A Comparative Analysis: QA Evaluation Questions versus Real-world Queries

Johannes Leveling

Centre for Next Generation Localisation
School of Computing
Dublin City University
Dublin 9, Ireland

Abstract

This paper presents a comparative analysis of user queries to a web search engine, questions to a Q&A service (`answers.com`), and questions employed in question answering (QA) evaluations at TREC and CLEF. The analysis shows that user queries to search engines contain mostly content words (i.e. keywords) but lack structure words (i.e. stopwords) and capitalization. Thus, they resemble natural language input after case folding and stopword removal. In contrast, topics for QA evaluation and questions to `answers.com` mainly consist of fully capitalized and syntactically well-formed questions. Classification experiments using a naïve Bayes classifier show that stopwords play an important role in determining the expected answer type. A classification based on stopwords is considerably more accurate (47.5% accuracy) than a classification based on all query words (40.1% accuracy) or on content words (33.9% accuracy). To simulate user input, questions are preprocessed by case folding and stopword removal. Additional classification experiments aim at reconstructing the syntactic wh-word frame of a question, i.e. the embedding of the interrogative word. Results indicate that this part of questions can be reconstructed with moderate accuracy (25.7%), but for a classification problem with a much larger number of classes compared to classifying queries by expected answer type (2096 classes vs. 130 classes). Furthermore, eliminating stopwords can lead to multiple reconstructed questions with a different or with the opposite meaning (e.g. if negations or temporal restrictions are included). In conclusion, question reconstruction from short user queries can be seen as a new realistic evaluation challenge for QA systems.

1. Introduction

User queries to search engines usually consist of 2-3 words (Spink et al., 2001; Teevan et al., 2006) and rarely are formulated as full sentences or questions (Ozmutlu et al., 2003). Query processing for information retrieval (IR) systems typically involves transforming the original user query by successively applying case folding, stopword removal, and stemming. Thus, user input to search engines already resembles results from query processing (as illustrated in Table 1) in that it typically lacks capitalization and stopwords, but still contains full word forms. Provided that most users are accustomed to web search engines but not familiar with QA systems, or that users mistake QA systems for information retrieval systems, they will try to formulate requests to QA systems as short keyword queries.

A comparative analysis of queries and questions investigates differences between queries (i.e. user input to search engines or preprocessed questions after case folding and stopwords removal) and questions (i.e. full natural language requests to QA systems). Different aspects of queries and questions are analyzed, including average length, case information, occurrence of stems and full word forms, wh-words (interrogative words), and sentence delimiters. The comparison aims at finding differences and similarities between QA questions and real-world queries.

The analysis demonstrates that much of the information present in full natural language questions is missing in short user queries. Thus, natural language processing tasks for QA, such as determining the expected answer type, cannot be performed as reliable as for full questions. However, a simple classification experiment illustrates that part of a question (the syntactic frame including the wh-word) can be correctly generated for 25.7% of the queries, despite of problems such as ambiguous queries.

2. Related Work

Brown and Coden describe an approach to reconstruct capitalization in text, trained on news stories (Brown and Coden, 2002). Their system assumes full punctuation of text. They infer that any word that does not appear in their capitalization dictionary (i.e. out-of-vocabulary) is most likely a proper noun and should be capitalized. Their best approach is based on capitalization dictionaries, phrases and other context information such as punctuation and achieves a precision of 90.3% and recall of 88.2%.

Gravano, Jansche et al. try to recover capitalization and punctuation in automatic speech transcripts using an n-gram language model (Gravano et al., 2009). Experiments are based on 1989 Wall Street Journal corpus and Broadcast News and show that using larger training corpora improves performance, but increasing the gram size from 3 to 6 does not. They assume that at most one punctuation symbol can occur between two words and use a limited set of punctuation characters (e.g. quotation marks are excluded).

Edmonds investigates lexical choice (Edmonds, 1997). He uses lexical co-occurrence networks based on mutual information and significance scores to fill word gaps with the most typical synonym. The system was trained on the part-of-speech tagged 1989 Wall Street Journal. Results show that including second-order co-occurrences improve performance of the system.

In summary, the reconstruction of punctuation and capitalization has been researched in automatic speech recognition and machine translation (MT) (Brown and Coden, 2002; Gravano et al., 2009; Huang and Zweig, 2002), but typically focuses on processing full text (e.g. news stories or automatic speech transcripts) instead of short queries. In addition, most of the research so far has ignored that stopwords and interrogative words are much more important in QA than in IR. For example, the wh-word (interrogative word) is an important feature in determining the expected answer type (EAT) of a question and *full* natural language questions are required if a QA system builds on deep syntactic-semantic parsing of questions or on other complex NLP methods. However, short user queries seldom contain interrogative words (cf. Table 2).

Approaches to finding questions describing information needs are also realized in systems for FAQ (frequently asked questions) search. Instead of trying to reconstruct questions from user input, the input is compared to questions in a question collection to find similar ones. Commercial solutions such as `q-go.com`[1] focus on closed domains (typically a single web site) which limits the type and number of possible questions. Using syntactic and morphologic information, user queries are mapped to sets of possible questions and multiple alternatives are presented to the user. For the open domain, these approaches would require a huge number of previously entered questions.

Spink, Wolfram et al. analyzed Excite query logs with more than 1 million queries and found that the average query length is 2.4 words (Spink et al., 2001). They also find that among the top 75 frequent terms used in queries, many are no-content terms (e.g. *'and'*, *'of'*, *'the'*, *'in'*, *'for'*, *'+'*, *'on'*, *'to'*, *'or'*, *'&'*, *'a'*). Similarly, Teevan, Adar et al. found that the average query length to search engines is 2.7 words (Teevan et al., 2006).

Leveling manually annotated the MultiNet Bibliographic Query Corpus, which consists of 12.946 user questions to a German natural language interface (NLI) to information providers on the internet (Leveling, 2006). 28.2% of the annotated queries contain some form of error, including wrong capitalization and spelling errors. Also, users were observed to formulate longer queries compared to search engines once they found out that the system can process full sentences and questions. As in web search, users of the NLI often enter one or two-word queries, confusing the NLI with a keyword-based search engine. Some other users entered much longer requests, similar to a dialogue with a human librarian. In contrast to web search, the natural language questions to this NLI contain 7.58 words on average. Using the query collection as a test set, structured database queries could be generated even for short queries or malformed information requests, using an automatic classification of terms. This approach increases the number of correctly transformed queries by about 30%.

Clearly, there is a gap between real-world user queries and questions used in evaluation campaigns such as TREC[2] or CLEF[3]. For example, questions and queries used in evaluation campaigns are typically grammatically well-formed, but user queries (e.g. in search engine logs or mailing lists) are not necessarily.

3. Analysis of questions and queries

3.1. Corpora

Six corpora containing queries, questions, and sentences were analyzed.

1. The question collection from Webclopedia (Hovy et al., 2000), a question answering system which has been evaluated at TREC. This collection originates from `answers.com`[4], a commercial Q&A service, providing answers to user questions. The hierarchical Webclopedia question typology (Hovy et al., 2002) was developed on an annotated set of questions from this corpus.[5]

2. The Excite log (Excite) of user queries, as distributed in Pig[6]. Pig is a software tool for analysis of large data sets and query logs and is being developed as an open source project under the Apache Software Foundation.

3. The Wikipedia article names[7] (titles) of the English Wikipedia.

4. The English 1 million sentence corpus from the Leipzig corpus collection[8], which contains samples from newspaper articles (EN1M). This resource originates from newspaper articles and has been collected for co-occurrence analysis (Quasthoff et al., 2006).

5. More than 2300 questions from the main TREC question answering track (see, for example (Voorhees and Tice, 2000)).

6. The combined English questions from the multi-6, multi-8, and multi-9 corpora (short: multi-X). Parts of these corpora have been used for official QA evaluation at CLEF QA 2003-2006 (see, for example (Magnini et al., 2006)).

The following processing steps were carried out to determine if a word form is a base form (stem): The Porter stemmer (Porter, 1980) was applied to the words in the text. If the stemmed result is equal to the input, the word is presumed to be the base form. Note that this approach is only a heuristic, because overstemming or understemming might produce results different from a correct base form. Also, stemming may result in words resembling stopwords.

For the data analysis, all text was tokenized by splitting at special characters (e.g. underscore, ampersand, brackets, the at-sign, etc.) and punctuation symbols (i.e. *','*, *';'*, *'?'*, *'!'*, *':'*, *'.'*, *'-'*). Following this tokenization method, URLs are not recognized as a single token, but are split into several tokens, including words (e.g. *'http'* and special characters (e.g. *':'*).

3.2. Queries and Questions

Results of the analysis of this data are shown in Table 1, confirming that the average length of user queries (in column Excite) is 2–3 words (Spink et al., 2001; Teevan et al., 2006). In addition, the following observations have been made: User queries (Excite) rarely contain stopwords,

[1] `http://www.q-go.com/`

[2] `http://trec.nist.gov/`

[3] `http://www.clef-campaign.org/`

[4] `http://www.answers.com/`

[5] `http://www.isi.edu/natural-language/projects/webclopedia/Taxonomy-data/WH_Analysis.html`

[6] `http://hadoop.apache.org/pig/`

[7] `http://dumps.wikimedia.org/`

[8] `http://corpora.uni-leipzig.de/`

punctuation symbols, or uppercase words in comparison to the full sentence corpus. Special characters (e.g. quotation marks or '-') often indicate web queries with special syntax, e.g. a phrase search or exclusion of terms. Wikipedia article names contain an even higher proportion of capitalized words, but capitalization occurs in expected places, e.g. at the beginning of sentences. Thus, the percentage of capitalized words is much higher in comparison with corpora containing full sentences. Users still enter full words forms as query terms for a web search (52.9% stems, 47.1% non-stems for Excite), assuming that the search engine will handle morphological variation or exact matching of query terms. Contrary to expectation, short user queries still contain full word forms, but the corresponding values are much higher for full sentences, for evaluation questions, and for questions to `answers.com` (67.1%-76.6%).

The analysis shows that case information and stopwords are mostly missing in web queries. This is not the case for questions from evaluation benchmarks such as TREC QA or QA@CLEF, and for questions to `answers.com`, where queries are typically well-formed, because syntax and orthography are expected to be correct or because malformed questions may prove expensive.

The following conclusions can be drawn from the analysis: Real-world user queries are short, contain few stopwords and lack capitalization. Thus, natural language processing (NLP) tasks such as part-of-speech (PoS) tagging, named entity recognition (NER) and classification, or parsing of user queries will most likely fail. Typically, these tasks are solved by approaches employing statistical methods trained on large corpora consisting of syntactically correct sentences. For example, in contrast to full text, short queries will contain different n-grams and no real syntactic structure because stopwords are missing. Furthermore, missing capitalization makes named entity recognition more difficult because proper nouns are not capitalized. In the EN1M corpus, only a small fraction of all sentences are questions. That means that part-of-speech tagging for queries and questions will be difficult even if a tagger is trained exclusively on questions extracted from a larger corpus (because of the smaller training set). Annotated corpora consisting of user queries are still too small to be useful in practice or are not yet available to the research community.

While the query logs provided with the Pig tool may seem a bit dated, user queries for web search engines do not seem to change over long periods of time. Silvestri (Silvestri, 2010) presents comparative statistics for query logs from Altavista and Excite from 1997-2002 which are based on experiments by Spink et al. (Spink et al., 2002). He observes that query behaviour has not changed from a statistical point of view over a period of four years and shows that query characteristics such as the number of terms per query vary only slightly or remain unchanged over time.

3.3. Identifying questions in query logs

The following characters were defined as sentence delimiters: '?', '!', '.', ';', '"', ')', ']', and '}'. Interrogative words of the following types were considered as wh-words to identify questions (here denoted by tags from the CLAWS tagset, (Garside et al., 1997)): AVQ (wh-adverb, e.g. *'when'*, *'how'*, *'why'*), DTQ (wh-determiner, e.g. *'whose'*, *'which'*), and PNQ (wh-pronoun, e.g. *'who'*, *'whoever'*).

The list of wh-words was compiled from a tagged subset of the British National Corpus (BNC). Therefore, some spelling mistakes and contracted forms of interrogative words are also included. Table 2 shows the top 10 most frequent words, wh-words, and sentence delimiters for the corpora.

The use of question marks as sentence delimiters indicates that most topics in QA evaluation and in questions from `answers.com` constitute proper questions. In some cases, the entries are multiple choice queries and answers were provided together with the question. In these cases, the user input does not end with a question mark.

Many questions to `answers.com` take the form of a natural language question, but some requests are formulated as imperative sentences or simple statements. In rare cases, imperative forms of verbs indicate a request for information (e.g. *"give ..."*, *"find ..."*, *"list ..."*), i.e. the queries should end with an exclamation mark (but typically do not).

The Excite log contains several special characters among the top-ten (most frequent) terms (e.g. ':', '/', '+', and '.'). These are artefacts from splitting up URLs into several tokens and from special operators used in search engines to denote the inclusion or exclusion of terms. Thus, these special characters appear among the top frequency terms.

Another experiment aims at identifying natural language questions in the Excite query log by looking for wh-words in the first five terms of a question and for a question mark in the last three tokens. If any of these is found, the entry is flagged as a potential question. In the Excite log with about 1 million queries, less than 5000 entries were found to be questions. The most frequent type of question observed are *"how to"*-questions (e.g. *"how to write a resume"*). This type of question is difficult to answer even for automatic QA systems. However, this asserts that some users seem to be looking for answers to this kind of question (which can be answered reasonably well by providing a web page).

3.4. Duplicate questions and ambiguity

After case folding and stopword removal have been applied to the questions from `answers.com`, duplicate questions were identified and their annotated classes (EAT) compared. To test if two queries are duplicates, they are represented as sets of content words Q_1 and Q_2. If $Q_1 / Q_2 = Q_2 / Q_1 = \emptyset$, the queries are regarded as duplicates, i.e. if they consist of the same content words. If two duplicates are tagged with different classes, this indicates either that a) the annotation was inconsistent or b) a possible question reconstruction is ambiguous, because more than one syntactic frame can be generated for queries with the same content words.

For example, the single word input *"berlin"* might be transformed into *"Where is Berlin?"* or *"What do you know about Berlin?"*; and vice versa: both of the latter queries are reduced to the single word query *"berlin"* after preprocessing. The detection of duplicate queries shows that 773 questions out of 22223 (3.5%) are duplicates.

Alternative interpretations of short user queries (ambiguity)

Table 1: Analysis of English corpora and topics.

		Excite	EN1M	Wikipedia	TREC	multi-X	answers.com
type		User queries	Sentences	Titles	Questions	Questions	Questions
entries		0.94M	1M	7.18M	2393	2580	35287
tokens		2.45M	25.1M	23.3M	20381	23238	381482
avg. length		2.6	25.1	3.2	8.52	9.00	10.81
uppercase	[%]	0.7	13.8	66.6	27.4	31.4	23.6
lowercase	[%]	81.8	70.6	17.7	58.0	53.6	61.7
numeric	[%]	4.9	2.1	2.4	0.4	1.7	1.1
punctuation	[%]	6.8	11.2	5.3	13.8	13.1	13.1
special	[%]	5.8	2.3	7.9	0.2	0.3	0.5
stopwords	[%]	7.8	49.0	11.7	53.4	51.9	53.3
non-stopwords	[%]	92.2	51.0	88.3	46.6	48.1	46.7
stems	[%]	52.9	28.5	8.3	30.6	23.4	32.9
non-stems	[%]	47.1	71.5	91.7	69.4	76.6	67.1

might be resolved by a simple popularity vote, i.e. using web search engines to obtain the frequencies of different questions via an exact search and selecting the most frequent (i.e. the most popular) alternative. However, this approach will not work for questions aiming at recent events because web search engines have to be updated regularly and modified content is indexed and available with some delay. Furthermore, users may actually mean the less popular interpretation because otherwise a simple web search might suffice to fulfil the information need.

In conclusion, generating a single question from short user input may not increase user satisfaction if the question can not be generated correctly. Instead, different questions should be suggested to the user for selection. Selecting full questions from a set of alternatives shown in the QA system interface might also help alter the user behaviour faster if the user learns that a QA system accepts or expects full natural language input.

4. Question Reconstruction

4.1. A simple approach to reconstructing the syntactic wh-word frame

To obtain a full natural language request for a QA system from a user query, the syntactic wh-word frame has to be created. The wh-word frame is defined as the longest stopword sequence at the start of a question, which is used as a class label in the following experiments. For example, the query *"capital Ethiopia"* is missing the wh-word frame *"what is the"*, the preposition *"of"* between *"capital"* and *"Ethiopia"*, and a trailing question mark to form the question *"What is the capital of Ethiopia?"*. There may be more than one correct wh-word frame, e.g. *"boston tea party"* could mean *"When was the Boston Tea Party?"*, *"Where was the Boston Tea Party?"*, or even – assuming a changed word order – *"Which party in Boston makes tea?"*. The corresponding wh-word frames for these examples are *"when was"*, *"where was"*, and *"which"*.

A trivial method for question reconstruction is to add a single generic syntactic frame *"Find information about ..."* and a single type of stopword (i.e. *'AND'*) to the user query to form a full request. This default approach works reasonably well for user queries containing a single word or proper nouns. Multiple content words can be connected by adding the word *'AND'* between them. For example, the query *"violence schools"* would be transformed into *"Find information about violence and schools"*.

However, this approach creates only general requests for *information* on a topic, not specific questions. The EAT for this type of question is overly generic and all reconstructed questions will be associated with the same EAT, i.e. this type of questions would be more suitable for a web search engine, not for a QA system. In a real-world QA scenario, users may be more interested in specific aspects of a topic. Furthermore, adding *'AND'* between all words also breaks up multi-word expressions and multi-word names (e.g. *'AND'* should not be inserted between *"New"* and *"York"*). Hence, a non-trivial solution for question reconstruction is needed.

In this paper, question reconstruction focuses on finding the wh-word frame given a case-folded query after stopword removal (which simulates the user query). This task seems to be similar to finding the expected answer type for QA, but there are some important differences: In contrast to finding the EAT, PoS information, named entity tags, and even capitalization information is not reliable or available for short queries. For instance, part-of-speech taggers are typically trained on an annotated corpus with full sentences (which contains fully capitalized words and stopwords, see EN1M in Table 1). Tagging will not be accurate if properties of the input (user queries) do not match properties of the training data. In addition, the word ordering may be different from the order in the final (or intended) question.

Table 3 shows results of classification experiments using a naïve Bayes classifier to determine the expected answer type and the wh-word frame. The training data consists of the questions from `answers.com`, together with their annotated expected answer type (EAT, called *qtarget* in Webclopedia) from the taxonomy used in the Webclopedia QA system. The question collection from `answers.com` was processed by filtering out entries missing an EAT and cor-

Table 2: Analysis of English corpora and topics.

	Excite	EN1M	Wikipedia	TREC	multi-X	answers.com
Top-10 words	*’NUM’*	*’the’*	*’(’*	*’the’*	*’the’*	*’the’*
	’+’	*’,?’*	*’NUM’*	*’what’*	*’is’*	*’what’*
	””	*’of’*	*’of’*	*’is’*	*’what’*	*’is’*
	’/’	*’to’*	*’-’*	*’of’*	*’in’*	*’of’*
	’and’	*’NUM’*	*’,’*	*’in’*	*’of’*	*’in’*
	’-’	*’a’*	*’the’*	*’was’*	*’who’*	*’a’*
	’of’	*’in’*	*’in’*	*’who’*	*’was’*	*’was’*
	’,’	*’and’*	*’and’*	*’how’*	*’which’*	*’who’*
	’the’	*’-’*	*’List’*	*’did’*	*’NUM’*	*’,’*
	’:’	*””*	*’de’*	*’a’*	*’did’*	*’NUM’*
Wh-words						
’what’	472	16322	3315	1286	873	18422
’how’	1453	9090	2112	304	312	2929
’when’	67	29968	1666	202	179	515
’who’	326	47291	4203	297	549	4748
’where’	276	12432	1239	161	185	1494
’which’	33	48545	288	63	385	927
’what’s’	22	836	555	23	3	2255
’why’	94	3161	740	8	1	846
’whom’	2	1324	77	6	5	42
’where’s’	3	34	163	1	0	54
’who’s’	45	356	398	1	2	204
’whose’	1	4208	93	1	6	159
’howe’	38	88	597	0	0	0
’whatsoever’	10	92	0	0	0	0
’whatever’	10	917	213	0	0	6
’wherever’	1	135	33	0	0	0
’how’s’	0	7	0	0	0	1
other	942095	825194	7165622	39	80	2685
Delimiters						
’?’	391	5218	4048	2353	2486	33938
’.’	4471	970144	43158	35	90	523
’!’	45	250	8959	0	0	8
””	44929	22054	7698	3	1	315
”’	1601	2334	37339	0	0	8
’:’	264	0	83	0	0	0
’)’	151	0	743051	0	0	22
’]’	3	0	0	0	0	1
other	893093	0	6336978	1	3	472

recting spelling errors in the class labels. Disjunctions of EAT were resolved by using only the first question annotation. There are 130 EATs used in the annotated questions. The data was divided into a training set containing 22223 questions (about 90%) and a test set containing the remaining 2222 instances.

Using lower case words after stopword removal as classification features, a naïve Bayes classifier was trained on the annotated questions. For the first three classification experiments, the class to be determined is the EAT. The annotated questions are associated with 130 classes, which correspond to the fine-grained hierarchical taxonomy of answer types used in Webclopedia (Hovy et al., 2002; Hovy et al., 2000). For the final experiment, the stopword sequence at the beginning of the original question (the wh-word frame) was used as a class label. This type of classification will automatically determine the syntactic wh-word frame of a question via the class label.

The results for the first three experiments seem to indicate that correct classification may rely largely on present stopwords in the question. The classification experiment using stopwords only as features achieves a much higher accuracy (47.5%) than EAT classification based on all words (40.1%) and on content words only (33.9%). Some natural language processing tasks for QA rely on stopwords (e.g. n-gram models), and if stopwords or other information is missing from input to a QA system (as was observed for web search queries), NLP processing will likely show degraded performance compared to applying the same method on full sentences or questions.

The final classification experiment investigated if the syntactic frame of a question can be generated for short user input. The user input (which is to be classified) is simulated by case folding full natural language questions to lower case and removing all stopwords. For this experiment, a much lower accuracy has been observed, compared to classification of EAT. However, the number of classes corresponds to the different surface realizations of the wh-word frame in a question (2096 classes compared to only 130 classes for classifying the EAT), which makes a classification much more difficult. Therefore, the results of this baseline experiment seem promising: in almost 26% of all simulated queries, the correct wh-word frame can be generated to form a full natural language question from only partial information.

Improvements for this approach are obvious, but may be difficult to realize: Using additional information such as the part-of-speech or named entity class of words in the input will help to improve accuracy. However, this information may not be obtained with high accuracy from lower case keywords. There is no need to exactly reproduce the wh-word frame of the original question. There may be different paraphrases of the same question expressing the same meaning. Currently, paraphrases of the wh-word frame are counted as errors. Recovering capitalization, missing stopwords (and possibly full word forms) will help to create a full natural language question which can be used in QA systems as a replacement for the terse original user input. However, the *full* query reconstruction is beyond the scope of this paper.

4.2. Discussion

Do user queries contain enough information to reconstruct a full natural language question? There are many problems making the task of question reconstruction a difficult one. So far, little research has investigated the problems that arise from reconstructing full natural language questions from partial information.

Ambiguous input. The user input *"bush fire sydney"* (after stopword removal and case folding) can be transformed into different questions, e.g. *"Will Bush fire Sydney?"* and *"Are there any Bush fires near Sydney?"*. Note that these questions have a different EAT and will be treated differently by QA systems, i.e. QA systems are expected to generate different answers. Without additional knowledge on the user, domain, or document collection, this ambiguity cannot be resolved.

Question paraphrases. There often are several possible alternatives which can be reconstructed for a given input. For example the query *"food lions"* could imply the intended question *"How much food do lions eat?"* or *"What food do lions eat?"*. In both cases, a verb which is closely related to the query topic has to be added (*'eat'*).

Word class conversion. User queries often contain nouns instead of adjectives or adverbs. For example, the query *"height Bruce Willis"* may have to be reformulated as *"How tall is Bruce Willis?"* instead of *"What is the height of Bruce Willis?"*.

Question types (overspecific or underspecific). Yes-no questions are mainly implied by stopwords, e.g. *"Has it ever snowed in Miami, FL?"* or *"Is a cello larger than a viola?"* are difficult to create from the user input *"snow Miami"* and *"cello (larger) viola"*. Instead, a more specific question might be reconstructed, e.g. *"When did it snow in Miami, FL?"*. In contrast, users may also be interested in general information about a subject (e.g. *"What do you know about artificial intelligence?"*).

Converse, contrastive or negated meaning of resulting question. The proposed approach for query reconstruction will have some limitations inherited from traditional IR: negations and contrary meanings can be reconstructed accurately only for the most frequent cases and will have to be ignored otherwise. Similar problems arise from temporal or spatial restrictions (e.g. *"without work"* vs. *"with work"*; *"hotels in X"* vs. *"hotels outside of X"*; *"X after 1995"* and *"X before 1995"*).

Negation expressed by stopwords can not be recovered at all (e.g. *'non'*, *'no'*) if these words do not occur in a query. Users interested in exact or specific answers will likely try to explicitly express this and include some form of negation in their query.

Complex questions. One assumption that is often made is that user input consists of a single sentence. However, questions from `answers.com` show that sometimes the answer is already contained in the question (e.g. *"Is Bill Clinton a lefty or righty?"* , *"True or false: ..."*). The question *"Do people only use 10% of their brains? If so, why?"* also shows that user questions can be more complex. In this example, the assumption of a single query does not hold. Two questions are asked, which are associated with different answer types: Y:N (yes-no-question) and REASON (reason explanation).

Fortunately, most of these linguistic phenomena (e.g. negation) are outside the scope of state-of-the-art QA systems. In conclusion, a first approach at full question reconstruction should follow the principle of Occam's razor and assume that the simplest question that can be constructed is the question intended by the user.

5. Conclusions and Outlook

The major findings from the analysis and experiments described in this paper are: i) User queries to search engines lack capitalization and stopwords. In properties such as average length, they are most similar to questions after case folding and stopword removal. In contrast, questions to `answers.com` and questions in QA evaluations are mostly formulated as full natural language questions.

ii) Given that many users are accustomed to web search, but few know QA systems and their capabilities, user behaviour (and queries) for QA systems can be presumed to be similar to web search. Question reconstruction will help to improve the input to QA systems until users have adapted to QA systems. If this assumption is valid, questions from QA evaluation campaigns do not adequately represent the challenge of question answering in the "real world" (e.g. in mailing lists or web fora). Future QA research should include question reconstruction from short user queries, which poses

Table 3: Classification experiments for expected answer type and wh-word frame.

Experiment	Features	# Classes	Correct	Incorrect
EAT	all words	130	892 (40.1%)	1330 (59.9%)
EAT	stopwords	130	1055 (47.5%)	1167 (52.5%)
EAT	non-stopwords	130	754 (33.9%)	1468 (66.1%)
wh-word frame	non-stopwords	2096	570 (25.7%)	1652 (74.3%)

challenges including resolving the ambiguity of restored questions and handling negation.
iii) NLP tasks for short user queries will be more difficult, because in comparison to full natural language questions, important information for classification and other processing is missing.
Future work will include investigating part-of-speech tagging for short user queries to help constructing full questions and employing state-of-the-art approaches to classification (e.g. support vector machines).

6. Acknowledgments

This research is supported by the Science Foundation Ireland (Grant 07/CE/I1142) as part of the Centre for Next Generation Localisation (CNGL) project.

7. References

Eric W. Brown and Anni R. Coden. 2002. Capitalization recovery for text. In Anni R. Coden, Eric W. Brown, and Savitha Srinivasan, editors, *Information Retrieval Techniques for Speech Applications*, volume 2273 of *LNCS*, pages 11–22. Springer.

Philip Edmonds. 1997. Choosing the word most typical in context using a lexical co-occurrence network. In *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics*, pages 507–509, Morristown, NJ, USA. ACL.

Roger Garside, Geoffrey Leech, and Anthony McEnery. 1997. *Corpus annotation: Linguistic Information from Computer Text Corpora*. Longman, London, New York.

Agustin Gravano, Martin Jansche, and Michiel Bacchiani. 2009. Restoring punctuation and capitalization in transcribed speech. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2009)*, pages 4741–4744. IEEE.

Eduard Hovy, Laurie Gerber, Ulf Hermjakob, Michael Junk, and Chin-Yew Lin. 2000. Question answering in Webclopedia. In *Proceedings of the Ninth Text REtrieval Conference (TREC-9)*, pages 655–664.

Eduard Hovy, Ulf Hermjakob, and Deepak Ravichandran. 2002. A question/answer typology with surface text patterns. In *Proceedings of the second international conference on Human Language Technology Research*, pages 247–251, San Francisco, CA, USA. Morgan Kaufmann.

Jing Huang and Geoffrey Zweig. 2002. Maximum entropy model for punctuation annotation from speech. In *7th International Conference on Spoken Language Processing (ICSLP'02)*, pages 917–920.

Johannes Leveling. 2006. *Formale Interpretation von Nutzeranfragen für natürlichsprachliche Interfaces zu Informationsangeboten im Internet*. Der andere Verlag, Tönning, Germany.

Bernardo Magnini, Danilo Giampiccolo, Lili Aunimo, Christelle Ayache, Petya Osenova, Anselmo Peñas, Maarten de Rijke, Bogdan Sacaleanu, Diana Santos, and Richard Sutcliffe. 2006. The multilingual question answering track at CLEF. In Nicoletta Calzolari, Khalid Choukri, Aldo Gangemi, Bente Maegaard, Joseph Mariani, Jan Odjik, and Daniel Tapias, editors, *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC'2006)*, pages 1156–1163, Genoa, Italy, 22-28 May 2006.

Seda Ozmutlu, Huseyin Cenk Ozmutlu, and Amanda Spink. 2003. Are people asking questions of general web search engines? *Online Information Review*, 27(6):396–406.

Martin F. Porter. 1980. An algorithm for suffix stripping. *Program*, 14(3):130–137.

Uwe Quasthoff, Matthias Richter, and Chris Biemann. 2006. Corpus portal for search in monolingual corpora. In Nicoletta Calzolari, Khalid Choukri, Aldo Gangemi, Bente Maegaard, Joseph Mariani, Jan Odjik, and Daniel Tapias, editors, *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC'2006)*, pages 1799–1802, Genoa, Italy, 22-28 May 2006.

Fabrizio Silvestri. 2010. Mining query logs: Turning search usage data into knowledge. *Foundations and Trends in Information Retrieval*, 4(1-2):1–174.

Amanda Spink, Dietmar Wolfram, Major B. J. Jansen, and Tefko Saracevic. 2001. Searching the web: The public and their queries. *Journal of the American Society for Information Science and Technology*, 52(3).

Amanda Spink, Bernard J. Jansen, Dietmar Wolfram, and Tefko Saracevic. 2002. From E-Sex to E-Commerce: Web search changes. *Computer*, 35(3):107–109.

Jaime Teevan, Eytan Adar, Rosie Jones, and Michael A. S. Potts. 2006. History repeats itself: Repeat queries in Yahoo's query logs. In *SIGIR 2006: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, Washington, USA, August 6-11, 2006*, pages 703–704. ACM.

Ellen M. Voorhees and Dawn M. Tice. 2000. Building a question answering test collection. In *SIGIR 2000: Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Athens, Greece, July 24-28, 2000*, pages 200–207. ACM.