University of Hagen at CLEF 2007: Answer Validation Exercise

Ingo Glöckner
Intelligent Information and Communication Systems (IICS),
FernUniversität in Hagen, 58084 Hagen, Germany
iglockner@web.de

Abstract

MAVE (Multinet-based Answer VErification) is an answer validation system based on deep linguistic processing and logical inference originally developed for AVE 2006. Robustness of the entailment check is obtained by embedding the theorem prover in a constraint relaxation loop. The system can also be used for answer selection, which is then guided by the joint evidence of all available text passages supporting a considered answer. Recent improvements of the basic MAVE system target at boosting answer selection. In order to profit from redundancy, the validation set is actively extended by additional supporting text passages. Three question-answering (QA) systems developed in our group were used to generate such candidates which were then filtered by methods for spotting answer-answer relationships. A novel technique called evidence reassignment (ERA) restructures the validation set by assigning each piece of evidence to all answers potentially supported by it. Further changes to MAVE comprise the integration of large lexical-semantic resources; backing the main logic-based features with overlap-based methods which are robust against parsing failure; implementation of two additional sanity checks (assessing the usefulness of an answer to a definition question, and compatibility of expected and actual answer types for factual questions); and finally the use of separate thresholds for selecting/rejecting the best answer and validating/rejecting the remaining choices, in order to capture the fact that many factual questions have a unique answer. Resources used include HaGenLex (a semantic-based lexicon for German developed in our group), GermaNet, and OpenThesaurus. Moreover MAVE has a very limited domain model. The results obtained confirm the proposed approach to answer selection, with f-measure in the 70 percent range, and 93 percent selection performance compared to an optimal selection strategy.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Information filtering, Selection process*; H.3.4 [**Information Storage and Retrieval**]: Systems and Software—*Performance evaluation (efficiency and effectiveness)*; I.2.4 [**Artificial Intelligence**]: Knowledge Representation Formalisms and Methods—*Predicate Logic, Semantic networks*; I.2.7 [**Artificial Intelligence**]: Natural Language Processing

General Terms

Experimentation, Measurement, Verification

Keywords

Logical Answer Validation, Answer Selection, Question Answering, Recognising Textual Entailment (RTE), Information Fusion, Robust Inference

1 System description

1.1 Introductory remarks

The first prototype of the MAVE answer validator [2] was developed for the Answer Validation Exercise 2006 and subsequently extended to support answer selection [3, 4]. This paper will focus on changes to the basic system and details omitted here will likely be found in one of these earlier publications.

1.2 Active enhancement of validation sets

The most important improvement to the MAVE system is the possibility to extract supporting text passages from the QA@CLEF news collection and the German Wikipedia. These passages are then added to the validation set as sources of additional evidence. This process of active enhancement can be described as follows. The AVE07 development set and the AVE07 test set comprise all 'original' validation items. (Depending on the context, both collections can be considered the 'validation set'). Each validation item can be represented by a quadruple (q, a, w, o) which relates to a certain question q and specifies an exact answer a and a witness text w extracted from the document corpus which is intended to justify the answer. The fourth component indicates if the validation item belongs to the original validation set (official validation item with $o = 1$), or if the item has later been added by active enhancement of the validation set (auxiliary item with $o = 0$). It is important to make this distinction between 'official' and 'auxiliary' validation items since only official validation items are admissible choices for answer selection. The auxiliary items need not support an answer which directly occurs in the test set. Generally speaking, such an item will also be useful if it refers only to a paraphrase or spelling variant of one of the answers of interest. Moreover a more detailed answer can be useful which includes one of the answers of interest as a special case. Two methods for leveraging such answer-answer relationships have been developed for MAVE.

The first method, already explained in [4], uses a simplification function σ in order to map the original answers to simplified answer keys by translating into lowercase, removing accents, omitting stopwords, omitting whitespace, etc. The resulting simplified answers are then used for grouping validation items which share the same cluster key, and evidence is aggregated for all validation items which belong to a given answer cluster. This method can be used for extending the original validation set. For that purpose, one needs QA systems which generate additional answer candidates and supporting text passages. Those generated answers which correspond to a cluster-key variant of one of the answers in the original validation set are then added as auxiliary validation items.

The second method, called evidence reassignment (ERA), makes it possible to combine answer validation and information fusion in a way consistent with non-local and non-monotonic phenomena in natural language (see [1] for an exposition of such phenomena).

To explain how ERA works, let us assume a simple method for hypothesizing answer-answer relationships which just checks the two answers for ordered character containment. Further suppose that $v = (\text{'}\textit{Who is Di Mambro?}\text{'}, \text{'}\textit{a prophet}\text{'}, w, 1)$ is the original validation item in the test set (actually, an item which should be rejected) and further suppose that $v' = (\text{'}\textit{Who is Di Mambro?}\text{'}, \text{'}\textit{a false prophet}\text{'}, w', 0)$ was found by one of the QA systems used for enhancing the validation set, where w' is a supporting text passage which mentions that Di Mambro is a false prophet. Then *'a false prophet'* would be wrongly considered as a specialization of *'a prophet'* by the simple inclusion-spotting method. A subsequent aggregation of evidence, treating the text passage mentioning that Di Mambro is a false prophet as evidence for Di Mambro being a prophet, would thus mean rewarding a false conclusion.

The easiest way to ensure sound results despite such non-monotonic and non-local phenomena is to simply reassign evidence by building new validation items. To this end, the ERA method builds a new validation item $v'' = (\text{'}\textit{Who is Di Mambro?}\text{'}, \text{'}\textit{a prophet}\text{'}, w', 0)$, while the extracted item v' (which does not refer to the exact answer of interest) will be discarded. The basic entailment check of the validation system will therefore check if the hypothesis *'Di Mambro is a prophet'* (rather than *'Di Mambro is a false prophet'*) is entailed by the witness text w'. Since this text only mentions that Di Mambro is a false prophet, this test will fail (at least in our system which does not treat *'false'* as an intersective adjective). Because the new supporting text is reassigned to the actual answer of interest (i.e. *'a prophet'*) and validated against this answer rather than the presumably including answer extracted by the inclusion spotting method (i.e. *'a* *false prophet'*), errors in the inclusion hypotheses will no longer spoil the results of logical validation and aggregation of evidence.

Four methods for recognizing inclusions and answer variants have been implemented for supporting the basic ERA approach: a) recognition of variants by clustering simplified answer strings; b) unsorted lexical overlap test; c) sequential lexical overlap, i.e. also considering the order of occurrence. This method is usually a strengthening of b., but might find additional matches in certain circumstances; d) logical inclusion as checked by an exact proof of the representation of the more general answer from the more specific answer given the background knowledge.

1.3 Processing of validation items

1.3.1 Syntactic-semantic analysis and coreference resolution

The WOCADI parser [5] is used for a deep linguistic analysis of the questions, answers, and supporting text passages. This process results in information about the tokens, lemmata, and word senses in the analyzed natural language expression. Moreover the parser constructs a semantic representation of the natural language input, which is expressed in the MultiNet formalism [6] (a variant of semantic networks specifically suited for natural-language processing). WOCADI handles intrasentential coreference resolution of pronouns and nominal anaphora and also gathers the necessary data for MAVE to perform a subsequent intersentential coreference resolution.

1.3.2 Postprocessing and synonym normalization

The post-processing of parsing results includes correction of a few known parsing problems and refinement of information on the facticity and genericity of objects and events mentioned in the text, which is also encoded in the MultiNet representation. The most important postprocessing step consists in the so-called *synonym normalization*, i.e. a list of synonyms and near-synonyms is used for replacing all lexical concept constants in the semantic representation with a canonical synset representative. This normalization eliminates the need of handling synonyms on the level of knowledge processing. To ensure this, the method must also be applied to the logical axioms, i.e. all lexical constants which occur in the axioms must be syno-normalized as well. The synonyms known to MAVE now cover 111,436 lexical constants and spelling variants which form 48,991 synonym classes (synsets). The core synonyms were taken from GermaNet and OpenThesaurus[1] and filtered by several quality criteria; they cover 24,619 word senses of non-compound words (simplicia). In German, nouns do not form multi-word units as in English; such groups rather combine into a single compound noun. Synsets for compounds were automatically computed from 920,429 compounds found in the QA@CLEF news corpus and the German Wikipedia by identifying compounds with synonymous parts. A total of 86,817 word senses for compound nouns were covered in this way. Some examples of discovered synonyms are *'Anrainerstaat'* vs. *'Nachbarland'* (neighboring country), *'Backsteinplastik'* vs. *'Ziegelsteinskulptur'* (brick sculpture) etc.

1.3.3 Hypothesis construction

The following statistics on parse qualities of our parser for the AVE06 test set demonstrates the difficulties of constructing textual hypotheses for a highly inflecting language like German. As remarked in [4], 93.3% of the questions in the AVE06 test set can be parsed with optimal quality, compared to only 64.7% of the (automatically constructed) textual hypotheses. Moreover parsing failed completely for 9.4% of the hypotheses, but only for 1.1% of the questions. These observations motivated the use of a different approach in MAVE, which generally avoids the construction of a textual hypothesis in favor of the direct construction of a *logical* hypothesis from the logical representations of the question and of the answer (see [4] for an example). Apart from avoiding parsing problems, the method has the additional advantage of being more precise. Consider a question like *'In which city is the Walk of Fame?'* and the answer *'USA'*. Constructing a textual hypothesis which expresses the full information given by question and answer can be difficult in such a case (e.g., *'The Walk of Fame is in the city (of) USA'*.) A simpler hypothesis like

[1] http://www.openthesaurus.de/

'The Walk of Fame is in USA', however, would drop the important information that the question asks for a city while the answer wrongly specifies a country. This kind of constraints is easily preserved if the representations of question and answer are combined into a hypothesis representation on the logical level.

1.3.4 Logic-based entailment test

In order to achieve a robust entailment test, the theorem prover of MAVE is embedded in a relaxation loop which drops part of the hypothesis literals when a proof of the current hypothesis fragment fails or can not been found within the time limit. By subsequently removing 'critical' literals from the query, this process always finds a (possibly empty) query fragment provable from the assumed knowledge. The number of skipped query literals not contained in the provable fragment is used as a numeric indicator `synth-failed-literals` of (non-)entailment strength which is robust against slight errors in the logical representation of hypothesis and supporting text, but also against a few gaps in knowledge modeling.

Resources used by the prover include: 8,464 relations covering nominalizations of verbs; 947 subordination relationships (most of them relating female forms of occupation terms (like *'Pilotin'* – female pilot) to the male form of the occupation term (like *'Pilot'*), which for purposes of answer-validation can be considered as the gender-neutral form rather than refering e.g. to male pilots. Finally there are 76 entailment relationships between adjectives kept from the first version of MAVE. Apart from these lexical-semantic relations, the background knowledge of MAVE comprises 109 implicative rules. These rules correspond to those already used in AVE06, with some restructuring due to changes in the lexicon.

1.3.5 Fallback matching method

The logic-based test depends on semantic representations which are not available when the parser fails. To gain robustness against this case, an alternative matching-based method was implemented which determines the overlap between the concept and numerals contained in question and answer, and the concepts and numerals which occur in the supporting witness text. The number of concepts and numbers extracted from question and answer which cannot be matched with concepts and numbers in the representation of the snippet is stored in the indicator `match-err-count`. It should be remarked that the fallback matcher eliminates stopwords before attempting a match. Moreover the matching rules for numbers are rather liberal. Thus *'April'* matches with *'4'* and vice versa. *'1 828'* matches with *'1'*, with *'828'*, and with *'1.828'* (assuming German number syntax). Moreover, English notation for numbers is accepted when it does not create ambiguity, e.g. *'1.3'* is correctly interpreted though it would normally be written as *'1,3'* in German. The simple matching method can also leverage lexical-semantic relations which serve for generating further options for the overlap check. It integrates all lexical-semantic resources also used by the prover, including the synonyms, but also some further resources: a larger list of 27,814 nominalization relationships, which also contains ambiguous cases[2] as well as a list of 15,052 nominalizations of adjectives, like *'Kleinheit'* (smallness) derived from *'klein'* (small), which are not yet part of the logic-based subsystem.

1.3.6 Determining entailment error levels

The validation and selection decision of MAVE is based on the computation of error levels. These include the logic-based `synth-failed-literals` count, the overlap-based `match-err-count` obtained by matching lexical concepts and numbers, and the following additional indicators for non-entailment.

- `synth-proof-facticity`: The facticity score of the proof, determined by collecting all involved entities and checking their facticity. A score of 2 signals that the proof involved a non-existing entity or non-real event; a score of 1 signals that the proof involved a hypothetical entity with unknown facticity (this happens e.g. with certain modal embeddings).

- `synth-dropped-names`: The number of full person names (i.e. specifying first name and last name) mentioned in the question or answer string for which only the last name occurs in the witness text, but not the first name. In this case, the first name is eliminated from the logical hy-

[2] e.g. *anspannen.1.1 – 'put (a cart) before a horse)'* vs. *anspannen.1.2 – 'to strain'*, with only one available nominalization *anspannung.1.1* at the moment.

pothesis so that it does not prevent a logical proof, and the error is reported by incrementing the `synth-dropped-names` count.

- `synth-name-conflicts`: The number of person names in the logical hypothesis such that the last name of the queried person coincides with the last name of a person mentioned in the witness text, but the first name differs. This means that the text is likely about a different person which happens to share the last name with the person mentioned in the query, an indication of non-entailment.
- `synth-missing-constraints`: The number of numerals in the question and answer string which were lost in the semantic representation due to parsing errors.
- `synth-nonbound-focus`: This binary feature signals if the focus variable (i.e. the variable in the logical hypothesis which represents the queried information) was actually bound during the relaxation proof. If all hypothesis literals in which the focus variable occurs are skipped, the result is 1, which strongly indicates that an entailment relationship does not hold.
- `synth-nonbound-vars`: This novel feature counts the number of variables of the logical hypothesis which were not bound to entities in the representation of the witness text by the relaxation proof. Since none of the literals connecting such a variable to the remaining query could be proved, this indicates a serious problem concerning entailment.

When question, answer and witness text can be parsed, then the logic-based error level `synth-err-count` is defined as follows:

$$\begin{aligned}&\texttt{synth-err-count} = \texttt{synth-name-conflicts} + \texttt{synth-dropped-names}\\&\quad + \min(\texttt{synth-failed-literals} + \texttt{synth-missing-constraints}\\&\quad + \texttt{synth-proof-facticity} + 2\,\texttt{synth-nonbound-focus}\\&\quad + \texttt{synth-nonbound-vars}, 3) + \texttt{match-err-count}\,.\end{aligned}$$

Otherwise `synth-err-count` is undefined and `match-err-count` will serve as a fallback replacement for the logical entailment test which is robust against parsing failure.

1.3.7 Computing error probabilities for individual validation items

In order make the logic-based `synth-err-count` levels and the fallback `match-err-count` levels comparable, we abstract from error levels by assigning corresponding error probabilities. Thus, what we are really interested in is not the concrete error count but rather the probability that the validation item is correct (or wrong) given this error count. For convenience, we assume that the failure probability can be expressed in the form $\min(\alpha e^{-\beta\ell}, 1)$ where $\alpha, \beta \in \mathbb{R}_0^+$ and ℓ is the error level. Using `synth-err-count` for ℓ, we then obtain the probability estimate `synth-err-prob` describing the failure probability of the considered validation item judging from its witness text in isolation. Similarly, using `match-err-count` for ℓ (with a different error model of α', β') results in a fallback failure probability `match-err-prob`. A method for extracting the parameters α, β from the `synth-err-count` and `match-err-count` levels in an annotated validation set is described in [4]. There are many conceivable ways how these failure probabilities might be combined but for the time being MAVE makes an optimistic choice by selecting the minimum of the available values, i.e. $\texttt{err-prob} = \min(\texttt{synth-err-prob}, \texttt{match-err-prob})$ if `synth-err-count` is defined, and $\texttt{err-prob} = \texttt{match-err-prob}$ otherwise.

1.3.8 Aggregation of evidence for clusters of validation items

In the next step, aggregation is used in order to compute the combined justification of an answer judging from all validation items which support the answer or a variant of it. A simplification function σ can be used for assigning a simplified answer string or 'cluster key' to each answer (see [4]), and validation items which share the same answer key are then grouped together. However, some configurations of MAVE also use the identity for σ; in this case aggregation is only possible for validation items which support identical answers. Generally speaking, the answer is logically justified if it is logically justified from at least one witness text, i.e. it is false only if all validation items in the cluster are false. Assuming independence of the validation items, we can then express the combined failure probability as $\texttt{mult-err-prob} = \prod_c \texttt{err-prob}_c$, where c ranges over all validation items which share the answer key with the validation item of interest. However, independence is likely a wrong assumption in this context, and tends to underestimate the actual error. Therefore a less optimistic combination like $\texttt{min-err-prob} = \min_c \texttt{err-prob}_c$ makes sense. Moreover, since the validation item of interest must eventually be presented to the user as a convincing justification of the answer, its immediate failure probability `err-prob` should also be considered. Since the proper way of combining these probability estimates is not known, a simple average $\texttt{combined-err-prob} = (\texttt{err-prob} + \texttt{mult-err-prob} + \texttt{min-err-prob})/3$ was used.

1.3.9 Judging witness quality

The basic quality of the witness is computed from the correctness probability $(1-\texttt{combined-err-prob})$ and a heuristic quality factor describing non-aggregable preferences on 'good' supporting text passages like high parsing quality, conciseness etc. Using the same set of heuristic features and weights as in [4], we let

$$\begin{aligned}\texttt{wn-heuristic-quality} = {} & c(0.2, \texttt{wn-occurrences}) \cdot c(0.2, \texttt{wn-parse-quality}) \\ & \cdot c(0.2, \texttt{producer-score}) \cdot c(0.1, \texttt{wn-num-sentences}) \cdot c(0.1, \texttt{wn-num-chars}) \\ & \cdot c(0.3, \texttt{wn-special-chars}) \cdot c(0.2, \texttt{wn-relativizing-words}) \cdot c(0.2, \texttt{wn-qn-focusing}).\end{aligned}$$

The weighting function $c(w, x) = 1 - w + wx$ adjusts the relative impact of each criterion on the final quality score. The aggregated logical justification and the heuristic criteria are then combined into the basic witness score $\texttt{wn-quality} = \texttt{wn-heuristic-quality} \cdot (1 - \texttt{combined-err-prob})$.

1.3.10 False positive tests and heuristic answer quality

Capturing witness quality is not sufficient for answer validation since there are cases when an answer must be rejected even if it is perfectly justified from the logical perspective. A typical example are trivial answers to questions *'Who is X?'* which simply repeat X. MAVE therefore supports a number of false-positive checks. These criteria do not depend on the supporting text passages and are therefore not aggregable.

- `aw-not-trivial`: eliminates trivial answers. The feature is now defined as a disjunction of a logic-based triviality test which checks if the answer be proved from the question, and a matching-based overlap test which checks if all lexical constants (or alternatives linked by lexical-semantic relations) and numerals that occur in the answer are also found in the question.

- `aw-significant-def`: This test for informative answers to a definition question is based on the observation that isolated nomina agentis (like *'the founder'* compared to *'the founder of Microsoft'*) or isolated role terms (like *'the foreign minister'* rather than *'the German foreign minister'*, or *'the brother'* rather than *'Wladimir Klitschko's brother'*) are hardly suitable for answering *'Who is X?'* style definition questions. MAVE uses a list of 2,856 nomina agentis (i.e. nouns denoting an agent of a verb) and relational nouns in order to recognize such cases. If the considered question is a definition question, then the lexical constants from this list will be deleted from the list of question concepts and the overlap-based triviality test is applied to the reduced list of question concepts. Thus an answer is penalized if it contains only nomina agentis and relational terms and no additional content beyond that already mentioned in the question.

- `aw-not-circular`: eliminates circular answers. The circularity test of [4] was changed such that it only applies to constituent-level answers (not to full-sentence answers which repeat part of the question). Thus, *'The inventor of the modern car is Carl Benz'* is a legal answer to the question *'Who is the inventor of the car?'*, while *'the inventor of the modern car'* is not.

- `aw-eat-fat-compat`: Checks if the expected answer type of the question and the found answer type of the answer are compatible. In order to preserve recall levels, the filter was designed to reject only answers which are *clearly wrong* with respect to the answer type criterion while unclear cases are not eliminated. The filter is implemented by a hand-coded decision tree abstracted from known annotations of QA@CLEF questions and answers for the years 2004–2006.

Apart from these sanity checks, there are also heuristic preferences on ‘good’ answers, encoded by the features `aw-incompleteness` (completeness of the answer as a function of its length), `aw-overlength` (penalty for overlong answers), and `aw-parse-quality` (the parse quality of the answer), see [4].

The false-positive tests and soft preferences are combined as follows,

$$\begin{aligned}\texttt{aw-heuristic-quality} &= c(0.1, \texttt{aw-incompleteness}) \cdot c(0.1, \texttt{aw-overlength})\\ &\cdot c(0.2, \texttt{aw-parse-quality}) \cdot c(1.0, \texttt{aw-not-trivial}) \cdot c(0.5, \texttt{aw-significant-def})\\ &\cdot c(0.6, \texttt{aw-not-circular}) \cdot c(1.0, \texttt{aw-eat-fat-compat})\,.\end{aligned}$$

Note that variants of the same answer (i.e. answers with the same cluster key and thus the same ‘content’) can differ with respect to parse quality. This means that the answer quality factor relates to exact answers.

1.3.11 Computing the validation score

The final validation score used for selection and validation decisions is given by

$$\texttt{validation-score} = \texttt{aw-heuristic-quality} \cdot (\texttt{wn-quality} + \texttt{bonus-wn-quality})/2\,,$$

where `wn-quality` refers to the considered validation item and `bonus-wn-quality` is the maximal `wn-quality` achieved by any original or generated validation item in the enhanced validation set which supports the given exact answer (when using the ERA method) or a cluster-key variant of the answer (when using cluster variants). Using the bonus term makes sense because the number of ‘unsupported’ cases (i.e. correct answer though not entailed by the snippet) can almost be neglected compared to the number of wrong answers. Therefore correctness of an answer, as judged from the existence of a very good supporting text passage in the original validation set or actively found in the document collection, is also a strong indicator for validity of the considered validation item.

1.3.12 Applying thresholds for the selection and validation decision

The `validation-score` is used as the metric for selecting the best answer in the AVE07 validation set (candidate for SELECTED or REJECTED) and also as the basis for validation/rejection of the remaining answers (candidates for VALIDATED or REJECTED). For factual questions, there is often only one correct answer (e.g. *‘In welchem Alter starb Elvis Presley?’*, En: *‘At which age did Elvis Presley die?’*). As soon as the best answer has been found, alternative answers should only be accepted if they are ‘really convincing’. To achieve this, MAVE uses separate thresholds $\texttt{f-sel-thresh} \leq \texttt{f-val-thresh}$ in its decision rules for accepting/rejecting the best answer candidate and the remaining alternatives. The best validation item for a given question (i.e. the item in the test set which maximizes `validation-score`) is predicted SELECTED if $\texttt{validation-score} \geq \texttt{f-sel-thresh}$, and REJECTED otherwise. If the best official witness was SELECTED and the `validation-score` of a non-best answer exceeds `f-val-thresh`, then the item is predicted VALIDATED, otherwise it is predicted REJECTED.

The thresholds `f-val-thresh` and `f-sel-thresh` are determined from the development set. Two ways of optimizing the thresholds were considered. If the system targets at maximizing the f-measure (signaled by letter ‘F’ in the later experiments), then thresholds are optimized by considering all levels of `validation-scores` in the development set and selecting the two thresholds `f-sel-thresh` for the best validation item and $\texttt{f-val-thresh} \geq \texttt{f-sel-thresh}$ for the non-best items which maximize the f-measure over the development set. If emphasis is placed on optimizing successful selection, however (signaled by ‘Q’ for ‘qa-accuracy’ in later runs), then we let $\texttt{f-sel-thresh} = 0$, which forces the best answer to a question to be chosen in every case. Using $\texttt{f-sel-thresh} = 0$ for selection, `f-val-thresh` is then optimized for non-best answers such as to maximize the overall f-measure on the development set.

1.3.13 Assigning confidence scores

For computing confidence scores, MAVE first extracts two more thresholds `p-sel-thresh` (for the best validation item) and `p-val-thresh` (for the remaining alternatives) which are chosen to maximize the likelihood of correct decision (i.e. the 'accuracy') on the assumed development set. Every validation item is associated a 'signature' $(\texttt{tv}, \texttt{bv}, \texttt{tpa}, \texttt{bpa})$ where $\texttt{tv} =$ SELECTED|VALIDATED|REJECTED ('this value') is the predicted value for the validation item; $\texttt{bv} =$ SELECTED|REJECTED ('best value') is the predicted value of the best 'official' validation item for the considered question (i.e. item in the original AVE07 test set); the value is obtained by applying the `f-sel-thresh` to the best item as explained above; $\texttt{tpa} =$ YES|NO ('this p-accept') is the acceptance decision obtained by applying `p-sel-thresh` or `p-val-thresh` to the validation score of the current validation item (depending on whether the item is the best one or supports an potential alternative answer); $\texttt{bpa} =$ YES|NO ('best p-accept') is the acceptance decision obtained by applying `p-sel-thresh` to the best witness for the question.

Now suppose that every validation item in the AVE07 development set has been associated a signature $(\texttt{tv}, \texttt{bv}, \texttt{tpa}, \texttt{bpa})$. Items which share the same signature are grouped into classes. For each class, MAVE determines the empirical correctness probability $p(c)$, i.e. the relative frequency that the item is either predicted SELECTED/VALIDATED and annotated VALIDATED, or predicted REJECTED and annotated REJECTED, considering only validation items which are not annotated UNKNOWN. After that, classes are arranged in ascending order of probability, i.e. $c_1, c_2, \ldots, c_m$ such that $p(c_1) \leq \cdots \leq p(c_m)$. For each class, the 'radius' $r(c)$ is determined as follows:

$$r(c) = \begin{cases} (p(c_2) - p(c_1))/2 & : \quad c = c_1 \\ \min(p(c_i) - p(c_{i-1}), p(c_{i+1}) - p(c_i))/2 & : \quad c \neq c_1, c \neq c_m \\ (p(c_m) - p(c_{m-1}))/2 & : \quad c = c_m \end{cases}$$

Now let $n(c_i)$ be the number of elements from the test set mapped to class c_i. If $\texttt{tv} =$ SELECTED or $\texttt{tv} =$ VALIDATED, then a validation item with a high validation score is more reliable. We use this to improve the coarse ranking given by the empiricial correctness probabiliy of the class. Hence suppose that the elements are arranged in increasing order of validation score, i.e. $e_1, \ldots, e_{n(c_i)}$ with $\texttt{validation-score}(e_k) \leq \texttt{validation-score}(e_\ell)$ for $k \leq \ell$. We then associate with e_j the confidence score $C(e_j) = p(c_i) - r(c_i) + 2\,\frac{j-1}{n(c_i)-1} r(c_i)$ or in the case of a singleton e_1, $C(e_j) = p(c_i)$. Thus, answers with the highest `validation-score` are also judged the most reliable within their signature class. If $\texttt{tv} =$ REJECTED, by contrast, then a witness with a low validation score is a more reliable negative example. Therefore the elements $e_1, \ldots, e_{n(c_i)}$ are arranged in decreasing order of validation score in this case, so that the elements with the highest validation scores will be assigned the least confidence.

2 Evaluation

This section describes the two configurations of MAVE used for generating the runs submitted to AVE07. It then discusses the performance scores achieved by these configurations and presents a series of ablation studies which elucidate the effect of the main system components.

The following naming conventions are used for describing configurations of MAVE. The letter 'C' indicates the use of simplified answer strings (also dubbed 'cluster keys') for clustering answers. In this case, active enhancement of the validation set will add all validation items found by the external QA systems which support an exact answer in the original test set or a cluster-key variant of it. Use of the exact answer string for grouping answers is marked by 'N' for 'non-clustering' (when ERA is not used) or 'E', when ERA is used for enhancing the test set. Use of the plain validation items without active enhancement of validation sets is marked by 'P'. The letter 'F' indicates optimization of thresholds for maximizing f-measure on the development set while 'Q' indicates optimization of the thresholds for maximizing qa-accuracy and selection performance on the development set.

Following these conventions, the first run submitted to the AVE07 is named CF, i.e. it uses the cluster-key for grouping answers and optimizes thresholds for f-measure. Parameter estimation is based on the development set enhanced by cluster key variants which were filtered from earlier runs of the InSicht, QAP and MIRA QA systems on the CLEF questions [4]. In this way, the 504 validation items in the original AVE07 development set were extended by another 907 supporting items contributed by our QA systems. Similarly, the original AVE07 test set with its 282 validation items was extended by 1,386 additional supporting items filtered from candidate lists of InSicht, QAP and MIRA.

Table 1: Results of the two runs submitted to the AVE07

model	f-meas	f-gain	prec	recall	qa-acc	sel-rate
CF/Run1	0.72	0.79	0.61	0.90	0.48	0.89
EQ/Run2	0.68	0.69	0.54	0.94	0.50	0.93

Table 2: Reference results for the ablation experiments

model	f-meas	f-gain	prec	recall	qa-acc	sel-rate
CF*	0.73	0.81	0.62	0.90	0.49	0.90
CQ*	0.70	0.74	0.56	0.94	0.50	0.93
EF*	0.73	0.82	0.62	0.91	0.50	0.92
EQ*	0.69	0.71	0.55	0.93	0.50	0.93

The second submitted run (EQ) uses the ERA method both for enhancing the development set, which affects the determined error model and thresholds, and for extending the test set. Thresholds were optimized for qa-accuracy in order to achieve a high percentage of correct selections. Since ERA handles answer variants by reassignment of validation items, the actual aggregation (i.e. computation of `mult-err-prob` and `min-err-prob`) must be limited to validation items which support the same exact answer. The same applies to determining `bonus-wn-quality` where support for identical answers is required. In order to enhance the validation set using the ERA method, the four methods for spotting answer-answer relationships were applied to the total of 12,837 answer candidates with 30,432 supporting text passages generated by the InSicht, QAP and MIRA QA systems for the considered questions. This process resulted in 2,320 supporting items being added to the original test set.

The results obtained for the two submitted runs are shown in Table 1. Here and in the following tables, ‘f-meas’ denotes the f-measure, ‘f-gain’ is the increase in f-measure compared to the 100% YES baseline [7], ‘prec’ is precision, ‘qa-acc’ is qa-accuracy, i.e. the number of correct selections divided by the number of all questions, and ‘sel-rate ’ or selection rate is the relative performance of the system compared to optimal selection, i.e. the number of correct selections divided by the number of questions with at least one correct answer in the annotated test set. The results obtained, in particular the 93 percent selection rate in the second run, confirm the suitability of the proposed method for boosting answer selection performance. Moreover the system can also handle the basic validation task, as shown by the achieved f-measure scores around 70 percent.

In order to find out which of the proposed techniques contribute most to these results, a series of ablation experiments will now be presented. However, it makes more sense to relate these experiments to the current version of MAVE, which differs from the AVE07 system in two respects. First, a bug in the answer-type filter has been fixed in the meantime, which resulted in misclassification of COUNT/MEASURE questions. The second change is related to the ERA method. It was noticed that evidence reassignment might result in a disbalance of training examples when applied to the development set: In the extreme case, the method might add a large number of additional supporting items for only one of the answers in the validation set. To avoid the risk of disbalance, application of ERA has now been restricted to the test set, while the error model (i.e. the α, β parameters for determining `synth-err-prob` and `match-err-prob`) and the thresholds are determined from the original AVE07 development set enhanced by all validation items from the InSicht, QAP and MIRA runs which literally support the considered answer.[3]

For the current version of MAVE, we then obtain the reference results shown in Table 2. Fixing the error in the answer-type filter and changing the parameter selection for the ERA runs has (slighly) improved overall performance, resulting in better qa-accuracy and selection rate for the f-measure oriented run, and improved f-measure in the qa-accuracy oriented run. In the following ablation studies, the reference con-

[3] Another solution would be applying the ERA method to the development set as well, but avoiding disbalance by some *weighting* of examples which ensures that no single validation item dominates parameter estimation.

Table 3: Results of MAVE without enhancement of validation sets

model	f-meas	f-gain	prec	recall	qa-acc	sel-rate
PNF	0.68	0.67	0.61	0.76	0.41	0.75
PCF	0.68	0.67	0.61	0.76	0.41	0.75
PCF+	0.68	0.68	0.60	0.78	0.42	0.77
PNQ	0.66	0.63	0.53	0.88	0.48	0.89
PCQ	0.66	0.63	0.53	0.88	0.48	0.89
PCQ+	0.67	0.66	0.54	0.90	0.48	0.89

Table 4: Results of MAVE using a joint threshold for selection and validation

model	f-meas	f-gain	prec	recall	qa-acc	sel-rate
EJF	0.68	0.68	0.55	0.88	0.46	0.85
EJQ	0.45	0.11	0.29	0.97	0.50	0.93

figurations shown in the table will be varied by switching off one or more functional components of MAVE and observing the changes of obtained results.

The first experiment is concerned with the effect of active validation, i.e. of enhancing the test set or training set. In the ablation runs shown in Table 3, the 'plain' development set and validation set was used, i.e. no additional supporting text passages were added. The models labeled 'N' do not use cluster-key simplification, i.e. only identical answers are grouped, which eliminates any opportunity for aggregation because the AVE07 development set and test set are free of answer redundancy. Compared to the EF* reference, the PNF run shows 5 percent difference in the f-measure and even 17 percent points loss in selection rate. The selection-oriented PNQ run loses 3 percent of f-measure and 4 percent of selection rate. These findings reveal a strong positive effect of active validation. It was also tried to use the cluster-key to group answer variants for aggregation (see runs labeled 'C'). This had no effect on results, however, since answer variants which share the same cluster key are not present in the AVE test set.

Finally it was tested if the decrease in validation and selection performence results from the smaller size of the development set (compared to the actively enhanced variant), which might result in less reliable probability estimates. The runs PCF+ and PCQ+ therefore used the development corpus enhanced by cluster-key variants for estimating parameters, while using the original test set without enhancements. As witnessed e.g. by the selection rate of PCF+, which is still very low, the smaller size of the development set when not using active enhancement is not responsible for the observed drop of selection rate. Thus active validation mainly profits from the additional supporting text passages in the enhanced test set and not so much from the more reliable parameter estimates due to increased size of the development collection.

Next let us consider the use of separate thresholds for selection/rejection of the best validation item and validation/rejection of alternative answers; see Table 4. Both runs are based on the ERA method. EJF uses a joint threshold for selection and validation chosen to maximize f-measure on the development set, while EJQ uses a joint threshold chosen to maximize qa-accuracy on the development set. Compared to the EF* reference, we notice a loss of f-measure by 5 percent and a loss of selection rate by 7 percent in the EJF case. The EJQ variant which optimizes qa-accuracy still shows a selection rate of 0.93, but suffers a drastic loss of f-measure by 24 percent points compared to the EQ* reference. Thus, using separate thresholds for selecting the best answer and for accepting alternative answers pays off in the AVE context.

The next series of experiments is concerned with the effect of the various sanity checks used by MAVE. The results shown in Table 5 should be compared to the EF* and EQ* reference. The following naming scheme is used: 'A' denotes deactivation of the answer-type filter (`aw-eat-fat-compat`), 'T' means deactivation of the trivial answer filter (`aw-not-trivial`), 'S' means deactivation of the significant answer filter for definition questions (`aw-significant-def`), 'Z' means deactivation of the circular answer filter (`aw-not-circular`), and * means deactivation of all of the above sanity checks. Though

Table 5: Results of MAVE without false-positive tests

model	f-meas	f-gain	prec	recall	qa-acc	sel-rate
EAF	0.73	0.79	0.60	0.91	0.49	0.90
ETF	0.71	0.74	0.58	0.90	0.49	0.90
ESF	0.73	0.80	0.61	0.91	0.50	0.92
ESTF	0.69	0.70	0.56	0.90	0.49	0.90
ECF	0.73	0.82	0.62	0.91	0.50	0.92
E*F	0.68	0.68	0.55	0.90	0.48	0.89
EAQ	0.69	0.69	0.54	0.93	0.50	0.92
ETQ	0.67	0.66	0.54	0.91	0.50	0.92
ESQ	0.69	0.70	0.55	0.93	0.50	0.93
ESTQ	0.66	0.63	0.52	0.91	0.50	0.92
EZQ	0.69	0.71	0.55	0.93	0.50	0.93
E*Q	0.65	0.61	0.51	0.91	0.49	0.90

Table 6: Results of MAVE without using logic-based features

model	f-meas	f-gain	prec	recall	qa-acc	sel-rate
LCF	0.72	0.78	0.59	0.93	0.48	0.89
LEF	0.72	0.79	0.59	0.94	0.49	0.90
KCF	0.56	0.39	0.44	0.78	0.42	0.77
LCQ	0.68	0.69	0.53	0.96	0.50	0.92
LEQ	0.68	0.68	0.53	0.96	0.50	0.92
KCQ	0.55	0.37	0.43	0.79	0.42	0.79

none of these filters applies very often, they all show a consistent positive effect on the performance of MAVE. When deactivating all filters, f-measure drops by 5 percent points comparing E*F to EF* (or 4 percent points comparing E*Q to EQ*). Selection rate too drops by 3 percent compared to the reference. Due to the overlap of the 'T' and 'S' filters, simultaneous deactivation of both filters was also tested. As shown by the ESTF run, these filters contribute up to 4 percent points to the achieved f-measure compared to EF*, and 3 percent in the EQ* case. This means that the two variants of triviality filtering contribute most to the overall effect of sanity checking.

Finally let us assess the effect of using a theorem prover and logical rules compared to the fallback matching method, and the effect of the lexical-semantic relations compared to using no background knowledge at all.

The introduction of the fallback matching method is justified by the following statistics: due to parsing failures, the proof-based validation feature `synth-err-count` was defined only for 168 of the 282 validation items in the original AVE07 test set; for 1,223 of the 1,668 validation items using cluster-key variants, and for 1,840 of the 2,602 validation items in the ERA-enhanced test set. The fallback method thus helps to better exploit the available evidence. In order to assess the quality of the fallback method, a number of experiments was conducted with the theorem prover switched off. The naming scheme for the ablation runs shown in Table 6 is as follows: 'L' indicates that all logic-based features are deactivated, while 'K' disables not only the prover, but also the use of any lexical-semantic relations in the fallback matching method. Notice that disabling the prover affects several features which involve the use of logic: the `synth-err-count` (which must be totally disabled), the triviality check (which is then determined by the fallback matching method only), the logic-based circularity test (which is not backed by a matching method and thus entirely switched off). Moreover the 'E' runs based on ERA use only the answer-answer analyzers which do not involve the use of logic (clustering, matching and sequential matching) but disable the logical inclusion test. The 'C' runs group answers by cluster-key as usual.

Table 7: Additional tests of MAVE with the logical prover switched off

model	f-meas	f-gain	prec	recall	qa-acc	sel-rate
LCAF	0.71	0.76	0.58	0.93	0.47	0.87
LEAF	0.72	0.77	0.58	0.94	0.48	0.89
PLCF	0.65	0.62	0.57	0.76	0.40	0.74
PLNF	0.65	0.62	0.57	0.76	0.40	0.74
LCAQ	0.68	0.67	0.52	0.96	0.49	0.90
LEAQ	0.67	0.66	0.52	0.96	0.49	0.90
PLCQ	0.64	0.58	0.51	0.87	0.46	0.85
PLNQ	0.64	0.58	0.51	0.87	0.46	0.85

Surprisingly, the performance of the f-measure oriented LEF run shows only a minimal decrease compared to the f-measure (1 percent point loss) and selection rate (2 percent points loss) of the EF* reference. For LEQ, the picture is similar, with 1 percent point loss compared to EQ* both for f-measure and selection rate. The effect of robust logical inference is positive but very small.

The picture changes when disabling not only the use of the prover, but also the use of lexical-semantic relations for fallback matching, as shown by KCF vs. CF* with a 17 percent difference in f-measure and 13 percent difference in the selection rate. Comparing KCQ to CQ*, f-measure decreases by 15 percent points while selection-rate loses 14 percent points. These findings demonstrate that lexical-semantic knowledge now makes a very strong contribution to the performance of MAVE. In particular, the massive extension of lexical-semantic knowledge compared to the first prototype pays off (no clear effect was found in earlier experiments based on a small repository of lexical-semantic relations [3]).

A few additional tests were made in order to better understand the success of the simple overlap-based method. Questions to be clarified include: Does redundancy explain the results of the fallback method? Does it profit too much from the structure-sensitive answer-type filter which involves some use of the prover for graph matching (but no background knowledge or lexical-semantic knowledge)?

Table 7 uses the letters 'L' (no logic), 'P' (no active enhancement of development and test corpus), 'C' (group answers by cluster keys), 'E' (evidence reassignment, in this case without logical inclusion recognizer), 'N' (no clustering except for exact answer strings), and 'A' (no answer-type filter); 'Q' and 'F' have the usual meaning. Comparing LCAF to LCF, LEAF to LEF, LCAQ to LCQ and LEAQ to LEQ, we notice that deactivating the answer-type filter has a minor effect on the achieved f-measure (at most 1 percent points loss) and selection rate (at most 2 percent points loss). This means that the success of the simple matching method is not explained by the use of a structure-sensitive answer-type filter. Concerning the effect of redundancy, we notice a difference of 7 percent in f-measure comparing PLCF (no redundancy) with LCF (which uses an enhanced validation set). Selection rate even drops by 15 percent points. A similar pattern becomes visible comparing PLNF and LEF, with a 7 percent points loss in f-measure and a 16 percent loss in selection rate when there is no redundancy. There is also a clear effect when optimizing thresholds for qa-accuracy. Thus, the success of the fallback matching method relies strongly on redundancy created by active enhancement of validation sets. However, the same dependency was observed with the combined method of using logic-based and matching-based information, cf. Table 3. This means that the results of MAVE in its current configuration are mainly due to the active enhancement of the validation sets.

Turning to the ERA enhancement method, we consider the effect of different methods for spotting answer-answer relationships as the basis for evidence reassignment. Four methods were implemented for MAVE: variant answers which share the same cluster key (labeled 'clust' below); matching of an answer with a given one using the fallback matching method (labeled 'match' below); sequential matching of the tested answer with a given one (labeled 'seqm' below) where the word senses and numeric expressions must find matching expressions in the proper sequental order; and finally the logical inclusion test (labeled 'incl'). MAVE normally applies all four methods for spotting relationships between answers. In order to assess the relative merits of each technique, experiments using just one of these methods were conducted.

Table 8: Test of relationship-spotting methods for ERA

model	f-meas	f-gain	prec	recall	qa-acc	sel-rate
clust-F	0.73	0.81	0.62	0.90	0.49	0.90
match-F	0.73	0.80	0.61	0.90	0.49	0.90
seqm-F	0.73	0.80	0.61	0.90	0.49	0.90
incl-F	0.73	0.80	0.62	0.88	0.48	0.89
clust-Q	0.70	0.72	0.56	0.93	0.50	0.93
match-Q	0.69	0.71	0.55	0.93	0.50	0.93
seqm-Q	0.69	0.71	0.55	0.93	0.50	0.93
incl-Q	0.70	0.73	0.56	0.93	0.50	0.93

As shown by these experiments, results for the four methods are very similar. Thus for the AVE07 data, shallow and deep methods for recognizing answer-answer relationships perform about the same. This again illustrates that axioms and inference are either not needed yet (due to simplicity of the test cases or due to redundancy created by active enhancement of the validation set); or that axioms are needed but not available at this time so that their potential can not yet be fully demonstrated.

3 Conclusion

The proposed changes to the basic MAVE system have boosted f-measure beyond 70 percent and selection rate even into the 90 percent range. As shown by the ablation studies, these improvements can be attributed to the following methods: a) active enhancement of the validation set by additional supporting text passages generated by external QA systems; b) clustering of validation items for answer variants or use of the more flexible ERA method which offers a sound solution for leveraging inclusion of answers in an answer validation framework. In order to treat non-local and non-monotonic phenomena consistently, ERA handles inclusions of answers by reassigning evidence to all compatible validation items; c) integration of a large repository of lexical-semantic relations; d) use of various sanity checks for eliminating false positives; e) use of separate thresholds for the selection of the best answer and for acceptance/rejection of the remaining alternatives; and finally f) provision of robust fallback solutions for the main logic-based features. The excellent results of these fallback methods which do not involve any theorem proving or structural matching (only one percent loss in performance when the prover is switched off) are somewhat irritating. Obviously the AVE task which involves only a few *end results* of QA systems means a simplification and results might look different in an answer selection setting where the system must select from the top k answers for $k \gg 1$. Moreover the QA systems represented in the AVE07 test set for German might have used some structure-sensitive validation themselves so that applying a similar validation method once again is no longer effective. In order to clarify these issues, future work will include experiments with variants of the robust inference method and a comparison with overlap methods and approximate graph matching. The first change to MAVE will be leveraging machine learning, however, which will replace the hand-coded formulas for the main validation decision and also for the answer-type check.

References

[1] Marie-Catherine de Marneffe, Bill MacCartney, Trond Grenager, Daniel Cer, Anna Rafferty, and Christopher D. Manning. Learning to distinguish valid textual entailments. In *Proc. of the 2nd Pascal RTE Challenge Workshop*, 2006.

[2] Ingo Glöckner. University of Hagen at QA@CLEF 2006: Answer validation exercise. In *Working Notes for the CLEF 2006 Workshop*, Alicante, Spain, 2006.

[3] Ingo Glöckner. Filtering and fusion of question-answering streams by robust textual inference. In *Proceedings of KRAQ'07*, Hyderabad, India, 2007.

[4] Ingo Glöckner, Sven Hartrumpf, and Johannes Leveling. Logical validation, answer merging and witness selection: A study in multi-stream question answering. In *Proc. of RIAO-07*, Pittsburgh, 2007.

[5] Sven Hartrumpf. *Hybrid Disambiguation in Natural Language Analysis*. Der Andere Verlag, Osnabrück, Germany, 2003.

[6] Hermann Helbig. *Knowledge Representation and the Semantics of Natural Language*. Springer, 2006.

[7] Anselmo Peñas, Álvaro Rodrigo, Valentin Sama, and Felisa Verdejo. Testing the reasoning for question answering validation. *Journal of Logic and Computation, Special Issue on Natural Language and Knowledge Representation*, to appear.