Towards Genre Classification for IR in the Workplace

Luanne Freund[1], Charles L.A. Clarke[2], Elaine G. Toms[3]

[1]Faculty of Information Studies, University of Toronto, Canada
[2]School of Computer Science, University of Waterloo, Canada
[3]Faculty of Management, Dalhousie University, Canada
{luanne.freund@utoronto.ca, claclark@plg2.math.uwaterloo.ca, etoms@dal.ca }

Abstract. Use of document genre in information retrieval systems has the potential to improve the task-appropriateness of results. However, genre classification remains a challenging problem. We describe a case study of genre classification in a software engineering workplace domain, which includes the development of a genre taxonomy and experiments in automatic genre classification using supervised machine learning. We present results based on evaluation using real-life enterprise data from this work domain.

Keywords: contextual information retrieval, genre-dependent applications, genre classification, enterprise search

1 Introduction

Searching for information in the workplace can be daunting: in some cases more daunting than the task that prompted the search in the first place. The rapid increase in digital resources, corporate portals and intranets, together with the growth of the World Wide Web, means that employees working in large companies now have vast libraries of digital resources readily available to support their work activities. These “libraries” tend to be ad hoc and distributed collections of heterogeneous resources, lacking in unifying standards and quality controls [1]. Unlike the Web, where information is often pushed to the appropriate community, internal corporate information is more passively, or even begrudgingly shared [2]. Finding something useful under these conditions, requires that employees: a) know where to start searching, b) know how to search in that context, c) be capable of distinguishing a useful document when they see it, and d) have the time and patience to sort through a lot of useless material along the way. Needless to say, these are high demands on any employee and become particularly onerous in tight timeframes when information is mission critical.

Enterprise search system developers are focusing much attention on the technical challenges of searching across these distributed and heterogeneous information spaces. There are also ongoing efforts to develop systems that incorporate semantic tools to improve the topical relevance of search results [3]. However, relatively little attention has been paid to developing task-centric search tools capable of retrieving situationally relevant or useful search results [4]. This is surprising, given that workplace searching is primarily motivated by the need to complete work tasks. For example, recent studies of information seeking and searching of engineers indicate that some of the most important evaluation criteria for information are features such as: “importance to my work” [5], “appropriateness to task” [6], and “in the right format, and at the right level of detail” [7]. Our research with a group of software engineers supports these findings and shows that there is a strong relationship between “task-appropriateness” and genre [8], [9].

It follows that the integration of genre into task-based workplace search systems is likely to bring benefit; however, there are many open questions with respect to the implementation of such a system. Genre is a complex document characteristic and has not been studied to the same extent as text semantics for application to IR. This paper describes our work towards developing a workplace search system that uses relationships among different tasks and genres to filter the search results. In this paper, we focus on the basic issues of genre classification, which must be addressed in order to implement such a system. The research questions guiding this work are:

- Is it possible to identify a small set of core genres that can classify a significant portion of the documents used by this group?
- Are text-based automatic classification methods effective for genre? How well do they perform on real-world enterprise data?

2 Background

The role of tasks in motivating and framing information behaviour is receiving increasing attention in information science research [10], [11]. Research has shown that significant relationships exist between various characteristics of work tasks (time constraints, importance, stage of completion, complexity, etc.) and various measures of information seeking behaviour (selection of channels, number and type of sources consulted, cognitive activities, etc.) [10], [11], [12]. However, task-based approaches have had limited application in IR to date. One of the reasons that the relevant research has been difficult to apply to IR system design, is that is focuses primarily on the tasks, the task performers and their behaviour. This is a one-sided approach that is very difficult to apply to an essentially two-sided matching problem between searchers and documents.

The value of genre is that it has the potential to provide the other side of the equation: a task-centric approach to documents. The concept of genre, classification of things by types or styles, is ancient: it has been used to organize texts and other media for thousands of years. However, current genre theory has moved beyond classification to stress the functional role of genres in communication. From this perspective, genre can be defined as “a distinctive type of communicative action, characterized by a socially recognized communicative purpose and common aspects of form” [13]. Accordingly, genre classes are defined and recognized based on a mixture of elements of form, content and communicative purpose.

Document genres are a characteristic feature of organizational information use environments. Genre repertoires, sets of genres that are in common usage, exist within most work domains and serve to promote mutual understanding and make documents more recognizable to members of the group [13], [14]. Studies in the workplace indicate that genre serves as a context carrier; it can represent elements of the "who, what, where, when, why and how" of a document, which makes it particularly useful when people are trying to apply information in a real-world situation [15]. Current approaches to genre stress the situated and dynamic nature of genres, which emerge within a particular community, are adaptive to new conditions and may disappear over time.

Genre, like a range of other non-topical features of documents, has been underexploited in IR algorithms to date, despite the fact that we know that searchers rely heavily on such features when evaluating and selecting documents [16]. Genre is particularly valuable in this regard, as it provides visual cues in the "physical landscape of a document," and contains "distinctive, salient features that inform users about a document's identity" [16]. Recently, genre has been recognized as a possible means of supporting more targeted information retrieval through the identification of personal and task-base genre correlations and the implementation of techniques for the automatic identification of genres [17], [18], [19]. However, in practice, implementations of genre in information retrieval have been limited to providing the searcher with the option to either limit or cluster results according to genre [19],[20] rather than weighting genre in the matching or ranking algorithms.

One of the challenges of incorporating genre into IR systems is classification. Classification of texts by subject using a range of machine learning techniques is well established and has proven to be effective. In contrast, genre classification is still in the research stage, with very little application to real world data. The most successful approaches reported to date have used a large number of document features: textual, structural and linguistic [21], [22]; however, some promising results have also been obtained relying primarily on term frequencies [23], [24]. Classification approaches to web documents have typically included a wider range of features, drawing upon HTML mark-up; however, high rates of success have been achieved only for small sets of genre classes. Accuracy seems to be reduced when multiple subjects are represented in the corpus [25], when the corpus contains documents not belonging to any category [26], and when fine-grained classification schemes are used [19].

3 Study Framework

The setting for this study was the software services division of a large multi-national high-tech firm. Our target population was a physically distributed group of software engineering consultants, who provide a wide range of expert technical services to customers using the company's software products. The information environment in their work domain is composed of a mixture of resources from the Web, the company's public website, the corporate intranet, and a number of shared restricted-access databases and forums in use by members of the group. Genre is a strong characteristic of information in this environment, although no standard genre taxonomy is in use.

Our study of genre was conducted in two stages as part of a larger project to implement a contextual workplace search system for this group. The first stage consisted of a genre analysis, which used a combination of user- and document- centred methods to identify and define a genre taxonomy for this domain. The second stage focused on the application of the SVM Light tool to conduct genre classification experiments using the taxonomy developed in the first stage.

4 Genre Analysis

The goal of the genre analysis was to identify a core genre repertoire in use by this group. We were seeking a fairly stable genre set, small enough to be manageably implemented in a retrieval system, but with a scope broad enough to include a large portion of the documents in heavy use by this group. The objectives of the analysis were as follows:

- to identify the core genre repertoire of this work domain;
- to develop a standard taxonomy to represent it;
- to develop operational definitions of the genre classes in the taxonomy, including identifying features in terms of form, function and content to facilitate manual and automatic genre classification.

We used a bottom-up classification approach based on a study of the current information environment and commonly used genres rather than a formal or theoretical approach, in order to preserve the domain-specific, "situated" nature of genre use and of existing task-genre relationships.

4.1 Identifying the Genre Repertoire

In order to get a broad perspective on genre usage in this domain, we collected data from two sources: the user community and existing document repositories. From the user community, we extracted lists of genres mentioned by participants in the course of a focus group (7 participants) and private semi-structured interviews (14 participants). Both the focus group and the interviews focused on workplace information seeking and searching practices in general and all participants were software engineering consultants working in two different product groups (a more detailed description of the methods can be found in [9]). We then surveyed the over 40 websites and repositories in use in this domain, and found six that use genre taxonomies. After combining the list of user-identified genres with genres from the six repositories we had a list of 65 unique genres, more than half of which were found in only one repository or user-identified source. Some of these less common genres are specialized subgenres, such as *customer support plans*, while others, such as *sales kits*, are simply less frequently used in this work domain. Table 1 lists the 29 genres identified in more than one source, showing the distribution across sources and the total frequency of occurrence for each.

Table 1. Distribution of Common Genres by Source*

	Genres used in Repositories (R 1-6)						User-Identified Genres		
Genre Class	R1	R2	R3	R4	R5	R6	Interviews	Focus Group	Frequency
courses, training materials	X		X			X	X	X	5
manuals	X		X		X		X	X	5
presentations	X		X			X	X	X	5
product documents	X		X		X		X	X	5
technotes, tips	X				X	X	X	X	5
tutorials and labs	X		X			X	X	X	5
white papers	X		X		X	X	X		5
best practices	X		X				X	X	4
design patterns		X	X	X			X		4
discussions / forums			X		X		X	X	4
engagement materials	X		X				X	X	4
FAQs	X				X	X	X		4
websites	X		X				X	X	4
cookbooks & guides		X	X				X		3
demos	X		X				X		3
engagement summaries		X	X				X		3
problem reports					X		X	X	3
reusable assets		X	X				X		3
solutions				X	X		X		3
source code		X	X					X	3
technical articles	X		X				X		3
demos	X		X				X		3
downloads	X				X				2
flashes					X	X			2
lessons learned		X					X		2
product pages							X	X	2
roadmaps			X				X		2
templates		X					X		2
tools			X				X		2

*Genres are listed in order of decreasing frequency of occurrence across sources; genres selected for the final taxonomy are marked in bold.

4.2 Finding the Core Genres

The next step was to reduce the set to a smaller and more functional taxonomy, which could be used to classify the document collection using automatic methods. To guide selection, we relied primarily on genre prevalence, measured by the frequency of occurrence in Table 1. However, we took some additional practical considerations into account:

- dominant sub-genres were preferred over very broad catch-all categories, which would be harder to characterize and classify (i.e. engagement summaries was preferred over engagement materials and tutorials and labs was preferred over courses and training materials);
- textual information genres were preferred over software tool genres, such as source code, tools, and reusable assets, as these types of files would be difficult to classify using text-based automatic classification.
- We reduced the set to 16 genres (marked in bold in Table 1), all of which were identified in at least 3 different sources.

4.3 Characterizing the Core Genres

It was necessary to characterize each genre further to serve as a guide for the manual collection of training data for automatic classification. We developed definitions for each genre class based on descriptions used in the repositories and in web-based dictionaries (onelook.com). We then surveyed ~20 examples of each genre from different document repositories to identify characteristic features of each with respect to purpose, form, content (style and subject matter), and related genres (see Table 2). The definition and features served as operational guidelines for the manual identification of genres for the classification experiments.

Table 2. Two Sample Genre Characterizations

Best Practice description of a proven methodology or technique for achieving a desired result, often based on practical experience	**Purpose:** instruct, recommend
	Form: primarily text, many formats, variable length,
	Style: use of imperatives (you, do, should), term “best practices” will often appear
	Subject matter: new technologies, design, coding
	Related Genres: cookbook, design pattern, documentation, technical article, whitepaper
Cookbook: Step-by-step description of how to implement a particular technology or process.	**Purpose**: demonstrate, guide
	Form: steps are numbered or bulleted; includes examples and/or templates, screenshots and diagrams; length varies
	Style: mix of short sentences and point form, imperatives, and sequencing terms (i.e. do this next): common phrases: “how-to” , “click”, “step-by-step”
	Subject matter: specific products and technologies
	Related genres: documentation, manual, technical article – any of these may contain cookbook sections

4.4 Assessing the Coverage

Given that our taxonomy represented a subset of all genres in this domain, we were interested to know what portion of documents likely to be used by this group would be covered. We extracted a set of 15 domain specific queries, representing a range of work tasks and information goals, from questions asked in online discussion groups used by this group. We then submitted these queries to a search engine with approximately 8 GB of documents gathered from the Internet, the corporate intranet, and from document repositories heavily used by this group. The crawl used to collect the documents was customized for this group based on a targeted set of seed URLS. We combined the top 30 documents from each query; after eliminating a large number of identical and very similar documents (i.e. slight variations on the same page for different versions of the same product), we had a set of 275 unique pages. We manually classified these pages using the taxonomy, allowing for multiple classifications of each page. Due to the time involved and the need to protect confidentiality of the documents, only one judge was used. Based on this assessment, we found that the taxonomy accounted for over 75% of the documents retrieved, and only one class, demo, was not represented in the results (Table 3). Overall, there was an overlap of about 20% among the genre classes, due to documents belonging to more than one class.

Table 3. Distribution of Results Classified by Genre Taxonomy

Class	**# docs per class**	**% of docs in class**
not classified	63	22.9
product documentation	69	25.1
cookbook	37	13.5
technical article	36	13.1
discussion threads	25	9.1
manual	20	7.3
presentation	17	6.2
design pattern	15	5.5%
best practice	11	4.0%
FAQ	10	3.6%
product page	10	3.6%
whitepaper	9	3.3%
technote	4	1.5%
tutorial	4	1.5%
engagement summary	3	1.1%
problem report	1	0.4%
demo	0	0%
Total	334	121.7%

4.5 Summary

We conducted a genre analysis targeted specifically to a software engineering workplace domain. We drew upon evidence of genre usage from our user community directly and from the information tools and resources used by the community, and developed an operational genre taxonomy of 16 prevalent genres. The taxonomy is not exhaustive, but our analysis suggests that it represents a significant portion of the documents in common usage by out target population.

5 Automatic Classification

The results of our genre analysis were promising; suggesting that a large portion of the document collection in this domain can be classified using a fairly small set of genre categories. However, our analysis was based on manual classification, an approach that is not scalable in this information environment due to issues of size, data volatility, and decentralization of control. Based on the promising, albeit preliminary, results of work done in automatic genre classification to date, we decided to explore a machine learning approach to classifying our enterprise document collection. The goal of this part of the study was to assess how well simple automatic text classification techniques work in an operational environment, using real world data and a functional taxonomy.

5.1 Method

We made use of SVM Light, an open source classification package, which is readily available [27]. Support vector machines have been shown to perform well in comparison with other methods, including Naïve Bayes, C4.5 decision trees [21] and neural networks [22], and are well-suited to text classification [28]. An SVM classifier uses a set of examples to train separate classifiers for each class. These classifiers run independently over the whole collection, so that each object can be assigned to multiple categories. This approach is well-suited to our operational taxonomy. Due to the challenges of extracting structural document features from a collection in a wide range of file formats, we elected to use a simple ‘bag of words’ approach with no feature selection. We did not use stemming, which is common in subject-based classification, as there is insufficient evidence to suggest that word stems are more expressive of genre than inflected terms.

SVM Light uses tuning parameters to set the rate of error tolerance and the relative weight of positive versus negative examples in the training set. We reasoned that since genre was to be used as a supplementary filter for search results in our proposed application, high recall was more important than precision. That is, we would prefer to err on the side of including a document in a class, rather than excluding it. In lieu of any guidelines on setting these parameters in this type of application, we used trial and error and set parameters uniformly for all classes. In the results below, we compare performance both on the training set and a sample of the corpus data for two different settings of these parameters.

Using the definitions and class descriptions developed for the taxonomy as guidelines, we manually collected a set of training data from websites, databases and document repositories used by the target population. We collected almost 800 documents, consisting of approximately 50 examples for each of the 16 different genres in the taxonomy. Although there is some natural overlap in the taxonomy, as indicated in section 4.4 above, we initially assigned each document in the training set to a single class. In order to avoid over-training of the classifiers to specific document templates that exist within different repositories, we made an effort to collect examples of each genre from a range of different locations within the organizational information space. Figure 1 shows the distribution of the training data across more than 50 host Internet domains, with each shade representing a different server domain from which examples were extracted. Some genres are strongly associated with specific repositories and domains, and others are spread widely through the information space.

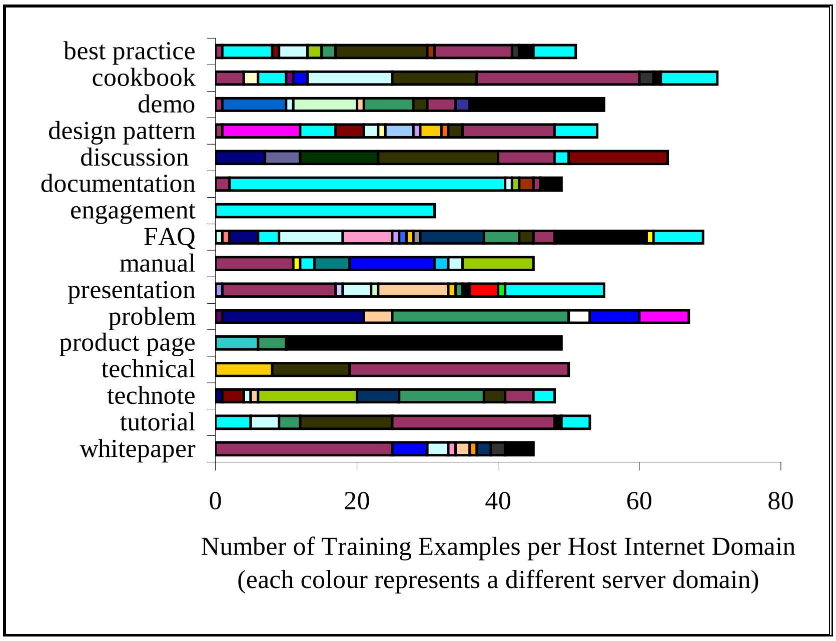

Fig. 1. Distribution of training examples across organizational host Internet domains

5.2 Evaluation

After the classifiers were trained, we evaluated performance using the training set and a small corpus of manually classified enterprise data. Based on the results of the first run, we adjusted the tuning parameters and evaluated a second run. Results for the two runs are presented below.

5.2.1 Performance Estimates from Training Corpus

SVM Light computes a leave-one-out cross validation on the training data to provide performance estimates. Table 4 provides the results obtained on the training data from two runs. The first results are based on a relatively high error tolerance and a heavy weight in favour of positive examples. Using these settings, we achieved the high recall we were hoping for, but the cost in terms of precision meant that there was heavy overlap among the classes. For the next run, we reduced the error tolerance to achieve a more balanced trade-off between recall and precision.

Some classifiers clearly perform better than others, as there is a wide variation in precision when high recall levels are set. Because the classes in the taxonomy are not mutually exclusive, low precision could be the result of classification error or of inherent ambiguity. To test this, we manually re-classified the training data, assigning multiple classes where applicable. We found about 25% overlap. A number of the genres with low levels of precision using automatic classification (best practices, cookbooks and documentation) also show heavy overlap in manual classification, suggesting that some of the low precision is, indeed, due to ambiguous classes. However, in other cases of low precision (technical articles, manuals, and presentations), there was no overlap in manual classification, suggesting that classification error is the cause. The most interesting difference between the two runs is for FAQs, which jumps from the lowest precision overall to the second highest.

Table 4. Precision Recall Estimates on Training Data

	Run 1		Run 2	
	%recall	%precision	%recall	%precision
best practice	90	19	64	49
cookbook	97	22	83	41
demos	91	94	82	100
design patterns	89	54	74	82
discussion thread	97	74	94	94
documentation	90	39	63	60
engagement summary	83	52	73	82
faq	99	13	65	96
manual	98	71	98	76
presentation	88	31	66	72
problem report	97	81	96	91
product page	96	68	92	92
technical article	96	21	84	48
technotes	81	36	63	68
tutorial	84	83	78	93
whitepaper	91	37	78	56
overall average	*92*	*52*	*79*	*76*

5.2.2 Real-World Data – Preliminary Results

After training the classifiers for each of the two runs, we crawled and classified around 8 GB of live data with each, and created evaluation collections using the same method described in section 4.4. We then had three separate sets of results for the same set of 15 queries; one classified manually, one classified using the Run 1 classifiers, and one classified using Run 2 classifiers. The results retrieved in the automatically classified runs were filtered, so they did not include any unclassified documents. We then assessed the automatic classification of all documents that had been manually classified and coded each as: 1-exact match, 2-includes all manually assigned classes plus more, 3-includes some of the manually assigned classes, or 4-no match. It should be noted that the samples used for this evaluation were quite small, and thus, this can only be considered a preliminary evaluation. The results are presented in Figure 2.

In both cases, over 50% of the results are either an exact match or include all of the classes assigned manually. Run 2 clearly shows stronger performance in that the number of exact matches is higher and there are fewer no matches. The main difference is that Run 1 has much lower precision, assigning 2.8 classes per document on average, in contrast to 1.5 in Run 2, and 1.2 assigned manually. However, it is likely that the cost of the higher precision in Run 2 is in the number of documents left unclassified, which does not show up in this evaluation. Because of the small sample size of this evaluation set, it was not possible to calculate precision and recall for each class, however, indications are that average precision and recall for the real-world data is well below the estimates obtained on the training data. Further evaluation needs to be done with a larger collection of manually classified data to understand the tradeoffs involved and the variation across different genre classes.

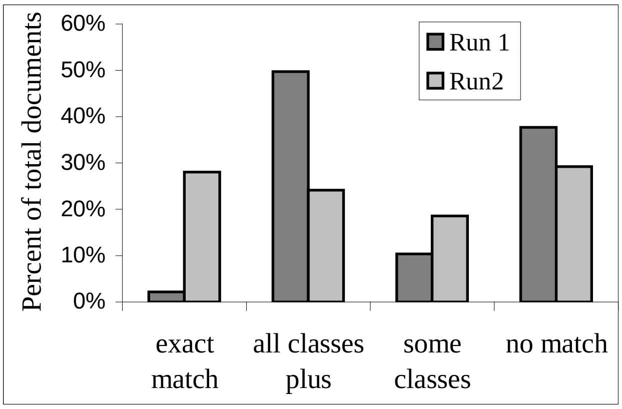

Fig. 2. Evaluation and comparison of two classification runs on real world data

5.3 Summary

We conducted a preliminary assessment of automatic genre classification using supervised machine learning and a "bag of words" approach. Performance estimates on the training data show significant variation between the genre classes. For some genres, such as problem reports and demos, the classifiers were able to achieve high rates of both recall and precision, but for others, such as technical articles, the cost of forcing high recall was unacceptably low rates of precision. The second run was more balanced and achieved average precision and recall in the high 70% range. Evaluation of the classifiers on a real world document collection indicates lower rates overall, but further evaluation needs to be done to assess performance for individual genres.

6. Discussion and Conclusions

The genre taxonomy presented here is intended to be operational rather than ideal. It is not exhaustive: it does not include the long tail composed of less common genres and specific sub-genres, nor does it include genres commonly used by other communities in the organization dealing with the same products and services. Neither are the categories mutually exclusive. Since genre is a composite attribute, made up of form, content and function, organic genres do not tend to be sharply delineated on all dimensions. A *best practices* document, which is defined primarily through its function, can be packaged in the form of a *presentation* or an *FAQ*. This type of classification scheme is more difficult to implement and evaluate, but has the most potential to reflect the way genre is actually used in information seeking and, therefore, can be of value for contextual IR systems.

A relatively small set of core genres did emerge quite clearly from this analysis. Given that over 75% of the search results in our evaluation could be classified using this set, we think that it has sufficient coverage to be of benefit. Through further evaluation, it would be possible to refine the taxonomy to weed out less common genres, such as demos, and those that do not prove to be strongly correlated with task. The taxonomy was relatively easy to develop given the extent to which genre is dominant in this information space, and it is likely that this would be true of many organizational settings.

Implementing the taxonomy is a much more challenging matter. The results of our initial experiments using a simple ‘bag of words’ approach are far from optimal, but are encouraging none the less. The extent to which genre can be identified based on textual features alone is surprising, as is the sensitivity of this method to identifying similarities among genre classes. For some classes, we were able to achieve reasonable results using textual features alone, but for others it seems that additional, perhaps non-textual, features are needed. In our manual analysis of genres, we identified a large number of grammatical and structural features, as well as some heuristics that have the potential of improving the automatic classification.

One of the open questions we are left with is, how good does genre classification need to be? When genre is used to display classified results or to provide a browsing structure, it clearly needs to be very accurate; however, if it is simply one element in the ranking algorithm in a task-based IR system, lower levels of accuracy may be acceptable. This will be difficult to determine until more research is done in this area.

Acknowledgments
This research is supported by an IBM CAS Fellowship, and SSHRC and Canada Research Program grants to the third author. Thanks to Julie Waterhouse and Peter Yeung for all their valuable help with this project.

References

1. Hawking, D.: Challenges in Enterprise Search, presented at the Australasian Database Conference, Dunedin, New Zealand (2004)
2. Fagin, R. Kumar, K. S. McCurley, J. Novak, D. Sivakumar, Tomlin, J. A., Williamson D. P.: Searching the Workplace Web. presented at WWW '03 International World Web Conference (2003)
3. Broder, A., Ciccolo, A.C.: Towards the Next Generation of Enterprise Search Technology. IBM Systems Journal 43 (2004) 451-454
4. Jarvelin, K., Ingwersen, P.: Information Seeking Research Needs Extensions towards Tasks and Technology. Information Research 10 (2004)
5. Anderson, C. J., Glassman, M, McAfee R. B., Pinelli, T.: An Investigation of Factors Afffecting how Engineers and Scientists Seek Information. Journal of Engineering and Technology Management 18 (2001) 131-155
6. Hertzum, M.: The Importance of Trust in Software Engineers' Assessment and Choice of Information Sources. Information and Organization 12 (2002) 1-18
7. Fidel, R., Green, M.: The Many Faces of Accessibility: Engineers' Perception of Information Sources. Information Processing & Management 40 (2004) 563-581
8. Freund, L., Toms, E.G., Clarke, C.L.A.: Modeling Task-Genre Relationships for IR in the Workplace. Annual International ACM SIGIR Conference, Salvador, Brazil (2005)
9. Freund, L., Toms, E.G., Waterhouse, J.: Modeling the Information Behaviour of Software Engineers using a Work - Task Framework. 68th Annual Meeting of the American Society for Information Science and Technology, Charlotte, NC (2005).
10. Vakkari, P.: Task-Based Information Searching. Annual Review of Information Science and Technology 37 (2003) 413-463
11. Bystrom, K., Hansen, P.: Conceptual Framework for Tasks in Information Studies. Journal of the American Society for Information Science and Technology 56 (2005) 1050-1061
12 Bystrom, K.: Information and Information Sources in Tasks of Varying Complexity. Journal of the American Society for Information Science 53 (2002) 581-591
13. Orlikowski, W. J., Yates, J.: Genre Repertoire: the Structuring of Communicative Practices in Organizations. Administrative Science Quarterly 39 (1994) 541-574
14. Toms, E. G.: Recognizing Digital Genre. Bulletin of the American Society for information Science and Technology 27 (2001) 20-22
15. Yoshioka, T., Herman, G., Yates, J., Orlikowski, W.J.: Genre Taxonomy: a Knowledge Repository of Communicative Actions. ACM Transactions on Information Systems 19 (2001) 431-456
16. Karlgren, J.: Non-Topical Factors in Information Access. Webnet '99, Honolulu, 1999.
17. Roussinov, D. G., Crowston, K., Nilan, M., Kwasnik, B., Cai, J., Liu, X.: Genre Based Navigation on the Web. presented at Hawai'i International Conference on Systems Sciences, Maui, Hawai'i (2001)
18. Muresan, G., Smith, C.L., Cole, M. Liu, L. Belkin, N.J.: Detecting Document Genre for Personalization in Information Retrieval. Proceedings of the Hawaii International Conference on System Sciences, Kauai, Hawai'I (2006)

19. Bretan, I., Dewe, J., Hallberg, A., Wolkert, N., Karlgren, J.: Web-Specific Genre Visualization. presented at WebNet '98, Orlando Florida (1998)
20. Glover, E. J., Lawrence, S., Gordon, M.D., Birmingham, W.P., Giles, C.L.: Web Search -- Your Way. Communications of the ACM 44 (2001) 97-102
21. Dewdney, N., VanEss-Dykema, C., MacMillan, R.: The Form is the Substance: Classification of Genres in Text. Proceedings of ACL Workshop on Human Language Technology and Knowledge Management (2001)
22. Meyer zu Eissen, S., Stein, B.: Genre Classification of Web Pages. Proceedings of the 27th German Conference on Artificial Intelligence, Ulm, Germany (2004)
23. Lee, Y.-B., Myaeng, S.H.: Text Genre Classification with Genre Revealing and Subject-Revealing Features. Proceedings of the 25th Annual International ACM SIGIR Conference (2002)
24. Stamatatos, E., Fakotakis, N., Kokkinakis, G.: Text Genre Detection using Common Word Frequencies. Proceedings of the 18th International Conference on Computational Linguistics (2000)
25. Finn, A., Kushmerick, N.: Learning to Classify Documents according to Genre. presented at IJCAI Workshop on Computational Approaches to Style Analysis and Synthesis (2003)
26. Shepherd, M., Watters, C., Kennedy, A.: Cybergenre: Automatic Identification of Home Pages on the Web. Journal of Web Engineering 3 (2004) 236-251
27. Joachims, T.: Learning to Classify Text using Support Vector Machines: Methods, Theory and Algorithms, Kluwer, Amsterdam (2002)
28. Joachims, T.: Text Categorization with Support Vector Machines: Learning with Many Relevant Features. Proceedings of the European Conference on Machine Learning (1998)