Question Answering with Joost at CLEF 2008*

Gosse Bouma, Jori Mur, Gertjan van Noord,
Lonneke van der Plas and Jörg Tiedemann
Information Science
University of Groningen
g.bouma@rug.nl

Abstract

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: H.3.1 Content Analysis and Indexing; H.3.3 Information Search and Retrieval; J.5 [**Arts and Humanities**]: Language translation; Linguistics

General Terms

Algorithms, Measurement, Performance, Experimentation

Keywords

Question Answering, Dutch, Wikipedia, Information Retrieval, Query Expansion

1 Introduction

In this paper we describe the system we used for the CLEF 2008 monolingual Dutch and multilingual English-to-Dutch question answering task.

Our QA-system, Joost, is largely the same as used in for previous participations in CLEF. Joost is an open-domain QA-system for Dutch, which makes heavy use of syntactic information in all of its components. The text collections used for CLEF (Dutch newspaper text and Wikipedia) are parsed using the Alpino parser (van Noord, 2006), which performs part-of-speech and named entity tagging, and syntactic analysis using dependency relations. The QA-system has two major modules, one for answering questions on the basis of a database of relational information that was compiled off-line, and one for answering questions on the basis of linguistic search in text snippets returned by an IR-engine. We use linguistic information for question analysis, for relation extraction, for building the IR-index, and for searching and ranking potential answer strings. The Joost-system is described in more detail in Bouma et al. (2005).

In the following sections, we describe the parts of the system that were developed recently. In section 2, we describe an attempt to use relation tuples extracted from Wikipedia infoboxes, in addition to the other off-line relation extraction techniques we use. Section 3 describes our work on automatic query expansion for IR, and section 4 describes our experience with using Google Translate for question translation. Section 5 gives an overview of the results and error analysis.

*This research was carried out as part of the research program for *Interactive Multimedia Information Extraction*, IMIX, financed by NWO, the Dutch Organisation for Scientific Research.

2 Information Extraction from Infoboxes

More and more lemma's in Wikipedia use templates to generate so-called 'infoboxes'. A template is basically a list of attribute-value pairs with relevant information for a given entity. For instance, the template for a river may contain attributes for its source, length, area which it flows through, etc.

Templates can be harvested easily from the XML dump of Wikipedia (see for instance Auer et al. (2008)). We used XQuery[1] to extract all all attribute-value pairs from all templates present in the November 2006 version of the Dutch Wikipedia. We extracted over 1.3 million tuples of the form ⟨`object, attribute, value, templatename`⟩, i.e. ⟨`AFC Ajax, stadion, Amsterdam ArenA, Voetbal_club_infobox`⟩. 1,438 different templates were found, of which 842 were used at least 10 times. 18,332 different attribute names were found, of which 1,788 were used at least 10 times. A fair number of attributes is ambiguous (i.e. the attribute `period` is used both for chemical elements, for royal dynasties, for countries, and for (historical) means of transportation). In addition, many templates use numbers as attribute names. Inclusion of the template name allows one to interpret such attributes more accurately.

The information in template tuples is potentially very useful for QA. The example tuple above could be used, for instance, to answer the questions *In which stadium does Ajax play?* and *Which soccer club plays in the Amsterdam ArenA?* Note, however, that it may not always be easy to link a question to the right attribute-value pair. For a number of frequent question types (inhabitants, birth date, capital) we specified manually what the names of matching attributes are. In addition, we tried to map general question types to a matching attribute. I.e. for all questions of the form *Who is the* `Predicate` *of* `Name`*?* we check whether a tuple ⟨`Name,Predicate,Value, Template`⟩ exists.

The number of questions that was actually answered using a tuple extracted from templates was small. Only two questions (about the number of inhabitants of a French and an Italian community) were answered this way. One of the answers was wrong because several French and Italian communities with the same name (*La Salle*) exist.

There are at least three reasons for the small impact of information extraction from templates:

1. The use of templates was not fully exploited in the November 2006 version of the Dutch Wikipedia. For instance, the current (2008) page for Piet Mondriaan mentions his birth date in an infobox, but the 2006 version of this page did not yet contain an infobox.

2. Connecting questions to the right tuples can be difficult. To some extent, the same problems arise as in QA in general, i.e. the predicate and name used in the user question are not necessarily identical to the attribute and spelling of the name in the tuple. For instance, no answer for the question *what is the area of Suriname?* was found, eventhough a tuple containing the answer exists. The tuple contains the attribute `km2`, however, which was not linked to questions about area.

3. The number of *simple* factoid questions in the test set, for which a tuple might exist at least in theory, is small.

Use of a more recent version of Wikipedia is likely to improve the coverage of templates. A more radical solution would be the automatic generation and completion of infoboxes, as proposed by Wu and Weld (2007). In addition, the connection between question analysis and tuple matching could be improved. Finally, inference rules could be added to enhance the information that can be obtained from tuples (i.e. if Louis XVIII is the *successor* of Napoleon Bonaparte, we may infer that Napoleon Bonaparte is also the *predecessor* of Louis XVIII).

[1] `www.w3.org/TR/xquery`

3 Query Expansion

Usually there are many possible ways to state a question corresponding to the user's information need. Often there is a discrepancy between the terminology used by the user and the terminology used in the document collection to describe the same concept. A document might hold the answer to the user's question, but it will not be found due to the TERMINOLOGICAL GAP. Moldovan et al. (2002) show that their system fails to answer many questions (25.7%), because of the terminological gap, i.e. keyword expansion would be desirable but is missing. Query expansion techniques have been developed to bridge this gap.

Besides the terminological gap there is also a KNOWLEDGE GAP (van der Plas and Tiedemann, 2008), i.e., documents are missed or do not end up high in the ranks, because additional world knowledge is missing. In these cases we are not speaking of simple synonyms but words belonging to the same subject field. For example, when a user is looking for information about the explosion of the first atomic bomb, in his/her head a subject field is active that could include: war, disaster, World War II.

For our CLEF 2008 submission we have experimented with various corpus-based methods to acquire semantically related words that can be used for query expansion: the SYNTAX-BASED METHOD, the ALIGNMENT-BASED METHOD, and the PROXIMITY-BASED METHOD. The nature of the relations between words found by the three methods is very different. Ranging from free associations to synonyms. Apart from these resources we have used categorised named entities, such as *Van Gogh* IS-A *painter*.

3.1 Extraction of Lexico-semantic Information

The automatic extraction of lexico-semantic information from corpora for query expansion is based on distributional methods. The following information sources have been applied:

- Nearest neighbours from proximity-based distributional similarity
- Nearest neighbours from syntax-based distributional similarity
- Nearest neighbours from alignment-based distributional similarity

The difference between these three approaches is the context used for computing the similarity between various word types. The way the context is defined determines the type of lexico-semantic knowledge we will retrieve.

For example, the proximity-based technique uses n surrounding words. In that case proximity to the head word is the determining factor. The nearest neighbours resulting from such methods are rather unstructured as well. They are merely associations between words, such as *baby* and *cry*. We have used the 80 million-word corpus of Dutch newspaper text (the CLEF corpus) that is also part of the document collection in the QA task to retrieve co-occurrences within sentences.

The second approach uses syntactic relations to describe the context of a word. We have used several syntactic relations to acquire syntax-based context for our headwords. This method results in nearest neighbours that at least belong to the same semantic and syntactic class, for example *baby* and *son*. We have used 500 million words of newspaper text (the TwNC corpus parsed by Alpino (van Noord, 2006)) of which the CLEF corpus is a subset.

The third approach uses cross-lingual information derived from parallel corpora to form the context of a word. For this we retrieved translations from automatically word-aligned corpora and, therefore, this method is called the alignment-based approach. This method results in even more tightly related data, as it mainly finds synonyms, such as *infant* and *baby*. We have used the Europarl corpus (Koehn, 2003) to extract word alignments from.[2]

The calculation of similarities between any headword is done in the same way for all three approaches. We gathered nearest neighbours for a frequency-controlled list of words, that was

[2]In van der Plas and Tiedemann (2006) there is more information on the syntax-based and alignment-based distributional methods.

Lex. info	Nouns	Adj	Verbs	Proper
Proximity	5.3K	2.4K	1.9K	1.2K
Syntax	5.4K	2.3K	1.9K	1.4K
Align	4.0K	1.2K	1.6K	
Cat. NEs				218K

Table 1: Number of words for which lexico-semantic information is available

still manageable to retrieve. We included all words (nouns, verbs, adjectives and proper names) with a frequency of 150 and higher in the CLEF corpus. This resulted in a ranked list of nearest neighbours for the 2,387 most frequent adjectives, the 5,437 most frequent nouns, the 1,898 most frequent verbs, and the 1,399 most frequent proper names. For all words we retrieved a ranked list of its 100 nearest neighbours with accompanying similarity score.

There is one additional source of information also derived automatically from corpora, categorized named entities. They are a by-product of the syntax-based distributional method. From the example in (1) we extract the apposition relation between *Van Gogh* and *schilder* 'painter' to determine that the named entity *Van Gogh* belongs to the category of painters.

(1) Van Gogh, de beroemde schilder huurde een atelier, Het Gele huis, in Arles.
'Van Gogh, the famous painter, rented a studio, The Yellow House, in Arles.'

We used the data of the TwNC corpus (500M words) and Dutch Wikipedia (50M words) to extract apposition relations. The data is skewed. The Netherlands appears with 1,251 different labels. To filter out incorrect and highly unlikely labels (often the result of parsing errors) we determined the relative frequency of the combination of the named entity and a category with regard to the frequency of the named entity overall. All categorised named entities with relative frequencies under 0.05 were discarded. This cutoff made the number of unwanted labels considerably lower.

In Table 1 we see the amount of information that is contained in individual lexico-semantic resources.

3.2 Query Expansion in Joost

We have implemented look-up functions for the integration of the information sources described above to be used for automatic query expansion in the passage retrieval component of Joost. The top-5 nearest neighbours obtained by the three distributional methods with scores above 0.2 were selected as expansion terms for words found in the query.

The categorised named entities were used in both directions, to expand named entities ("van Gogh") with the corresponding label ("painter") and to expand nouns ("painter") with possible instantiations of that label ("van Gogh, Rembrandt, ..."). In the second case we discarded nouns with more than 50 expansions as these were deemed too general and hence not very useful.

All expansion terms were added as root forms to the query using a keyword weight such that all expansions for one original keyword add up to 0.5.

For evaluation of the various expansion techniques we applied data collected from the CLEF Dutch QA tracks. We used the question sets from the competitions of the Dutch QA track in 2003, 2004, and 2005 (774 in total). Questions in these sets are annotated with valid answers found by the participating teams. We expanded these list of valid answers where necessary.

In order to look at the impact of query expansion on individual questions we list the number of questions that get higher and lower reciprocal rank scores after applying each resource (table 3).

Apart from expansions on adjectives, the impact of the expansion is substantial. The fact that adjectives have so little impact is due to the fact that there are not many adjectives among the query terms.[3] The proximity-based method clearly provides information about proper names that

[3]Moreover, the adjectives related to countries, such as *German* and *French* and their expansion *Germany, France* are handled by a separate list.

MRR				
SynCat	Syntax	Align	Proxi	Cat.NEs
Nouns	51.15	51.21	51.38	51.75
Adj	52.27	52.38	51.71	
Verbs	52.33	52.21	52.62	
Proper	50.16		53.94	**55.68**
All	51.21	51.02	53.36	**55.29**

Table 2: MRR scores for the IR component with query expansion from several sources. The baseline is 52.36

#questions (+/-)				
SynCat	Syntax	Align	Proxi	Cat.NEs
Nouns	28/61	17/58	64/87	17/37
Adj	1/2	1/2	31/47	
Verbs	5/10	8/32	51/56	
Proper	30/80		76/48	157/106
All	56/131	25/89	161/147	168/130

Table 3: Number of questions that receive a higher (+) or lower (-) RR when using expansions from several sources

are more relevant for the corpus used for QA, as it is built from a subset of that same corpus. This shows the advantage of using corpus-based methods.

The type of expansions that result from the proximity-based method have a larger effect on the performance of the system than those resulting from the syntax-based method. In Chapter 5 of van der Plas (2008) we explain in greater detail that the proximity-based method uses frequency cutoffs to keep the co-occurrence matrix manageable. The larger impact of the proximity-based nearest neighbours is probably partly due to this decision. The largest impact results from expanding proper names with categorised named entities. We know from Table 1, that this resource has 70 times more data than the proximity-based resource.

For most of the resources the number of questions that show a rise in RR is smaller than the number of questions that receive a lower RR, except for the expansion of proper names by the categorised named entities and the proximity-based method. The expansions resulting from the syntax-based method do not result in any improvements. As expected, the expansion of proper names from the syntax-based method hurts the performance most. Remember that the nearest neighbours of the syntax-based method often include co-hyponyms. For example, *Germany* would get *The Netherlands* and *France* as nearest neighbours. It does not seem to be a good idea to expand the word *Germany* with other country names when a user, for example, asks the name of the Minister of Foreign Affairs of Germany. However, also the synonyms from the alignment-based method do not result in improvements.

The categorised named entities provide the most successful lexico-semantic information, when used to expand named entities with their category label. The MRR is augmented by almost 3,5%. It is clear that using the same information in the other direction, i.e. to expand nouns with named entities of the corresponding category hurts the scores. The proximity-based nearest neighbours of proper names raises the MRR scores with 1,5%.

Finally we have tested the entire QA system with the different expansion settings in passage retrieval. The scores on the same development set (CLEF 2003-2005) are shown in table 2.

The syntax-based, and the alignment-based nearest neighbours we have used all expansions for all syntactic categories together. For the proximity-based nearest neighbours and the categorised named entities we have limited the expansions to the proper names as these performed rather well.

The positive effect of using categorised named entities and proximity-based nearest neighbours for query expansion is visible in the CLEF scores as well, although less apparent than in the MRR

CLEF score				
Syntax	Align	Proxi	Cat.NEs	Baseline
47.0	46.6	47.6	47.9	46.8

Table 4: CLEF scores of the QA system with query expansion from several sources

scores from the IR component in Table 2.

Remember from the introduction that we made a distinction between the terminological gap and the knowledge gap. The lexico-semantic resources that are suited to bridge the terminological gap, such as synonyms from the alignment-based method, do not result in improvements in the experiments under discussion. However, the lexico-semantic resources that may be used to bridge the knowledge gap, i.e. associations from the proximity-based method and categorised named entities, do result in improvements of the IR component. More details about our experiments and an error analysis can be found in (van der Plas and Tiedemann, 2008).

4 Multilingual QA

In CLEF 2006 and 2007 we used Babelfish for automatic translation of the English questions into Dutch. The automatically translated questions were then used as input to the QA-system. In 2007, we added a preprocessing module, which detects names and concepts in the source question, and translated these using, among others, Wikipedia cross-language links (Bouma et al., to appear). Names for which a Wikipedia lemma existed, but no translation in Dutch, were kept as is (this will give the correct result for most person names). This module was added since we noted that Babelfish was particularly poor at recognizing and adequately translating named entities.

Recently, Google Translate has added English to Dutch translation to the set of language pairs for which it can do translation. For CLEF 2008, we used this automatic translation system[4] instead of Babelfish.

One interesting difference is the treatment of named entities. Google Translate is far more accurate in this respect than Babelfish. Therefore, we ran an experiment in which we used Google Translate to translate both the original source questions and source questions in which named entities had been preprocessed as in the system used for CLEF 2007. The resulting translations contained 12 errors that could be attributed to wrong processing in case we did not do any preprocessing, and 27 errors in case we applied preprocessing. Preprocessing helps to find translations for entities such as *Captain Hook* (*Kapitein Haak*), *Drake Passage* (*Straat Drake*) and *Zimmer Tower* (*Zimmertoren*). On the other hand, the preprocessed data often incorrectly marks names and concepts (such as *Earth, Swiss, Dutch* and *the Netherlands*) for which no translation was found (i.e. no corresponding page exists in the Dutch Wikipedia). These are incorporated verbatim in the translation.

As a result of this evaluation, we decided not preprocess the questions.

5 Results and Error Analysis

The official CLEF evaluation results are given for all four submitted runs (Dutch monolingual with and without query expansion, and English-to-Dutch with and without query expansion) are given in table 5. Results per question type for the Dutch monolingual run with query expansion are given in table 6. The question classification used by CLEF is coarse, and does not necessarily correspond to the question types as they are assigned by our question analysis module. Therefore, we also performed an evaluation ourselves (see table 7). In this case, we can give more detailed results per question type. To get more robust results, we also compute the MRR over the first 5 answers.

[4]We used the api at `code.google.com/p/google-api-translate-java/`

Run	Accuracy (%)	Right	ineXact	Unsupported	Wrong
Dutch-mono	25.0	50	11	1	138
Dutch-mono + QE	25.5	51	10	3	136
En-Du	13.5	27	10	6	157
En-Du + QE	13.5	27	10	6	157

Table 5: Official CLEF scores for the monolingual Dutch task and the bilingual English to Dutch task (200 questions), with and without Query Expansion (QE) .

Q type	# q's	Accuracy (%)	Right	ineXact	Unsupported	Wrong
Factoids	151	24.5	37	4	3	107
List	10	0.0	0	4	0	6
Definition	39	35.9	14	2	0	23
Temp. Restricted	13	15.4	2	1	1	9

Table 6: Results per question type for the best Dutch monolingual run.

Q type	MRR	CLEF	#
born_loc	1.000	1.000	1
function	0.750	0.750	4
inhabitants	0.750	0.667	3
born_date	0.500	0.500	2
subject_predicate	0.500	0.500	2
number_of	0.499	0.474	19
name_of	0.367	0.333	3
definition	0.340	0.261	23
dimension	0.333	0.333	6
name	0.327	0.308	13
person	0.312	0.250	8
what	0.308	0.231	13
location	0.300	0.300	20
event_date	0.246	0.188	16
measure	0.222	0.167	6
organization	0.208	0.167	6
which	0.181	0.143	42
creator	0.167	0.000	1
age	0.000	0.000	3
founded	0.000	0.000	1
cause	0.000	0.000	1
yesno	0.000	0.000	1
function_of	0.000	0.000	1
nil	0.000	0.000	4
win_who	0.000	0.000	1
total	0.298	0.260	200

Table 7: Results per Joost question type a Dutch monolingual run. We give the Mean Reciprocal Rank (of the first 5 answers), accuracy of the first answer (CLEF), and the number of questions that was assigned this question type. Scores are based on our own judgements.

5.1 Question Analysis

Apart from definition questions, the most frequent question types according to the question analysis of Joost were *which*-questions (42, with a CLEF-score of 14%), *location*-questions (20, 30% correct), *number-of*-questions (19, 47% correct), *date*-questions (16, 18% correct), *what*-questions (13, 23% correct), and *who*-questions (13, 30% correct). These are all question types for which it is hard to use off-line methods. For a number of questions in the very general *which/what/who* question types, more specific question types could have bene given. 2 questions could not be analyzed (*What are the first names of her two sons?* and *How high can hairy sedge grow?*). In first name and dimension questions, the system expects to find a named entity, and these could not be found for the examples above. One question was (incorrectly) analyzed as a yes/no question (*Is the heart located left or right in the body?*).

5.2 Definitions

Definition questions are relatively easy to answer. In particular, the first Wikipedia sentence for a named entity or concept usually contains a definition. Still, of the 23 questions classified as definition questions by Joost[5] only 10 are answered correctly. Errors are mostly due to name and spelling variation and tokenization problems. I.e. we do not find a definition for *Mitchell Feigenbaum*, because the first sentence of the corresponding lemma mentions *Mitchell Jay Feigenbaum*, we do not find a definition for *provinciale landschappen* (*provincial landscapes*), although a definition for *Provinciale Landschappen* (upper case) exists.

5.3 Anaphora

The follow-up questions contained 43 expressions that were considered to be anaphoric by the system. Some definite NPs were incorrectly treated as anahoric (*de duivel (the devil), de moeder (the mother)*). The system found the correct antecedent in 21 cases (almost 50%). An important source of errors was resolving an expression to an incorrect answer of a previous question.

5.4 Off-line techniques

We used two methods for answering questions using information that was collected off-line. The method based on Wikipedia infoboxes was discussed in section 2. In addition we use a technique which extracts answers for frequently asked question types by searching the full corpus for relevant tuples (see Mur (2008) for a detailed overview).

33 questions were answered using table look-up, of which 16 were definition questions. The accuracy of the answers found by means of table look-up is somewhat better (33% correct) than for the system in general.

References

Auer, Sören, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2008. Dbpedia: A nucleus for a web of open data. pages 722–735.

Bouma, Gosse, Ismail Fahmi, Jori Mur, Gertjan van Noord, Lonneke van der Plas, and Jörg Tiedeman. 2005. Linguistic knowledge and question answering. *Traitement Automatique des Langues*, 2(46):15–39.

Bouma, Gosse, Geert Kloosterman, Jori Mur, Gertjan van Noord, Lonneke van der Plas, and Jörg Tiedemann. to appear. Question answering with Joost at QA@CLEF 2007. In Carol Peters et al., editor, *Proceedings of CLEF 2007*. Springer, Berlin.

[5]The official CLEF scores mention 39 definition questions, but these also include a number of questions (asking for the chairman or founder of an organisation, the highest point in Belgium) that are normally not considered to be definition questions.

Koehn, Philipp. 2003. Europarl: A multilingual corpus for evaluation of machine translation. unpublished draft, available from `http://people.csail.mit.edu/koehn/publications/europarl/`.

Moldovan, D., M. Passça, S. Harabagiu, and M. Surdeanu. 2002. Performance issues and error analysis in an open-domain question answering system. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Mur, Jori. 2008. *Off-line Answer Extraction for Question Answering*. Ph.D. thesis, University of Groningen, Groningen.

van der Plas, L. and Jörg Tiedemann. 2008. Using lexico-semantic information for query expansion in passage retrieval for question answering. In *Proceedings of the Coling workshop Information Retrieval for Question Answering (IR4QA)*. To appear.

van der Plas, Lonneke. 2008. *Automatic lexico-semantic acquisition for question answering*. Ph.D. thesis, University of Groningen. To appear.

van der Plas, Lonneke and Jörg Tiedemann. 2006. Finding synonyms using automatic word alignment and measures of distributional similarity. In *Proceedings of ACL/Coling*.

van Noord, Gertjan. 2006. At last parsing is now operational. In Piet Mertens, Cedrick Fairon, Anne Dister, and Patrick Watrin, editors, *TALN06. Verbum Ex Machina. Actes de la 13e conference sur le traitement automatique des langues naturelles*. pages 20–42.

Wu, Fei and Daniel S. Weld. 2007. Autonomously semantifying wikipedia. In *CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 41–50, New York, NY, USA. ACM.