Evaluating Multi-focus Natural Language Queries over Data Services

Silvia Quarteroni, Vincenzo Guerrisi, Pietro La Torre

DEI - Politecnico di Milano
via Ponzio, 34/5 - 20133 Milan, Italy
quarteroni@elet.polimi.it, guerrisi.vincenzo@gmail.com, latorre.pietro@gmail.com

Abstract

Natural language interfaces to data services will be a key technology to guarantee access to huge data repositories in an effortless way. This involves solving the complex problem of recognizing a relevant service or service composition given an ambiguous, potentially ungrammatical natural language question. As a first step toward this goal, we study methods for identifying the salient terms (or foci) in natural language questions, classifying the latter according to a taxonomy of services and extracting additional relevant information in order to route them to suitable data services. While current approaches deal with single-focus (and therefore single-domain) questions, we investigate multi-focus questions in the aim of supporting conjunctive queries over the data services they refer to. Since such complex queries have seldom been studied in the literature, we have collected an ad-hoc dataset, SeCo-600, containing 600 multi-domain queries annotated with a number of linguistic and pragmatic features. Our experiments with the dataset have allowed us to reach very high accuracy in different phases of query analysis, especially when adopting machine learning methods.

Keywords: Question Answering, Web Data Services, Machine Learning

1. Introduction

Web data providers have recently made a wealth of information available, be it via APIs as in Google Fusion Tables[1] or via search-specific languages such as the Yahoo Query Language[2]. Data sources are usually wrapped as *data services* specified by input and output parameters, supporting complex Web queries.

While it is generally possible for experts only to write logical queries or set up query interfaces to data services, it is widely believed that natural language (NL) interfaces will be key to guarantee effortless access to huge data repositories to a large community of non-expert users (Kaufmann and Bernstein, 2007). However, supporting a NL interface involves resolving the ambiguity of free-text queries and performing an accurate mapping from the lexical level of the question to the semantic level needed to compose a logical query, i.e. a statement describing relevant data services and their join criteria, constraints and selection conditions. We refer to this process as **query analysis**.

In Section 2. of this paper, we discuss related work on the NL query analysis problem leading to the creation of logical queries, a vital process in the Search Computing project (SeCo) that aims at on-the-fly composition of data services via a variety of application types (Braga et al., 2011). Since it is a requirement in SeCo to support conjunctive queries over data services but related work is centered on single-domain questions, Section 3. introduces the multi-focus, multi-domain NL question corpus that we have collected for the purpose of our studies. The corpus, named **SeCo-600**, has served as an evaluation dataset for a number of query analysis approaches, described in Section 4. evaluated in Section 5.

2. Related Work

In the Semantic Web area, natural language (NL) interfaces to ontologies have been proposed in a number of studies as an alternative to keyword-based interfaces or interfaces based on query languages (Damljanovic et al., 2010b; Kaufmann and Bernstein, 2007). Generally speaking, methods in this field attempt to perform an exact mapping of the NL query into a logical formula in order to access knowledge, structured in e.g. RDF triples. Typical approaches in this direction involve a combination of statistical techniques (syntactic parsing) and semantic operations to identify ontology concepts in the user's input. For instance, QUERIX (Kaufmann et al., 2006) combines the Stanford probabilistic parser (Klein and Manning, 2003) with the WordNet[3] lexical database to obtain RDF triples from natural language user queries. Similarly, in (Damljanovic et al., 2010a), the question is parsed, then the most relevant data service based on terminological similarity as found in a reference ontology; PANTO (Wang et al., 2007) translates the parse tree of a natural language query into SPARQL by exploiting a reference lexicon.

With respect to the above approaches, we aim at solving a different problem: not only we expect search engine-style, potentially ungrammatical interaction, but we also deal with a variety of heterogeneous data sources for which there is *a priori* no reference ontology. For these reasons, we cannot assume to have fully parsable queries or a consistent, stable domain lexicon.

A widely adopted method for the analysis of NL queries over ontologies is *focus extraction*, i.e. the identification of the question's salient term in the purpose of matching it to a relevant ontology concept. For instance, the method in (Damljanovic et al., 2010a), based on deep syntactic parsing, identifies the focus based on pre-preterminal nodes[4] in the query's parse tree: in particular, it returns the first pre-preterminal node tagged as a noun or noun phrase. Following this approach, the question "Where are cheap ac-

[1] code.google.com/apis/fusiontables

[2] developer.yahoo.com/yql

[3] wordnet.princeton.edu

[4] A node is a pre-preterminal if all its children are preterminals, i.e. nodes with one child which is itself a leaf. In Fig. 1, the pre-preterminal of *Where* is WHADVP.

commodation and a Japanese restaurant near Covent Garden?" syntactically parsed as in Figure 1, would be given focus *cheap accommodation*. The approach in (Li, 2010) exploits the chunked textual annotation obtained via a shallow parser, so that the rightmost noun phrase chunk before any prepositional phrase or adjective clause is marked as the syntactic focus. Following this rule, the above question, chunked as in Figure 2., would therefore have focus *a Japanese restaurant*.
It may be noted that both methods are designed for single-focus queries, making them unfit to locate the two foci in the example (*cheap accommodation* and *a Japanese restaurant*). The fact that they are rule-based and dependent on the deep annotation of text structure makes them struggle in capturing the semantics of queries with multiple foci, a task that however is the fundamental pre-requisite in SeCo as the identification of relevant data services depends on a correct identification of NL query foci.
Indeed, when it comes to identifying a relevant data service given question terms, we can take advantage of a large body of literature in the Question Answering field that has effectively applied machine learning approaches joined with lexical and shallow syntactic features to *question classification* (Li and Roth, 2002). We pursue a similar direction in Section 4.3., as our classification problem is analogous; however, in this case we need to account for questions characterized by multiple foci, i.e. to deal with the problem of splitting questions based on the span of their foci and then matching each sub-question to a data service class. This *question segmentation* problem is similar to the automatic segmentation of a spoken conversational turn into dialog act spans, for which we have observed effective results using machine learning solutions (Quarteroni et al., 2011).
Finally, the step of identifying relevant additional attributes of the query in order to route it to the most appropriate service is an information extraction problem for which different methods exist; generally speaking, such methods draw from both open-domain models for the identification of instances of generic entities (e.g. Named Entity recognizers) to handwritten (e.g. rule-based) methods for domain-specific extraction. We illustrate a number of these approaches in Section 4.4.

3. The SeCo-600 corpus

The SeCo-600 corpus[7] contains 600 spontaneous *multidomain* user queries collected to fit scenarios relevant to the SeCo project, such as finding accommodation and restaurants in a tourist area or interesting events taking place nearby. Such queries were collected in order to evaluate the four phases of NL query analysis in SeCo:

1. *focus extraction*, i.e. the identification of a question's salient noun(s) or noun phrase(s);
2. *question segmentation* according to its foci, leading to its subquestions;
3. *(sub)question classification*, i.e. the categorization of each (sub)question according to a chosen taxonomy;
4. *intent modifier extraction*, i.e. the interpretation of relevant (sub)question terms to provide the constraints leading to a full-fledged logical query.

For instance, in the question "Where can I find cheap accommodation in London?", the focus is *cheap accommodation*, that can be further reduced to its head *accommodation* using e.g. the rules in (Collins, 1999). Question class depends on a chosen taxonomy; in SeCo this coincides with the 7 following data service classes: *Cinema, Movie, Hotel, Restaurant, Event, Point of Interest (POI)* and *Other* (see Table 4 for relative frequencies). In the current example, the correct class label would be *Hotel*. Finally, *cheap* and *in London* are values of *intent modifiers*, i.e. constraints to be included in the logical query. Intent modifiers may be modeled as attributes of entities in a given domain representation, e.g. *Location.country* and *Restaurant.name* in a restaurant reservation domain. In SeCo, we have represented 14 different intent modifier types, characterized by a large coverage of SeCo scenarios: these appear in 1229 instances throughout the corpus.
SeCo-600 queries are delivered in both a textual version and an annotated version, as described below.

3.1. Textual version

The textual version of the corpus reports one question per line without any filter or normalization. Due to its spontaneous nature, the corpus offers a variety of syntax forms, ranging from keyword-style queries ("nice hotel paris cheap events of design", "movies with sean connery shown in Medusa cinemas") to full-fledged natural language containing anaphora ("where could i find a wellness center in palermo and a mcdonald close to *it*?").

3.2. Annotated version

In the annotated version of the corpus, question representations are separated by empty lines. First, questions have been automatically annotated with linguistic features derived using state-of-the-art resources. As questions have been manually split into sub-queries and each sub-query has been manually tagged according to its syntactic focus and to the 7 SeCo service classes. Additional information regarding each question's intent modifier has been added. As a result, each question q is represented in column format, where each row corresponds to a question word w_i and columns report the following annotation:

WORD w_i as it appears in the textual query file

POS w_i's Part-of-Speech tag, obtained via the OpenNLP chunker[8]

CHUNK label of the syntactic chunk w_i belongs to, obtained via the OpenNLP chunker (in IOB notation)

CLASS class of the subquestion where w_i is located (in IOB notation)

FOCUS whether w_i is within or outside the syntactic head of a question focus (in IOB notation)

[7] Available at: `http://search-computing.it/`

[8] `opennlp.sourceforge.net`

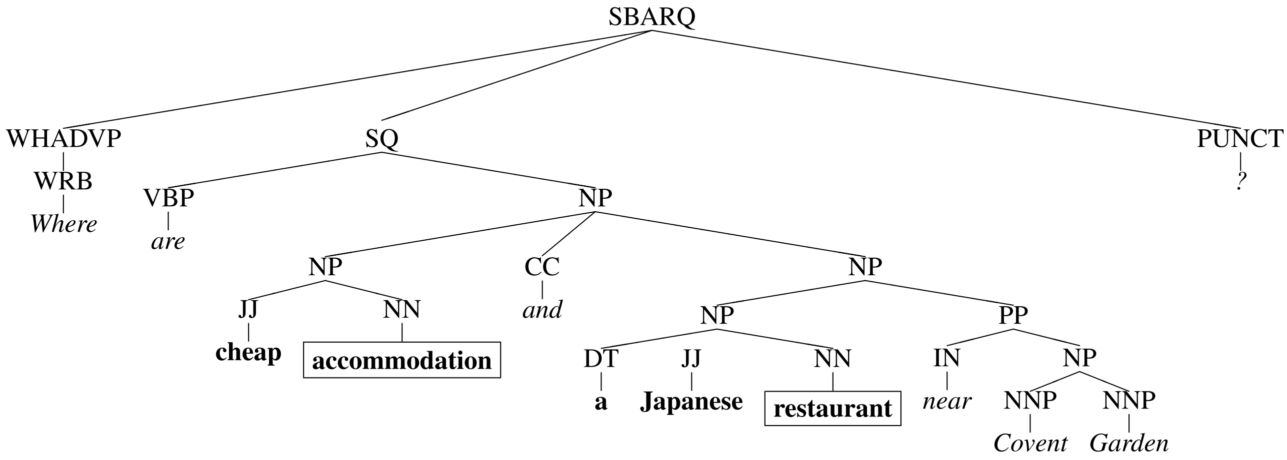

Figure 1: Top syntactic parse tree of the question "Where are cheap accommodation and a Japanese restaurant near Covent Garden?" according to the Stanford Parser (Klein and Manning, 2003). Actual foci are in boldface with focus heads boxed.

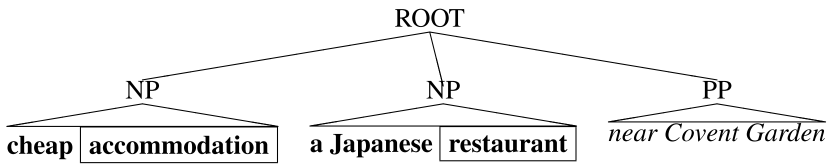

Figure 2: Phrases in the shallow syntactic parse tree of the question "Where are cheap accommodation and a Japanese restaurant near Covent Garden?" according to the OpenNLP chunker[6]. Actual foci are in boldface with focus heads boxed.

INTENT MODIFIER w_i's intent modifier class, if any (in IOB notation).

A representative example of the annotation for a SeCo query is reported in Table 1.

4. Query Analysis Models

We now describe a number of approaches methods applied to perform the NL query analysis steps sketched in Section 3. These have been validated on the SeCo-600 corpus as reported in Section 5.

4.1. Focus Extraction

As previously discussed, a number of focus extraction methods exploit regularities in the syntactic structure of natural language queries (Damljanovic et al., 2010a; Li, 2010). However, as illustrated in our experiments (Sec. 5.), these methods yield good results when dealing with single focus queries, but are ineffective at handling questions with multiple foci: this suggests that relying too heavily on the question syntax becomes more and more challenging as data quality deteriorates (as e.g. in free-form Web queries) and the complexity of the question increases.

To contrast this issue, we propose to combine the extraction of lexical and morphological annotations with the learning of robust discriminative classifiers.

We train machine learning classifiers to determine whether each word w in the question is a focus syntactic head (focus head, in brief) or not; this is similar to the problem of turn segmentation into dialog act markers tackled in e.g. (Quarteroni et al., 2011).

As a learning algorithm, we adopt first-order linear-chain Conditional Random Fields (CRFs), a category of probabilistic learners frequently used for labeling and segmenting structured data (Lafferty, 2001). CRFs are undirected graphical models that define a conditional probability distribution over label sequences (i.e. focus tags) given a particular observation sequence (i.e. words). We study different combinations of the following features (validated in Section 5.1.):

1. word unigrams situated within an interval of n words centered on the current word: $[-n, n]$, $n \leq 2$ (case $n = 0$ corresponds to the current word w only);

2. word Part-of-Speech (POS) tags taken in the same interval, as obtained via the OpenNLP toolkit[9];

3. word bigrams, i.e. sequences of two consecutive words comprising the current word w (i.e. taken in the interval $[-1, 1]$ with respect to w);

4. POS bigrams in the same interval.

4.2. Question Segmentation

The segmentation of a question q into its subqueries $q_i, i \in \{1, .., n\}$ is not trivial: for instance, a simple approach using q's conjunctions as subquery delimiters may lead to unintended results in the sentence "I'm looking for a bed and breakfast". For these reasons, we investigated a machine learning approach to question segmentation that consists in learning a binary classifier that, given a word w, determines

[9] `opennlp.sourceforge.net`

Table 1: Sample question from the SeCo-600 corpus: "single or double room in Rouen and nearest cinema"

WORD	POS	CHUNK	CLASS	FOCUS	INTENT MODIFIER
single	JJ	B-NP	B-hotel	O	O
or	CC	I-NP	I-hotel	O	O
double	JJ	I-NP	I-hotel	O	O
room	NN	I-NP	I-hotel	B-FOCUS	O
in	IN	B-PP	I-hotel	O	O
Rouen	NNP	B-NP	I-hotel	O	B-IM_LocCity
and	CC	I-NP	B-cinema	O	O
nearest	JJS	I-NP	I-cinema	O	O
cinema	NN	I-NP	I-cinema	B-FOCUS	O

whether w is situated at the beginning of a new segment or not. An evident criterion to make this distinction appears to be the word neighborhood: for instance, the absence of a previous word is a useful indicator for the beginning of a segment. POS tags are also potentially useful features, as e.g. a conjunction (CC tag) is a strong indicator of the presence of a new sub-query.
To leverage the above criteria, we adopt the same features devised for focus extraction (see Section 4.1.) – i.e. word and POS unigrams and bigrams – to build a binary CRF classifier for sub-question identification. Our results are reported in Section 5.2.

4.3. Subquestion Classification

Starting from a multi-domain question and given a taxonomy of available services as labels (e.g. *Cinema*, *Hotel*), the goal of question classification is to map each subquery q_i of a question q to its most likely label c_i.
The (sub)question classification problem is different from the problem of labeling a question word w as "focus" or "subquestion starter" in various respects: we are dealing with a multi-classification problem, the boundaries of q_i are known from previous steps (this task relies heavily on an accurate segmentation of q), and finally long-distance relationships between words are potentially useful.
We therefore conduct the classification task using a different discriminative approach, i.e. the learning of Support Vector Machine classifiers (SVMs), based on the set of previously split questions. A binary classifier is built for each question class, and results are combined according to a one-vs-all regime in order to assign the strongest label c_i to each subquery q_i. Due to the limited amount of sub-queries (about 700) in comparison with the number of labels (7) and the consequent data sparsity issue, we only study two types of features:

1. the bag-of-words (BOW) feature, consisting of all the words in q_i stemmed following (Porter, 1980) for sparsity reduction;

2. the FOCUS feature, representing q_i's focus head.

These features are combined by summing linear kernel functions in SVM-light (Joachims, 1999): our results are reported in Section 5.3.

4.4. Intent Modifier Extraction

The final step in understanding the user question is the identification of relevant terms expressing constraints for the logical query – intent modifiers. Such terms include locations, dates, proper nouns and (optionally) other domain-specific attributes; recognizing instances of these will result in the choice of a specific data service over another belonging to the same data service class. For instance, identifying a location instance in a question classified as *Cinema* might result in choosing to route the query to a service returning cinemas based on their location rather than one returning cinemas based on movie titles.
SeCo scenarios cover heterogenous domains and applications encompassing a wide variety of entity types; we focused on the effective recognition of the generic types of entities using domain-independent approaches. We here illustrate our location extraction models as a representative case of intent modifier extraction: locations are not only among the most widespread named entities (the SeCo-600 corpus contains about 250 distinct cities and 75 countries), but also the most challenging as they often require disambiguation, e.g. as distinguishing between New York intended as a city or as a state.
An obvious choice for location recognition is to use statistical NER systems, such as LingPipe[10]: however, these identify entities at a coarse-grained level, e.g. referring to cities, countries or states with the more general notion of "Location". Therefore, additional methods may be needed to conduct the classification at a finer level as a subsequent step of the identification.
Furthermore, we consider methods based on lexicon lookup. We built two gazetteers based on instances extracted from GeoNames[11] for both cities and countries, addressing the identification of these entities by looking for exact or approximate matching (for the latter case, we consider an edit distance-based similarity exceeding 0.75 as a match).
The disambiguation issue is addressed by the use of *wikifiers*, i.e. tools that annotate phrases in text in terms of relevant Wikipedia page by disambiguating amongst alternatives based on the distribution of hyperlinks to Wikipedia

[10] alias-i.com/lingpipe
[11] http://www.geonames.org/

pages[12]. Our choice fell on TagMe (Ferragina and Scaiella, 2010), due to its robustness to short and badly structured text (such as SeCo queries). We connected annotations output by TagMe (e.g. `http://en.wikipedia.org/wiki/Lodi,_Italy`) to entities in the YAGO reference knowledge base (Suchanek et al., 2007), most of which refer to Wikipedia via the *hasWikipediaURL* property. This allowed us e.g. to identify entities of type *yagoGeoEntity* as locations.

5. Experimenting with the SeCo Corpus

We now illustrate our experiments on focus extraction, question segmentation and question classification using the SeCo-600 corpus. We evaluate the accuracy of each task in terms of F1 measure, a standard information retrieval metric combining precision (P) and recall (R): $F1 = \frac{2(P*R)}{(P+R)}$.

5.1. Focus Extraction

We have reimplemented the (Damljanovic et al., 2010a) and (Li, 2010) to serve as a baseline for evaluating focus extraction on the SeCo-600 corpus. As reported in Table 2, both methods, designed for single-focus queries, yield an F1 below 65%.

Table 2: Focus extraction results on the SeCo-600 dataset

Method	Accuracy (F1)
(Damljanovic et al., 2010a)	57.0%
(Li, 2010)	63.1%
CRF W[0,0]	85.3%± 3.0%
CRF **W[-2,2]+POS[-2,2]**	**94.0%± 1.5%**

Further to rule-based methods, we experimented with Conditional Random Fields combining different combinations of the features described in Section 4.1. using the CRF++ implementation (Kudo, 2005). We evaluate classification performance via ten-fold cross-validation to ensure the consistency of our results and report the F1 measure of each classifier in Table 2. On the SeCo-600 corpus, the bag-of-words (BOW) model, using only the current question word as the only feature feature, reaches a very high value of 85.3% compared to the best-performing rule-based model (Li, 2010), achieving 63.1%. Further feature combinations, joining the contribution of word and POS in the [-2,2] word range, yield a very satisfying 94% average accuracy, confirming the effectiveness of the CRF approach.

5.2. Question Segmentation

Table 3 reports the performance of the question segmentation algorithm on the SeCo-600 corpus (containing 888 question segments) using different feature combinations. These highlight a similar situation to the one observed for focus extraction, with the BOW model (W[0,0]) starting at 89.1% and increasing to 94.1% for the best found feature combination, joining the word and POS feature in a neighborhood spanning from the previous to the following word [-1,1].

[12] `en.wikipedia.org`

Table 3: Question segmentation results: F1-measure of different CRF models on the SeCo corpus

CRF Model	Accuracy (F1)
W[0,0]	89.1 ± 3.5%
W [-1,1]	70.3 ± 4.9%
POS [0,0]	88.8 ± 3.5%
POS [-1,1]	90.5 ± 3.2%
W+POS[-1,1]	**94.1 ± 3.0%**
W+POS[-2,2]	93.6 ± 2.9%

5.3. Question Classification

We conducted the question classification task by learning binary classifiers for each class, so that the final class to be assigned to a specific subquery is chosen according to a one-vs-all regime. We adopted the SVM-light implementation (Joachims, 1999) to learn different combinations of linear kernel functions based on the set of manually split questions in the SeCo-600 corpus. Our results, reported in Table 4, were obtained in a 10-fold cross validation regime to ensure robustness. As a general comment, overall results are very encouraging as, despite the small training dataset, classification accuracy reaches 86.1% with the BOW model and 92.7% with the BOW + FOCUS model.

A more detailed analysis suggests that the contribution of BOW combined with FOCUS leads to better results than BOW in most classes, and especially for *Cinema*. Class *Other* denotes a low accuracy: indeed, it is chosen whenever questions are too heterogeneous to be mapped to the remaining classes. Also, as the word "place" is strongly represented in the *POI* class, expressions such as "places where to eat" are erroneously classified as *POI* instead of their actual class *Food*; indeed, *POI* reaches the lowest accuracy as the FOCUS feature is less helpful in this circumstance and questions are strongly affected by lexical ambiguity. Finally, we note that when a query for a specific entity type contains constraints involving other services (e.g. a cinema in proximity of a hotel), constraint terminology may lead to a wrong prediction.

Table 4: Question Classification accuracy on SeCo-600

		Accuracy (F1)	
Class	Frequency	BOW	BOW+FOCUS
Hotel	22.8%	89.8 ± 8.6	90.2 ± 8.1
Food	19.4%	96.2 ± 4.2	96.7 ± 3.7
Other	15.0%	89.9 ± 7.6	92.0 ± 6.6
POI	11.3%	82.2 ± 10.7	80.7 ± 11.9
Event	10.6%	95.8 ± 6.5	96.1 ± 6.0
Cinema	12.6%	84.5 ± 12.6	97.3 ± 4.2
Movie	8.3%	86.1 ± 15.3	95.1 ± 7.7
Overall		89.5 ± 4.2	**92.7 ± 3.1**

Table 5: Intent Modifier extraction on the SeCo-600 corpus: *Location* entity

Method	Precision	Recall	F1
lexicon lookup (edit distance threshold = 0.75)	79.0%	54.7%	64.7%
statistical NER (Lingpipe)	71.0%	60.2%	65.2%
TagMe + YAGO validation	74.4%	65.0%	**69.4%**

5.4. Intent Modifier Extraction

Intent modifier extraction results for instances of *Location* are reported in Table 5. We note that while approaches based on general-purpose statistical NER systems such as LingPipe result in an F1 around 65% and the GeoNames-based approach has similar error rates, the TagMe approach described in Section 4.4. yields slightly higher results leading to the best score of 69.4%[13]: this may be reconnected to the higher recall offered by a robust method referring to Wikipedia as a source of relevant entities.

6. Conclusions

In this paper, we have addressed the requirements of natural language interfaces to data services, highlighting the need to extract information from queries dealing with multiple foci and domains. In particular, we concentrate on the four subsequent steps of focus extraction, question segmentation into sub-queries, sub-query classification and information extraction from subqueries, these lead to identifying relevant data services given a natural language question.

Since related work highlights the shortcomings of rule-based approaches and deep syntactic analysis for the interpretation of multi-domain queries, we propose a variety of robust models based on Conditional Random Fields, Support Vector Machines and shallow linguistic annotations to approach the above problems. The need to experiment with such data led us to the production of a corpus of 600 multi-focus, multi-domain annotated queries, the SeCo corpus, over which we have successfully evaluated our approaches to the above-mentioned tasks. In future work, we will continue our research on natural language query processing over data services by performing an end-to-end evaluation of query analysis in the SeCo project.

Acknowledgments

This work has been partly funded by the Search Computing project (European Commission IDEAS grant no. 227793).

7. References

D. Braga, M. Grossniklaus, F. Corcoglioniti, and S. Vadacca, 2011. *Efficient Computation of Search Computing Queries*, volume 6585 of *LNCS*, pages 141–155. Springer.

M. Collins. 1999. *Head-driven statistical models for natural language parsing*. Ph.D. thesis, University of Pennsylvania.

D. Damljanovic, M. Agatonovic, and H. Cunningham. 2010a. Identification of the question focus: Combining syntactic analysis and ontology-based lookup through the user interaction. In *Proceedings of LREC*.

D. Damljanovic, M. Agatonovic, and H. Cunningham. 2010b. Natural language interfaces to ontologies: Combining syntactic analysis and ontology-based lookup through the user interaction. In *Proceedings of ESWC*, pages 106–120.

Paolo Ferragina and Ugo Scaiella. 2010. Tagme: on-the-fly annotation of short text fragments (by wikipedia entities). In *Proceedings of CIKM'10*, pages 1625–1628, New York, NY, USA. ACM.

T. Joachims. 1999. Making large scale SVM learning practical.

E. Kaufmann and A. Bernstein. 2007. How useful are natural language interfaces to the semantic web for casual end-users? In *The Semantic Web*, volume 4825 of *LNCS*, pages 281–294. Springer.

E. Kaufmann, A. Bernstein, and R. Zumstein. 2006. Querix: A natural language interface to query ontologies based on clarification dialogs. In *Proceedings of ISWC*, pages 980–981. Citeseer.

D. Klein and C.D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of ACL*, pages 423–430. ACL.

Taku Kudo. 2005. CRF++: Yet another CRF toolkit.

J. Lafferty. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICML, 2001*.

X. Li and D. Roth. 2002. Learning question classifiers. In *Proceedings of ACL*.

X. Li. 2010. Understanding the semantic structure of noun phrase queries. In *Proceedings of ACL*, pages 1337–1345.

M.F. Porter. 1980. An algorithm for suffix stripping. *Program: electronic library and information systems*, 14(3):130–137.

S. Quarteroni, A. V. Ivanov, and G. Riccardi. 2011. Simultaneous dialog act segmentation and classification from human-human spoken conversations. In *ICASSP*, pages 5596–5599.

Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. YAGO: a core of semantic knowledge unifying WordNet and Wikipedia. In *Proceedings of WWW '07*, pages 697–706.

C. Wang, M. Xiong, Q. Zhou, and Y. Yu. 2007. Panto: A portable natural language interface to ontologies. In E. Franconi, M. Kifer, and W. May, editors, *The Semantic Web: Research and Applications*, volume 4519 of *LNCS*, pages 473–487. Springer.

[13] the rho parameter was empirically set to 0.1