The first method, and probably the simplest, is to merge the candidate answers provided by all the systems: the first candidate answer of the first system is ranked in the first position; the first answer of the second system is ranked in the second position; the second answer of the first system is ranked in the third position, and so on. If one answer was already merged (because ranked at a higher rank by another system), it is not used. We choose to base the systems order given their individual score. The first system is QAVAL, the second RITEL-S and the third RITEL-P. Following that method, the accuracy (percentage of correct answers at first rank) is the one obtained by the best system. But we assume that the MRR at the top-n (with $n > 1$) would be improved.

5.1.2 Sum of the inverse of the rank

The simple interleaving method does not take into account the answer rank provided by the different systems. However, this information may be relevant and was used in order to merge candidate answer extracted from different document collection, Web articles and news paper (Chalendar et al.,). In our case, answers are extracted from the same document collection by the different systems. Then it is possible that the same wrong answers will be extracted by the different systems.

A first possible method to take into account the rank provided by the systems is to weight the candidate answer using this information. For a same answer provided by the different systems, the weight is the sum of the inverse of the rank given by the systems. To compare the answers the strict equality is applied. If a system ranks an answer at the first position and another system ranks the same answer at the second position, the weight is 1.5 ($1 + \frac{1}{2}$). The following equation express in a more formalized way this method.

$$weight = \sum \frac{1}{rank}$$

Comparing to the previous method, that one should allow to place more correct answers at the first rank.

5.2 Using confidence scores

In order to rank all their candidate answers, the systems used a confidence score associated to each candidate answer. We then wanted to use these confidence scores in order to re-rank all the candidate answers provided by all the systems. But this is only possible if all systems produce comparable scores. This is not the case. QAVAL produces scores ranging from -1 to +1. RITEL-P, being probabilistic, produces a score between 0 and +1. And RITEL-S does not use strict interval and the scores are potentially ranged from $-\infty$ to $+\infty$. The following normalization (a linear regression) has been applied to the RITEL-S and RITEL-P scores in order to place it in the range -1 to 1.

$$value_{normalized} = \frac{2 * value_{origin}}{val_{Min} - val_{Max}} - 1$$

5.2.1 Sum of confidence scores

In order to compare our methods with classical approaches, we used two methods presented in (Shaw and Fox, 1994):

- **CombSum** which adds the different confidence scores of an answer given by the different systems;
- **CombMNZ** which adds the confidence scores of the different systems and multiply the obtained value by the number of systems having found the considered answer.

5.2.2 Hybrid method

An hybrid method combining the rank and the confidence score has been defined. The weight is the sum of two elements: the higher confidence score and a value taking into account the rank given by the different systems. This value is dependent on the number of answers, the type of the equality (the answers are included or equal) which results in the form of a bonus, and the rank of the different considered answers. The weight of an answer *a* to a question *q* is then:

$$w(a) = s(a) + \prod be * (|a(q)| - \sum r(a)) \quad (1)$$

with *be* the equality bonus, *w* the weight, *s*, the score and *r* the rank.

The equality bonus, *found empirically*, is given for each systems pair. The value is 3 if the two answers are equal, 2 if an answer is included in the other and 1 otherwise. When an answer is found by two or more systems, the higher confidence score is kept. The result of this method is that the answers extracted by more than one system are favored. An answer found by only one system, even with a very high confidence score, may be downgraded.

6 Machine-learning-based method for answer re-ranking

To solve a re-ranking problem, machine learning approaches can be used (for example (Moschitti et al., 2007)). But in most of the cases, the objective is to re-rank answers provided by one system, that means to re-rank multiple hypotheses from one system. In our case, we want to re-rank multiple answers from different systems. We decided to use an SVM-based approach, namely SVMrank (Joachims, 2006), which is well adapted to our problem. An important aspect is then to choose the pertinent features for such a task. Our objective is to consider robust enough features to deal with different systems' answers without introducing biases. Two classes of characteristic should be able to give a useful representation of the answers: those related to the answer itself and those related to the question.

6.1 Answer characteristics

First of all, we should use the rank and the score as we did in the preceding merging methods. The problem may appear here because not all candidate answers are found by the different systems. In that case, the score and the rank given to these systems is then -2. It guarantees us that the features are out of the considered range $[-1, +1]$. Considering that, it may be useful to know which system provided the considered answer. For each answer all systems having found that answer are indicated. Moreover this information may help to distinguish answers coming from for example QAVAL and RITEL-S or RITEL-P from answers coming from RITEL-S and RITEL-P. The two RITEL systems share most of the modules and their answers may have the same problems. Concerning the answer, another aspect may be of interest: how many time this answer has been found? The question is not, how many times the answer appears in the documents but how many times the answer appears in a context allowing this answer

to be considered as a candidate answer. We used the number of different snippets selected by the systems in which that answer was found.

6.2 Question characteristics

When observing the results obtained by the systems on different questions, we observed that the "kind" of the question has an impact on the systems' performance. More specifically, it is largely accepted in the community that at least two criteria are of importance: the length of the question, and the type of the expected answer (EAT).

Question length We may consider that the length of the questions is more or less a good indicator for the complexity level of the question. The number of non-empty words of the question can then be a interesting feature.

Expected answer type One of the task of the question processing, in a classical Question-Answering system, is to decide of which type will be the answer. For example, for a question like *Who is the president of France?* the type of the expected answer will be a named entity of the class *person* and for a question like *what wine to drink with seafood?* that the EAT is not a named entity. (Grappy, 2009) observed that the QAVAL system is better when the EAT is of a named entity class. It is possible that adding this information will, during the learning phase, positively weight an answer coming from RITEL when the EAT is not a named entity.

The value of this feature indicates the compatibility of the answer and the EAT. We used the method presented in (Grappy and Grau, 2010) and already used for the answer validation module of the QAVAL system. This method is based on a ML-based combination of different methods using named entity dictionaries, wikipedia knowledge, etc. This system gives a confidence score, ranging from -1 to +1 which indicates the confidence the system has in compatibility between the answer and the EAT. In some cases, the question processing module may indicate if the EAT is of a more fine-grained entity. For example, the question *Who is the president of France?* is not only waiting for a *person* but more precisely for a person having the function of a *president*. A new feature is then added. If the EAT is a fine-grained named entity, then the value is 1 and -1 otherwise.

7 Experiments and results

7.1 Data and observations

For the training of the SVM model, we used the answers to 104 questions provided by the 2009 Quaero evaluation campaign (Quintard et al., 2010). Only 104 questions have been used because we need to have at least one correct answer provided by at least one system in the training base for each question. Models have been trained using 5, 10, 15 and 20 answers for each system.

For the evaluation, we used 147 factoid questions used in the 2010 Quaero[4] evaluation campaign. The document collection is made of 500,000 Web pages[5]. We used the Mean Reciprocal Rank (MRR) as it is a usual metric in Question-Answering on the first five candidate answers. The MRR is the average of the reciprocal ranks of all considered answers. We also used the top-1 metric which indicates the number of correct answers ranked at the first position.

The baseline results, provided by each of the three systems, are presented in Table 2. QAVAL and RITEL-S have quite similar results which are higher than those obtained by the RITEL-P system. We can observe that, within the 20 top ranks, 38% of the questions have an answer given by all the systems, 76 % by at least 2 systems and 21% receive no correct answers. The best possible result that could be obtained by a perfect fusion method is also indicated in this table (0.79 of MRR and 79% for top-1). Such a method would lead to rank first each correct answer found by at least a system. Figure 2 presents the answer repartition between ranks 2 and 20 (the numbers of correct answers in first rank are given in Table 2). This figure shows that the systems ranked the correct answer mostly in the first positions. That means that these systems are relatively effective for re-ranking their own candidate answers. Very

System	MRR	% top-1 (#)
QAVAL	0.45	36 (53)
RITEL-S	0.41	32 (47)
RITEL-P	0.26	18 (27)
Perfect fusion	0.79	79 (115)

Table 2: Baseline results

[4] http://www.quaero.org

[5] crawled by Exalead http://www.exalead.com/

few correct answers are ranked after the tenth position. Following these observations, the evaluations are done on the first 10 candidate answers.

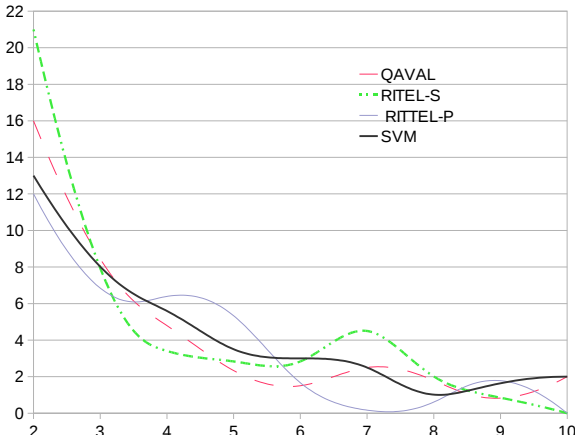

Figure 2: Answer repartition

7.2 Results and analysis

Table 3 presents the results obtained with the different merging methods: simple interleaving (*Inter.*), Sum of the inverse of the rank, CombSum, CombMNZ, hybrid method (*Hyb. Meth.*) and SVM model. In order to evaluate the impact of the RITEL-P (which achieved less good results), the results are given using two (QAVAL and RITEL-S) or three systems.

Method	MRR (2 sys. / 3 sys.)	% Top-1 (#) (2 sys. / 3 sys.)
Inter.	0.47 / 0.45	36 (53) / 36 (53)
$\sum \frac{1}{rang}$	0.48 / 0.46	38 (56)/ 36 (53)
CombSum	0.46 / 0.44	38 (56) / 34 (50)
CombMNZ	0.46/ 0.44	38 (56) / 35 (51)
Hyb. meth.	0.49 /0.44	40 (58) / 34 (50)
SVM	0.48 / **0.51**	39 (57) / **42** (62)
QAVAL	0.44	36 (53)

Table 3: General results.

As shown in Table 3, the different methods improve the results and the best method is the SVM-based model which allows an improvement of 19% of correct answer at first rank. This result is significantly better than the baseline result and this method can be considered as very effective. Figure 2 shows the results of this model. In order to validate our choice of using the SVM-Rank model, we also tested the use of a combination of decision trees, as QAVAL obtained

# candidate answers	% Top-1 (#)
20	39 (58)
15	39 (58)
10	43 (63)
5	37 (55)

Table 4: Impact of the number of candidate answers

normalization	MRR	# Top-1
without	0.49	58 (39%)
with	0.51	63 (43%)

Table 5: Impact of the normalization

good results with this classifier in the validation module. We obtained a MRR of 0.44 which is obviously lower than the result obtained by the SVM method. Generally speaking, the methods taking into account the answer rank allow better results than the methods using the answer confidence score. Another interesting observation is that the interleaving methods obtained better results when not using the RITEL-P system while the SVM one obtained better results when using the three systems. We assume that these two systems, RITEL-S and RITEL-P are too similar to provide strict useful information, but that a ML-based approach is able to generalize such information.

In order to validate our choice of using only the first ten candidate answers, we did some more tests using 5, 10, 15 and 20 candidate answers. Table 4 shows the results obtained with the SVM model. We can see that is is better to consider 10 candidate answers. Beyond the first 10 candidate answers it is difficult to re-rank the correct answer without adding unsustainable noise. Moreover most of the correct answers are in the first ten candidates.

In order to validate the confidence score normalization, we did experiments with and without this normalization. Table 5 presents results which validate our choice.

To better understand how the fusion is made, we observed the repartition of the correct answers at the first rank and at the top five ranks according to the number of systems which extracted them (figure 3 and figure 4). We do this for the three best fusion approaches: the ML method with 3 systems, the hybrid method and the sum of the inverse of the ranks with two systems. As we can

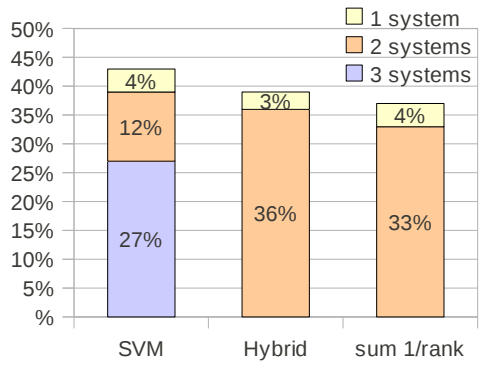

Figure 3: First rank

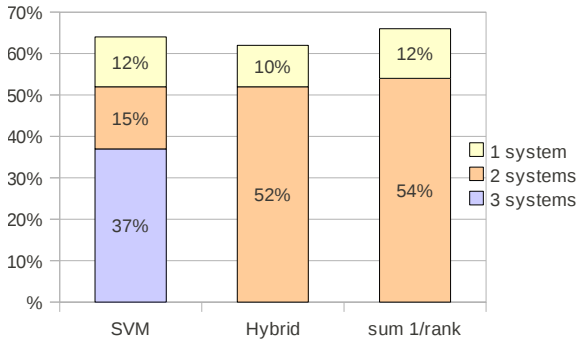

Figure 4: Top five ranks

see, in most of the cases, the three approaches often rank the correct answers found by all the systems. The best approach is the SVM-based one. It ranks 98 % of the correct answers given by the 3 systems in top 5 ranks. It also ranks better correct answers given by 2 systems (60% are ranked in the top 5 ranks versus about 48 % with the two other methods).

The rank-based method is globally reliable for selecting correct answers in the top 5 ranks. This behavior is consistent with the fact that our QA systems, when they found a correct answer, generally rank it in first positions.

Some correct answers given by only one system remain in the first position, and about 10% of them remain in the top 5 ranks and are not superseded by common wrong answers. However the major part of these correct single-system answers are discarded after the 5 first ranks (39% of them by the SVM method, 45% by the rank-based method and 53% by the hybrid method). In that case, a ML method is a better solution for deciding, however an improvement would be possible only if other features could be found for a better characterization of a correct answer, or maybe by enlarging the training base.

According to these results, we also can expect that with more QA systems, a fusion approach would be more effective.

8 Conclusion

Improving QA systems is a very difficult task, given the variability of the pairs (question / answering passages), the complexity of the processes and the variability of they performances. Thus, an improvement can be searched by the hybridization of different QA systems. We studied hybridization at different levels, internal combination of processes and merging of end-results. The first combination type did not proved to be useful, maybe because each system has its global coherence leading their modules to be more interdependent than expected. Thus it appears that combining different strategies is better realized with the combination of their end-results, specially when these strategies obtain good results. We proposed different combination methods, based on the confidence scores, the answer rank, that are adapted to the QA context, and a ML-method which considers more features for characterizing the answers. This last method obtains the better results, even if the simpler ones also show good results. The proposed methods can be applied to other QA systems, as the features used are generally provided by the systems.

References

R.M. Aceves-Pérez, M. Montes-y Gómez, L. Villaseñor-Pineda, and L.A. Ureña-López. 2008. Two approaches for multilingual question answering: Merging passages vs. merging answers. *International Journal of Computational Linguistics & Chinese Language Processing*, 13(1):27–40.

G. Bernard, S. Rosset, O. Galibert, E. Bilinski, and G. Adda. 2009. The LIMSI participation to the QAst 2009 track. In *Working Notes of CLEF 2009 Workshop*, Corfu, Greece, October.

G. De Chalendar, T. Dalmas, F. Elkateb-gara, O. Ferret, B. Grau, M. Hurault-plantet, G. Illouz, L. Monceaux, I. Robba, and A. Vilnat. The question answering system QALC at LIMSI: experiments in using Web and WordNet.

Arnaud Grappy and Brigitte Grau. 2010. Answer type validation in question answering systems. In *Adap-*

tivity, Personalization and Fusion of Heterogeneous Information, RIAO '10, pages 9–15.

Arnaud Grappy, Brigitte Grau, Mathieu-Henri Falco, Anne-Laure Ligozat, Isabelle Robba, and Anne Vilnat. 2011. Selecting answers to questions from web documents by a robust validation process. In *The 2011 IEEE/WIC/ACM International Conference on Web Intelligence*.

Arnaud Grappy. 2009. *Validation de rponses dans un systme de questions rponses*. Ph.D. thesis, Universit Paris Sud, Orsay.

Valentin Jijkoun and Maarten De Rijke. 2004. Answer Selection in a Multi-Stream Open Domain Question Answering System. In *Proceedings 26th European Conference on Information Retrieval (ECIR'04), volume 2997 of LNCS*, pages 99–111. Springer.

Thorsten Joachims. 2006. Training linear SVMs in linear time. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '06, pages 217–226, New York, NY, USA. ACM.

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.

Ludovic Quintard, Olivier Galibert, Gilles Adda, Brigitte Grau, Dominique Laurent, Veronique Moriceau, Sophie Rosset, Xavier Tannier, and Anne Vilnat. 2010. Question Answering on Web Data: The QA Evaluation in Quaero. In *LREC'10*, Valletta, Malta, May.

Joseph A. Shaw and Edward A. Fox. 1994. Combination of multiple searches. In *TREC-2*. NIST SPECIAL PUBLICATION SP.

Alberto Tellez-Valero, Manuel Montes Gomez, Luis Villasenor Pineda, and Anselmo Penas. 2010. Towards multi-stream question answering using answer validation. *Informatica*, 34(1):45–54.

Dave Toney, Sophie Rosset, Aurlien Max, Olivier Galibert, and Eric Bilinski. 2008. An Evaluation of Spoken and Textual Interaction in the RITEL Interactive Question Answering System. In European Language Resources Association (ELRA), editor, *Proceedings of the Sixth International Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, May.