ISSN 1744-1986

The Open University

Technical Report N° 2006/01

Evaluation of the CLEF query interface

Catalina Hallett
Donia Scott
Richard Power

22nd February 2006

Centre for Research in Computing
Department of Computing
Faculty of Mathematics and Computing
The Open University
Walton Hall,
Milton Keynes
MK7 6AA
United Kingdom

http://computing.open.ac.uk

Evaluation of the CLEF query interface

Catalina Hallett, Donia Scott and Richard Power
Centre for Research in Computing
Open University
Milton Keynes, U.K.

Abstract

As part of the CLEF project, we have developed a method for allowing subject matter experts to pose complex queries to a database. The method makes use of natural language generation, whereby users compose queries through interaction with a dynamic text that is generated "on the fly". We present here some first-step evaluations that we have conducted with our modest pool of available testers, the numbers of which are constrained for practical reasons having to do with data security. The results allow us to draw some useful conclusions about the usability of the method, and its training requirements, for subjects that are representative of our "typical" intended end-users.

1 Introduction: the CLEF interface

The Clinical e-Science Framework (CLEF) aims at providing a data repository of well organised clinical histories, which can be queried and summarised both for biomedical research and clinical care. In this context, the aim of the query interface is to provide efficient access to aggregated data for performing a variety of tasks, e.g., assisting in diagnosis or treatment, identifying patterns in treatment, selecting subjects for clinical trials, monitoring the participants in clinical trials. The intended users of this service are clinicians, biomedical researchers, and hospital administrators. Our current domain is cancer; however, the framework in principle supports a wide range of clinical fields.

An analysis of free text queries written by medical professionals show that they are mostly very complex and often ambiguous. This makes the design of the query interface to the CLEF repository particularly difficult, since our users will need to construct complex queries containing conditional and temporal structures.

Our repository of clinical histories currently contains some 20000 records of cancer patients, includes codes such as SNOMED and ICD, and is implemented as a relational database that stores patient records modeled on the archetype for cancer developed at UCL [Kalra et al., 2001]. Accessing relational databases involves expressing queries in a language that is understood by the database management system (typically SQL). Direct SQL querying requires specialist knowledge of the both the query language and the structure of the underlying database, and – in the case of medical databases – usually also knowledge of precise medical terminology codes. It clearly would be counter-productive to require this additional level of technical expertise of the clinicians and biomedical researchers who want to access the CLEF repository.

Attempts to overcome this problem in user interfaces to medical databases have traditionally made use of graphical devices such as forms, diagrams, menus, or pointers to communicate to the user the information content of a database (e.g., KNAVE [Shahar and Cheng, 1999] and TrialDB [Deshpande et al., 2001]),

and research shows that they are much preferred over textual query languages such as SQL, especially by casual and non-expert users. Nevertheless, empirical studies have reported high error rates by domain experts using graphical modelling tools [Kim, 1990] and a clear advantage of text over graphics for understanding nested conditional structures [Petre, 1995].

However, it is also well-known that queries expressed in free natural language are sensitive to errors of composition (misspellings, ungrammaticalities) or processing (at the lexical, syntactic or semantic level). A further drawback of natural language interfaces to databases is that such systems normally understand only a subset of natural language, and it is not always clear to casual users which are the valid constructions and whether the lack of response from the system is due to the unavailability of an answer or to an unaccepted input construction. On the positive side, natural language is far more expressive than SQL, so it is generally easier to ask complex questions and manipulate temporal constructions using natural language than using a database language.

2 The CLEF query interface

The CLEF query system is designed to answer questions relating to patterns in medical histories over sets of patients in the data repository. The current interface is designed for casual and moderate users who are familiar with the semantic domain of the repository but not with its technical implementation (e.g., clinicians, medical researchers and hospital administrators). For the reasons we described above, the guiding principle in the design of our interface is that its use requires no prior knowledge of the structure of the repository, no expertise in database access languages such as SQL, no familiarity with medical codes, and only minimal prior training. Users' interaction with the CLEF repository is *not* through SQL, or graphics or free text. Instead, query-construction is performed by interacting with an automatically-generated Natural Language feedback text (currently only English). This interaction method, based on the WYSIWYM technology [Power and Scott, 1998], allows users of the profile described above to construct in an intuitive way, unambiguous, syntactically correct, complex natural language queries, such as:

3 Evaluation

The best evaluation of any Question-Answer system is one which looks at real users making information-seeking requests in real-life contexts. Since the complete CLEF system is not yet ready for deployment, this is not yet possible. However, it is possible to perform usability tests on the query interface in isolation from the full system, and this is what we report on here. Our current evaluation study does not cover the "query to SQL translation" and the "answer retrieval" components, which are part of the server side of the query interface. This separation is not always possible in practice. For example, we cannot at this stage test the full range of queries that can be constructed in the query interface, since some are not yet supported by the back-end. Similarly, we can only assess the time necessary for editing queries, not for retrieving answers, since this is almost entirely dependent on the communication procedure and on the speed of the SQL translator.

Of necessity at this stage, our evaluation study must be rather modest in scope — not only for the reasons described above, but also because of the limited availability of subjects. These limitations come from three sources. Firstly, to properly reflect our target users, subjects must have a strong medical background

with a good understanding of the treatment of cancer; this knowledge is also required to understand the queries. Secondly, due to the highly confidential status of the data in our repository, subjects must also have authorised clearance for accessing the CLEF repository. Thirdly, due to other demands on (and the cost of) their time, the upper limit on the time subjects could devote to the study was half an hour. These restrictions necessarily constrain the design of any experiments at this stage in the work. As a result, the results of the experiments that we have conducted so far are rather more suggestive than one would normally expect. However, they do provide us with useful insights on user's performance on, and reaction to, this rather unusual style of query interface.

We have thus far conducted three experiments, to address the following questions:

- are users able to successfully compose complex queries using the system?
- can the system be used with minimal training?
- are the queries, as presented in the interface, ambiguous?
- how does the querying method compare to more traditional methods, in particular, SQL?

3.1 Experiment 1: Query composition

The main desideratum behind the design of our querying method is that it should be intuitive: medics and bio-informaticians should be able to use it to pose the kind of complex queries that they require of a system like CLEF, without the need for extensive training on the interface, or knowledge of the structure or language of the underlying repository. This experiment tests the extent to which our querying method fulfills these requirements.

3.1.1 Subjects

Fifteen medics and bio-informaticians participated in the experiment. All had previously been granted clearance[1] to see the confidential repository of patient records. All subjects were knowledgeable on the domain of cancer, and all but two had no knowledge of the representation language of the repository, or of how the data contained therein was structured; none had any previous experience with the query-formulation interface.

3.1.2 Methodology

Each subject was given a short (5–10 minute) introduction to the interface, which included a demonstration of the construction of a fairly simple query. Subjects were then given a set of four queries, which they were asked to compose using the CLEF query interface. To increase the difficulty of the task, the questions presented to the subjects were expressed in a language as different as possible from the language in the query interface. To avoid effects of practice, we varied randomly the order in which the questions in the set are presented to subjects. Subjects were allowed as much time as they needed to compose each query.

For each subject, we measured the time taken to build each query, and recorded the number of operations used for constructing a query.

[1] by the UK Medical Research Ethics Committee(MREC).

3.1.3 Materials

The materials for the experiment consisted of the following set of four queries:

> How many patients who received surgical treatment for malignant neoplasm of the central portion of the breast had no curative radiotherapy?
>
> How many patients between the ages of 40 and 60 when they were first diagnosed with lung cancer *(malignant neoplasm of bronchus or lung, unspec)* received radiotherapy and had a platelet count higher than 300 and a leukocytes count lower than 3?
>
> What percentage of patients under the age of 60 treated for breast cancer *(malignant neoplasm of breast, unspec)* died within 5 years of a mastectomy?
>
> How many patients with acute lymphoid leukaemia have been given chemotherapy?

These queries are representative of the range of types of queries that emerged as desirable in an earlier requirements analysis with a group of oncologists and cancer bio-informaticians. They also vary in their levels of structural complexity and in the number of interface operations required to successfully complete them.

Note: these are far more complex than standard queries put to search engines or to most other interactive query engines (e.g., *cite lots of other studies*).

3.1.4 Results

All subjects were able to successfully complete all four queries. The mean completion time per query was 3.9 minutes (noting that subjects were under no time pressure to complete the individual queries).[2] Figure 1, which gives the average time to completion across all subjects, shows that subjects learned to use the interface very quickly: they took much longer on the first query, and their performance asymptotes by the time they get to the second query. This effect is statistically significant: Analysis of Variance[3] shows a highly significant effect of order of presentation ($F=9.8427$; $p<.0001$). Furthermore, significant differences were found between subjects performance on the first query they composed compared to the second, third, and the fourth (each at $p<.01$ on the Tukey HSD test). However, application of the same test showed no significant difference in subjects performance on the second *versus* third, second *versus* fourth, or the third *versus* fourth composed query.

Since the queries vary in structural complexity, some will necessarily require the user to perform more interface actions to achieve completion than others, and so one would predict a difference in subjects performance (i.e., time to completion) on the individual queries; this was borne out by the analysis (ANOVA, $F=5.5015$; $p<0028$). More importantly, one would also predict that subjects' proficiency with the interface will increase as they move from the first query they encounter to the last. Figure 2 shows subjects' performance on the interface in terms of the number of interface operations (clicks and selections) they performed, normalised for complexity: a value of 1 would mean that subjects performed twice as many operations as were required; a value of 0 would mean that subjects performed exactly the number of operations required (i.e., perfect performance). The picture that

[2] For the last 5 subjects, who used a version of the interface that had been improved to respond faster to interface actions, this average went down to 2.7 minutes.

[3] One-way ANOVA for correlated samples.

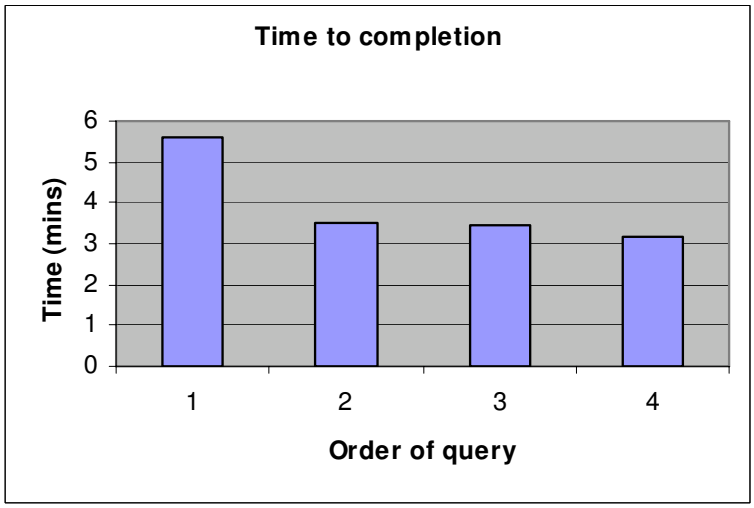

Figure 1: Mean completion time for queries in order of occurrence

emerges from this figure is one where overall, subjects are very efficient, achieving an average score of 0.19 over their first four encounters with the method. They make a fair number of false starts when composing their first query, but become extremely proficient by the time the get to their second query, and are near perfect by the time they get to the fourth. Analysis of Variance[4] shows a highly significant effect of order of presentation ($F=7.4993$; $p<.0004$). Once again, the Tukey HSD Test shows a significant difference between the first query encountered and each of the subsequent ones ($p<.01$) and that the differences between the second and third, the second and fourth and the third and fourth were nonsignificant.

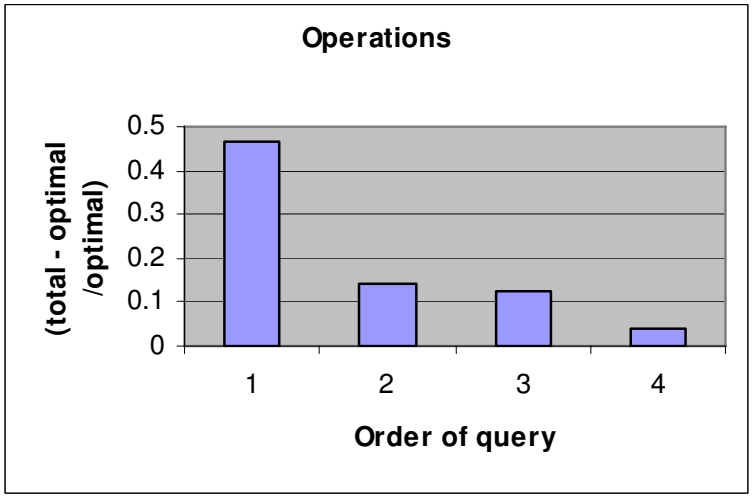

Figure 2: Proficiency with the interface as a function of experience

[4]One-way ANOVA for correlated samples.

3.2 Experiment 2: Clarity of the queries

A major hindrance to the success of natural language interfaces to databases, is that natural language is rife with ambiguity and imprecision, which makes the system's task of interpreting the user's queries particularly onersome. Our method of composing queries avoids this problem altogether: since the natural language feedback text is generated by the system itself, it is guaranteed to be completely unambiguous to the system. Of course, this is no guarantee that it is also clear to the user. We have tried to address this possibility in designing the feedback texts.

Our earlier requirements analysis revealed that queries of the type being posed to CLEF can be decomposed into four fields

Subjects: The particular characteristics of the patients of interest.

Treatments: The set of particular treatments of interest regarding that subset of patients.

Tests: The results of any particular tests.

Outcomes: The measure of interest for the particular intersection of subjects, treatments and tests.

The feedback text is structured to break the queries down into these explicit fields. For example, the query:

> What percentage of patients who have a family history of breast cancer, who are in the 35 – 65 age group and who suffered from severe weight loss, had a lumpectomy to both breasts and then two cycles of chemotheraphy followed by a radical mastectomy, and had low platelet count at some point, survived more that 10 years after the initial diagnosis.

would be set out in the feedback text as:

Relevant subjects: patients between the ages of 35 and 65, with a familial history of breast cancer, and who suffered from severe weight loss.

Treatment: a bilateral lumpectomy, followed by two cycles of chemotheraphy and then a radical mastectomy.

Tests: a haemotology test showing a low platelet count.

Outcome: the percentage of patients who survived more than 10 years after the initial diagnosis.

Of course, although this works to break down the complexity of the queries, it is still possible that the generated texts within the fields are unclear. This experiment explores the extent to which composed queries, as presented in the feedback texts, can be clearly understood.

3.2.1 Subjects

Fifteen subjects participated in the experiment. Of these, 10, had previously participated in Experiment 1; the new subjects had the same profile as those previously seen.

3.2.2 Methodology

Subjects were each given an paper-based questionaire containing 24 trails, each show a completed complex query as presented in the CLEF interface (i.e., as a WYSIWYM feedback text). Each such presentation was associated with three alternative interpretations, presented as full natural language questions. The subjects were given a forced-choice task to identify which of the three alternatives corresponded to the WYSIWYM feedback text. For each query presented, the alternatives solutions were (a) the intended meaning, (b) the unintended meaning and (c) a plausible but incorrect meaning.

The queries were presented to all subjects in the same (random) order. We devised five presentation sets, each containing a different ordering of the options for each query, and these were randomly assigned to subjects. We suggested to subjects that a useful strategy could be to read the alternatives before looking at the associated feedback text. There was no time limit to the experiment.

3.2.3 Materials

The materials comprised four examples each of six patterns of ambiguity:

attachment of temporal expression: Most events can have a temporal expression associated. When there is more than one even that could be subsumed by a temporal expression, the text may become ambiguous. For example:

> **Relevant subjects:** patients with a clinical diagnosis of breast cancer
> **Treatment:** patients who did not receive chemotherapy in the past year
> **Tests:** []
> **Outcome:** absolute number of patients
>
> Options:
> 1. How many patients diagnosed with breast cancer had no adjuvant chemotherapy in the past year?
> 2. How many patients treated for breast cancer in the past year had no adjuvant chemotherapy?
> 3. How many patients diagnosed with breast cancer in the past year had no adjuvant chemotherapy?

scope of conjunctions: Whenever a complex expression contains a combination of conjunctions and disjunctions, potential ambiguities may occur, especially when combined with negations or prepositional phrases. For example:

> **Relevant subjects:** patients with a clinical diagnosis of invasive ductal carcinoma
> **Treatment:** patients who received breast conservation surgery, no axillary surgery and radiotherapy
> **Tests:**[]
> **Outcome:** absolute number of patients
>
> Options:
> 1. How many patients diagnosed with invasive ductal carcinoma underwent breast conservation surgery, did not undergo axillary

surgery and received radiotherapy?
2. How many patients diagnosed with invasive ductal carcinoma underwent breast conservation surgery, did not undergo axillary surgery and did not receive radiotherapy?
3. How many patients diagnosed with invasive ductal carcinoma did not undergo breast conservation surgery, did not undergo axillary surgery and received radiotherapy?

scope of conjunctions plus attachment of temporal expression: This is an extension of the first two cases, where a temporal expression post-modifies an expression that is part of a conjunction of events. For example:

Relevant subjects: patients with a clinical diagnosis of malignant neoplasm, unspecified
Treatment: patients who received radiotherapy and chemotherapy within 1 year of the diagnosis
Tests:[]
Outcome:absolute number of patients

Options:
1. How many patients diagnosed with cancer had radiotherapy within 1 year of diagnosis and also had chemotherapy at any time?
2. How many patients diagnosed with cancer had radiotherapy and chemotherapy both within 1 year of diagnosis?
3. How many patients diagnosed with cancer had radiotherapy and chemotherapy and received any kind of treatment within 1 year of diagnosis?

combination of various query components: Events in a query can be linked to each other by various means, including temporal expressions, conjunctions and disjunctions. Complex combinations may render the feedback text ambiguous.For example:

Relevant subjects: patients with a clinical diagnosis of breast cancer and who had nausea within 1 year of the chemotherapy
Treatment: patients who received [some surgical procedure] [at some point in time] and chemotherapy but no radiotherapy within 1 year of the diagnosis
Tests:[]
Outcome: percentage of patients who were alive after 5 years of the diagnosis

Options:
1. How many patients diagnosed with breast cancer underwent a surgical procedure at any time, received chemotherapy at any time, had nausea at any time after chemotherapy and received no radiotherapy within 1 year of the diagnosis survived more than 5 years after diagnosis
2. How many patients diagnosed with breast cancer underwent a surgical procedure within 1 year of the diagnosis, received chemotherapy within 1 year of the diagnosis, had nausea at any time after chemotherapy and received no radiotherapy within 1 year

of the diagnosis survived more than 5 years after diagnosis
3. How many patients diagnosed with breast cancer underwent a surgical procedure at any time, received chemotherapy within 1 year of the diagnosis, had nausea after chemotherapy but within 1 year of the diagnosis and received no radiotherapy within 1 year of the diagnosis survived more than 5 years after diagnosis

complex queries, non ambiguous components: We introduced this category in order to test the readability of complex queries that do not necessarily contain ambiguous components. Since most queries in the medical domain are likely to be very complex, can the sheer number of query components render the query ambiguous to the users?. For example:

Relevant subjects: patients under the age of 50 at the time of diagnosis, with a clinical diagnosis of breast cancer
Treatment: patients who received [some surgical procedure] [at some point in time] and no chemotherapy within 1 year of the diagnosis
Tests: []
Outcome: absolute number of patients

1. How many patients with breast cancer, under the age of 50 had a surgical procedure within one year of the diagnosis and did not have chemotherapy within one year of the diagnosis
2. How many patients with breast cancer, under the age of 50 had a surgical procedure at any time and did not have chemotherapy within one year of the diagnosis
3. How many patients with breast cancer, below the age of 50 had a surgical procedure within one year of the diagnosis and had chemotherapy after one year of the diagnosis

attachment/interpretation of outcome: The outcome section generally describes a condition holding between a reference and a target set of patients. If the query contains multiple features describing the patient set, it may be difficult to differentiate between features that contribute to the reference set and features that contribute to the targe set. For example:

Relevant subjects: patients with a clinical diagnosis of breast cancer and who had anaemia after chemotherapy
Treatment: patients who received chemotherapy
Tests:[]
Outcome: percentage of patients who were alive after 5 years from the diagnosis

Options:
1. Out of the whole number of patients in the database, what percentage were diagnosed with breast cancer, developed anaemia after chemotherapy and survived 5 years after diagnosis
2. Out of the number of patients diagnosed with breast cancer, what percentage developed anaemia after chemotherapy and survived 5 years after diagnosis

3. Out of the number of patients diagnosed with breast cancer who developed anaemia after chemotherapy, what percentage survived 5 years after diagnosis
give example here

3.2.4 Results

If the presented WYSIWYM feedback text is unclear or confusing, the probability that subjects will select the correct interpretation will be 0.33 (i.e., they will get the right answer only a third of the time). Our results show that subjects' precision is 0.84: on average, they select the intended interpretation 84% of the time, rather than 33% as would be precidicted if their selections were random. Statistical analysis of these results, using a one-sample T-test, shows this effect to be highly significant (mean=0.8361, d=0.5028, t=16.76, p<.0001).

3.3 Exploratory comparisons with SQL

Having examined the usability of the CLEF query-interface, we now turn to its utility: is it easier to use than traditional database querying methods such as SQL? To properly test this, given the main desideratum mentioned earlier, it's not only necessary that our subjects be unfamiliar with CLEF interface, but they must also be highly proficient with SQL.[5] Additionally, subjects will have to have detailed knowledge of the structure, organisation and content of the CLEF database to be able to construct the SQL queries. Finally, subjects will also need to be familiar with the terminology codes (in this case, SNOMED and ICD) that are used in the database to refer to medical constructs. Because of these very strong constraints, our pool of subjects is obviously severely limited; indeed, at present there are only two people with these qualifications. As a result, our evaluation of this aspect of the CLEF interface can only be quantitative, and only indicative conclusions can be drawn from the results obtained. Clearly, the data gathered will not be amenable to statistical analysis. Nonetheless, they should provide valuable insights of the utility of the querying method.

Our two subjects had also participated in Experiment 1. Immediately following this, we asked them to compose in SQL one of the queries they had just composed with the WYSIWYM interface. To maximise their chances of successfully creating the SQL queries, they were given access to all the SNOMED and ICD codes required to build the SQL. They were therefore optimally advantaged for translating queries into SQL. The results of this test are shown in Figure 2.

The subjects each received different queries to compose. One was given the query

> How many patients with acute lymphoid leukaemia have been given chemotherapy?

He had previously successfully composed this query with the WYSIWYM interface in 2.3 minutes. He took 8.5 minutes to compose it in SQL; the resulting SQL query contained some errors.

The second subject was given the query

> How many patients between the ages of 40 and 60 when they were first diagnosed with lung cancer *(malignant neoplasm of bronchus or lung, unspec)* received radiotherapy and had a platelet count higher than 300 and a leukocytes count lower than 3?

[5] We expect that users with only intermediate knowledge of SQL would not be able to construct relatively complex queries that are required for CLEF.

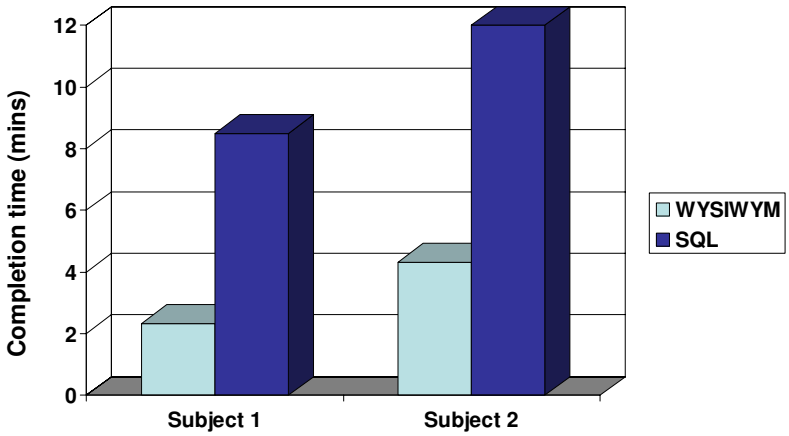

Figure 3: Performance with WYSIWYM *versus* SQL

He had previously successfully composed this query with the WYSIWYM interface in 4.5 minutes. After trying to compose it in SQL for approximately 12 minutes this subject gave up; at this point the SQL was almost complete, but he was frustrated and bored with debugging it.

4 Discussion and Conclusions

Ideally, a formal evaluation of this type would involve a large number of exemplars of each type of query supported by the system, and a large number of subjects. Given our constraints on the number of available subjects (and the concomintant effect this has on the possible design of any experiments), the evaluation reported here is necessarily more limited in scope, and the conclusions we can draw can, at this stage, only be preliminary. The picture that emerges from this study are nonetheless very encouraging. Our results suggest that subjects who are representative of class of users to which the system is aimed can, for queries that are representative of the types specified by this user community:

- use the WYSIWYM method to successfully compose complex queries, with no prior exposure to the method and with the benefit of only minimal training;
- become quickly proficient with the system, achieving near perfect performance by their fourth attempt at query composition.

Our evaluation also shows that the feedback text used to construct queries of a high degree of structural complexity are not difficult to understand. This is extremely important, as it means that users can be confident that the answers that they are getting are to the questions that they think they are asking, and not to some other similar question.

Finally, there is good reason to believe that the WYSIWYM method of query composition may be much more "user-friendly" than the traditional method of SQL, even for extremely skilled SQL coders with a high level of familiarity with the database and the domain. Our tests of this show that (an albeit small sample of) such experts, even in a situation that is heavily biased towards optimal performance of SQL codes, found it very much easier to compose queries with the WYSIWYM interface than in SQL. Not only did it take them more than three times longer, on average, to compose the query in SQL, but they were not able to produce the complete SQL in that time.

References

[Deshpande et al., 2001] Deshpande, A., Brandt, C., and Nadkarni, P. (2001). Ad hoc query of patient data: Meeting the needs of clinical studies. *Journal of the American Medical Informatics Association*, 9(4):369–382.

[Kalra et al., 2001] Kalra, D., Austin, A., O'Connor, A., Patterson, D., Lloyd, D., and Ingram, D. (2001). *Design and Implementation of a Federated Health Record Server*, pages 1–13. Medical Records Institute for the Centre for Advancement of Electronic Records Ltd.

[Kim, 1990] Kim, Y. (1990). *Effects of conceptual data modelling fomalsms on user validation and analyst modelling of information requirements*. PhD thesis, University of Minnesota.

[Petre, 1995] Petre, M. (1995). Why looking isn't always seeing: readership skills and graphical programming. *Communications of the ACM*, 38(6):33–44.

[Power and Scott, 1998] Power, R. and Scott, D. (1998). Multilingual authoring using feedback texts. In *Proceedings of 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics (COLING-ACL 98)*, pages 1053–1059, Montreal, Canada.

[Shahar and Cheng, 1999] Shahar, Y. and Cheng, C. (1999). Intelligent visualization and exploration of time-oriented clinical data. In *Proceedings of HICSS*, Maui, Hawaii.